

POLITECHNIKA WARSZAWSKA

DYSCYPLINA NAUKOWA - ARCHITEKTURA I URBANISTYKA/
DZIEDZINA NAUK - NAUKI INŻYNIERYJNO-TECHNICZNE

Rozprawa doktorska

mgr inż. arch. Tomasz Dzeduszyński

**Narzędzia projektowania hiperparametrycznego. Analiza
kontekstów kompozycyjnych przy pomocy sieci neuronowych.**

Promotor

dr hab. inż. arch. Maciej Lasocki

WARSZAWA 2024

*Z dedykacją dla najukochańszej
Żony, której doktorat był główną
inspiracją dla tej pracy.*

Streszczenie

Niniejsza praca doktorska pt. „Narzędzia projektowania hiperparametrycznego. Analiza kontekstów kompozycyjnych przy pomocy sieci neuronowych” bada możliwości i ograniczenia wykorzystania sieci neuronowych w architekturze i urbanistyce, koncentrując się na zagadnieniach przetwarzania kompozycji przestrzennych oraz automatycznym rozpoznawaniu i analizie głównych punktów kompozycyjnych w przestrzeni miejskiej.

Część teoretyczna rozpoczyna się od rozdziału wstępnego, w którym zdefiniowano problem badawczy, metodykę pracy oraz cele i pytania badawcze. Rozdział drugi opisuje aktualny stan wiedzy, w tym rozwój architektury parametrycznej oraz jej ewolucję w kierunku architektury hiperparametrycznej, wprowadzającej do praktyki architektonicznej głębokie uczenie maszynowe. W rozdziale przedstawione są również zastosowania sieci neuronowych w różnych skalach projektowych: urbanistycznej, architektonicznej i detalu. W związku z wykrytą luką badawczą dotyczącą neuronowego przetwarzania kontekstów kompozycyjnych, autor analizuje zagadnienie punktów głównych, opierając się na klasycznych teoriach kompozycji przestrzennej. W swojej analizie bierze pod uwagę aspekty geometryczne, procesy powstawania tych punktów, ich skalę, a także znaczenie w kontekście społecznym, symbolicznym i ekonomicznym. Rozdział kończy się identyfikacją narzędzi opartych o głębokie uczenie maszynowe, mogących służyć do analizy kompozycji przestrzennych w projektowaniu architektonicznym wspomagany komputerowo.

Część eksperymentalna pracy, stanowiąca rozdział trzeci dysertacji, obejmuje dwa eksperymenty: wstępny i główny. W podrozdziale opisującym eksperyment wstępny testowane są różne podejścia do neuronowego przetwarzania syntetycznych założeń przestrzennych opartych na złożonych zasadach kompozycyjnych. Doświadczenia zdobyte przy eksperymentcie wstępnym prowadzą do opracowania bardziej zaawansowanego algorytmu w eksperymentcie głównym. Stworzony prototyp, oparty o przeciwstawne, generatywne sieci neuronowe, służy do rozpoznawania głównych punktów kompozycyjnych i osi naprowadzających w tkance miejskiej.

W podsumowaniu omówione są wyniki eksperymentów i badań teoretycznych oraz ograniczenia wynikające z zastosowanych metod i obecnego stanu wiedzy. Autor rozważa potencjalne korzyści płynące z wprowadzenia głębokiego uczenia maszynowego do praktyki architektonicznej, ocenia praktyczne możliwości zastosowania tego typu narzędzi i przedstawia możliwe kierunki dalszego rozwoju dziedziny.

Słowa kluczowe

Projektowanie architektoniczne wspomagane komputerowo, punkty główne, kompozycja przestrzenna, głębokie uczenie maszynowe, sieci neuronowe, projektowanie hiperparametryczne

Abstract

The doctoral dissertation titled “Hyperparametric design tools. Analysis of compositional contexts using neural networks,” explores the possibilities and limitations of utilizing neural networks in architecture and urban planning. The research focuses on spatial composition processing and automatic recognition and analysis of focal compositional points in urban spaces.

The theoretical part begins with an introductory chapter that defines the research problem, methodology, as well as research goals and questions. The second chapter describes the current state of the art, including the development of parametric architecture and its evolution towards hyperparametric architecture, which introduces deep learning into architectural practice. This chapter also presents application of neural networks at various design scales: urban, architectural and detail. In response to the identified research gap concerning the neural processing of compositional contexts, the author examines the issue of main focal points based on classical spatial composition theories. The analysis considers the geometric aspects, the formation process of these points, their scale, as well as their significance in social, symbolic and economic contexts. The chapter concludes with the identification of deep learning-based tools that can be used for the analysis of spatial compositions in computer-aided architectural design.

The experimental part, which forms the third chapter of the dissertation, consists of two experiments: preliminary and main. In the subchapter describing the preliminary experiment, various approaches to neural processing of synthetic, complex spatial compositions are tested. The experiences gained from the preliminary experiment lead to the development of a more advanced algorithm in the main experiment. The created prototype, based on generative adversarial networks, serves to recognize main focal points and guiding axes in urban tissue.

The conclusion discusses the results of the experiments and theoretical studies, as well as the limitations stemming from the applied methods and the current state of the art. The author considers the potential benefits of introducing deep machine learning into architectural practice, assesses the practical applicability of such tools, and outlines possible directions for further development of the field.

Keywords

Computer-aided architectural design, focal points, spatial composition, deep learning, neural networks, hyperparametric design

Spis treści

Wykaz wykorzystanych akronimów i tłumaczenia pojęć obcojęzycznych	11
1. Wstęp.....	15
1.1. Problem badawczy	15
1.2. Struktura pracy	16
1.3. Metodyka pracy.....	17
1.4. Cele badawcze	19
1.4.1. Cele badania stanu wiedzy	19
1.4.2. Cele dotyczące zidentyfikowanej luki badawczej.....	19
1.4.3. Cele eksperymentu wstępnego	19
1.4.4. Cele eksperymentu głównego	20
1.5. Pytania badawcze	20
1.5.1. Pytanie główne	20
1.5.2. Pytania szczegółowe	20
2. Stan wiedzy	21
2.1. Od architektury parametrycznej do hiperparametrycznej	21
2.2. Sieci neuronowe w literaturze dotyczącej komputerowego wspomagania projektowania	27
2.2.1. Skala urbanistyczna – zastosowania sieci neuronowych	30
Urbanistyka: ekstrakcja i uczytnianie danych	30
Urbanistyka: ewaluacja i przetwarzanie danych	34
Urbanistyka: klasyfikacja	38
Urbanistyka: przewidywanie przyszłości	41
Urbanistyka: generowanie rozwiązań projektowych	43
Urbanistyka: optymalizacja.....	45
2.2.2. Skala architektoniczna – zastosowania sieci neuronowych	46
Architektura: generowanie rozwiązań projektowych.....	46
Architektura: ewaluacja i przetwarzanie danych	57

Architektura: klasyfikacja	64
Architektura: ekstrakcja i uczytnianie danych	68
Architektura: optymalizacja	69
2.2.3. Detal, materiały i komputerowe wspomaganie wytwarzania (CAM) – zastosowania sieci neuronowych.....	71
Detal, materiały, CAM: ewaluacja i przetwarzanie danych	71
Detal, materiały, CAM: algorytmy generatywne	75
Detal, materiały, CAM: optymalizacja	76
Detal, materiały, CAM: klasyfikacja.....	76
Detal, materiały, CAM: przewidywanie przyszłości	77
2.3. Luka badawcza – neuronowe przetwarzanie kontekstów kompozycyjnych, przesłanki	78
2.4. Punkty główne kompozycji przestrzennej.....	80
2.4.1. Elementy kompozycyjne i ich przetwarzanie maszynowe	80
2.4.2. Punkty główne w teorii kompozycji przestrzennej	85
2.4.3. Punkty główne spontaniczne i projektowane	87
2.4.4. Punkty główne a skala kompozycji	88
2.4.5. Znaczenia punktów głównych.....	89
Znaczenie funkcjonalne i społeczne.....	89
Znaczenie symboliczne	92
Znaczenie ekonomiczne	92
2.4.6. Rola punktów głównych w planowaniu przestrzennym	94
2.4.7. Automatyczne rozpoznawanie punktów głównych.....	98
2.5. Uwarunkowania do wykorzystania sieci neuronowych w CAAD	105
2.5.1. Sieci neuronowe – narzędzie wielu dyscyplin	105
2.5.2. Rodzaje sieci neuronowych wykorzystywanych w CAAD	114
Jednokierunkowe sieci neuronowe (FNN).....	116
Splotowe sieci neuronowe (CNN).....	118

Generatywne sieci przeciwstawne (GAN)	120
Głębokie uczenie ze wzmacnianiem (DRL).....	121
Rekurencyjne sieci neuronowe (RNN) i sieci długiej pamięci krótkotrwałej (LSTM)	123
Inne sieci neuronowe, transformery, modele dyfuzyjne	125
3. Część eksperymentalna	127
3.1. Eksperyment wstępny	127
3.1.1. Cele szczegółowe eksperymentu wstępnego.....	127
3.1.2. Metodyka eksperymentu wstępnego	127
3.1.3. Wyniki eksperymentu wstępnego	132
Generowanie całości kompozycji, scenariusz I.....	132
Generowanie fragmentu kompozycji, scenariusz II	134
Generowanie pojedynczych pikseli, scenariusz III	136
3.1.4. Wnioski z eksperymentu wstępnego	139
3.2. Eksperyment główny	141
3.2.1. Cele szczegółowe eksperymentu głównego	141
3.2.2. Metodyka eksperymentu głównego	143
Kodowanie kontekstów przestrzennych – mapa cech.....	143
Zakres pola percepcji	148
Wybór modelowych punktów głównych, tworzenie zbioru treningowego i testowego	149
Oznaczanie punktów głównych – mapa celu	155
Struktura sieci neuronowej	157
Testowanie algorytmu	159
3.2.3. Wyniki eksperymentu głównego.....	162
Wyniki ilościowe.....	162
Wsparcie projektowo-planistyczne	165
Uczytelnianie kompozycji miejskiej	167

3.2.4. Wnioski z eksperymentu głównego	171
4. Podsumowanie	173
4.1. Wnioski ogólne	173
4.2. Ograniczenia metody.....	174
4.3. Projektowanie hiperparametryczne – dyskusja	175
4.4. Okno na przyszłość	177
Bibliografia.....	181
Spis ilustracji	204
Aneks.....	I
Załącznik 1. Tabela źródeł kwerendy głównej wykorzystania sieci neuronowych w CAAD (do 2021 roku). Zastosowania.	I
Załącznik 2. Tabela źródeł kwerendy głównej wykorzystania sieci neuronowych w CAAD (do 2021 roku). Rodzaj sieci neuronowej, skala zastosowania.	XII
Załącznik 3. Tabela źródeł kwerendy głównej wykorzystania sieci neuronowych w CAAD (do 2021 roku). Tematyka, źródła danych.	XXIII
Załącznik 4: Lista punktów głównych wykorzystanych do treningu sieci neuronowej eksperymentu głównego oraz lista odrzuconych kandydatów.	XXXV
Załącznik 5. Tabela ewaluacji wytrenowanego algorytmu na zbiorze testowym.....	XLVII
Załącznik 6: Lista punktów głównych wykrytych przez sieć neuronową we fragmencie Warszawy	LV
Załącznik 7: Tabela rozmiarów warstw sieci eksperymentu głównego i liczba trenowalnych parametrów	LX
Załącznik 8. Wybór ciekawszych baz danych, które można wykorzystać w trenowaniu architektonicznych sieci neuronowych.	LXI

Wykaz wykorzystanych akronimów i tłumaczenia pojęć obcojęzycznych

AGI: ang. Artificial General Intelligence (silna sztuczna inteligencja)

AI: ang. Artificial Intelligence (sztuczna inteligencja)

Anything goes: ang. „Nic świętego” (hasło oddające istotę anarchizmu epistemologicznego Paula Feyerabenda, w tłumaczeniu Stefana Wiertlewskiego. Hasło tłumaczone jest czasem także jako „wszystko ujdzie”)

API: ang. Application Programming Interface (interfejs programowania aplikacji)

Bias: ang. Próg (w kontekście neuronów – próg aktywacji) lub Upředzenie (w kontekście sieci neuronowych – upředzenie względem struktury danych)

BIM: ang. Building Information Modeling (modelowanie informacji o budynku)

Black box: ang. Czarna skrzynka (algorytm przetwarzający dane wejściowe na wyjściowe w sposób trudny do przewidzenia i interpretacji)

Bottleneck: ang. Zwężenie w sieci typu autoenkoder

BPS: ang. Building Performance Simulation (symulacja wydajności budynku)

Bridging: ang. Mostkowanie. Druk 3D w powietrzu, łączący ze sobą dwa oddalone od siebie punkty bez użycia pośrednich podpór.

CAAD: ang. Computer-Aided Architectural Design (projektowane architektoniczne wspomagane komputerowo)

CAD: ang. Computer-Aided Design (projektowanie wspomagane komputerowo)

CAM: ang. Computer-Aided Manufacturing (wytwarzanie wspomagane komputerowo)

CCPI: ang. Climate Change Performance Index (indeks monitorujących postępy poszczególnych państw w łagodzeniu zmian klimatu)

City Country Fingers: ang. Palczaste zatoki miejsko-wiejskie (wydłużone pasma gęstej zabudowy rozdzielone przez pasma zabudowy wiejskiej zaproponowane w Języku Wzorców przez Alexandra)

CNC: ang. Computerized Numerical Control (komputerowe sterowanie urządzeń numerycznych)

CNN: ang. Convolutional Neural Network (splotowa sieć neuronowa)

Convergence: ang. Zbieganie się parametrów sieci neuronowej

Covariate shift: ang. Przesunięcie dystrybucji między zbiorem treningowym a testowym

CPU: ang. Central Processing Unit (Procesor)

DA: ang. Daylight Autonomy (autonomia światła dziennego)

Data Augmentation: ang. Zautomatyzowane zwiększenie liczby danych treningowych

Data Mining: ang. Ekstrakcja danych

Decision Tree: ang. Drzewo decyzyjne

DDPG: ang. Deep Deterministic Policy Gradient. Jeden z algorytmów sterujących agentem w głębokim uczeniu ze wzmacnianiem

DGP: ang. Daylight Glare Probability (współczynnik prawdopodobieństwa ośnienia światłem dziennym)

DRL: ang. Deep Reinforcement Learning (głębokie uczenie ze wzmacnianiem)

Feature space: ang. Przestrzeń cech

Fine tuning: ang. Strojenie wcześniej wytrenowanego modelu do nowego problemu

FNN: ang. Feedforward Neural Network (jednokierunkowa sieć neuronowa). Czasem akronimem tym określane są sieci neuronowo rozmyte (ang. Fuzzy Neural Networks), jednak w niniejszej pracy akronim FNN odnosi się jedynie do sieci jednokierunkowych (Tadeusiewicz i Szaleniec, 2015)

GAN: ang. Generative Adversarial Network (generatywna sieć przeciwstawna)

GAN loss: ang. Funkcja straty GAN (wykorzystywana w sieciach GAN)

G-code: ang. Geometric code (kod geometryczny). Popularny język instrukcji dla urządzeń CNC

Genius loci: łac. Duch miejsca

GPU: ang. Graphics Processing Unit (procesor graficzny)

Grasshopper: Graficzny język programowania dostępny w programie Rhino wykorzystywany często w projektowaniu parametrycznym

Grid search: ang. Przeszukiwanie siatki (metoda strojenia hiperparametrów)

GRU: ang. Gated Recurrent Unit (uproszczony wariant sieci typu LSTM)

Hierarchical clustering: ang. Klastrowanie hierarchiczne

IAAC: ang. Institute for Advanced Architecture of Catalonia (Instytut Architektury Zaawansowanej w Katalonii)

Integration core: ang. Rdzeń integracji

k-means clustering: ang. Klastrowanie metodą algorytmu centroidów

kNN: ang. k-Nearest Neighbors (k najbliższych sąsiadów)

L1 loss: ang. Funkcja straty L1 (wykorzystywana w CNN i GAN)

Latent Space: ang. Przestrzeń ukryta

LLM: ang. Large Language Model (duży model językowy)

LSTM: ang. Long Short-Term Memory (sieć długiej pamięci krótkotrwałej)

MOQO: ang. Multi-Objective Qualitative Optimization (wielocelowa optymalizacja jakościowa)

MSE: ang. Mean Squared Error (błąd średniokwadratowy)

Multi-hot: ang. Metoda kodowania przypisująca poszczególnym cechom wartości 0 lub 1 i zestawiająca je razem w postaci wielowymiarowego wektora

NLP: ang. Natural Language Processing (przetwarzanie języka naturalnego)

One-hot encoding: ang. Kodowanie „jeden z n”

OSM: ang. Open Street Map

Overfitting: ang. Przetrenowanie parametrów do zbioru treningowego

Overtuning: ang. Przestrojenie hiperparametrów do zbioru testowego

PCA: ang. Principal Component Analysis (analiza głównych składowych)

PPO: ang. Proximal Policy Optimization. Jeden z algorytmów sterujących agentem w głębokim uczeniu ze wzmocnieniem

Prompt: ang. Dane wejściowe, zazwyczaj w postaci tekstowej, na podstawie których model generuje dane wyjściowe. Wyróżnia się różne metody prompting, np. prompting

bezprzykładowy (zero-shot prompting), odpowiedź jednokrotną (one-shot prompting) lub odpowiedź wielokrotną (few-shot prompting).

Q-Learning: ang. Jeden z algorytmów sterujących agentem w głębokim uczeniu ze wzmocnieniem

Random search: ang. Przeszukiwanie losowe (metoda strojenia hiperparametrów)

RF: ang. Random Forest (losowe drzewo decyzyjne)

SL-Block: ang. Self-Interlocking Block (samounieruchamiające się bloki); bryły, które po odpowiednim złożeniu wzajemnie się unieruchamiają umożliwiając budowanie stabilnych struktur

SNARC: ang. Stochastic Neural Analog Reinforcement Calculator; pierwsza sieć neuronowa stworzona przez Marviną Minsky'ego

Sparse model: ang. Model skąpy (w którym wiele parametrów ma wartość zerową)

SVM: ang. Support Vector Machine (maszyna wektorów nośnych)

Target space: ang. Przestrzeń celu

t-SNE: ang. t-Distributed Stochastic Neighbor Embedding; algorytm redukcji wymiarowości, służący do wizualizacji danych wielowymiarowych

Transfer learning: ang. Uczenie transferowe (dostrojenie sieci neuronowej wytrenowanej na jednym zadaniu do innego zadania)

Transformer: ang. Typ sieci neuronowej wykorzystującej mechanizm uwagi

UAV: ang. Unmanned Aerial Vehicle (bezzałogowy statek powietrzny)

ViT: ang. Vision Transformer (transformer wizualny), jeden z nowszych typów sieci neuronowych stosowanych w przetwarzaniu obrazów

Weight: ang. Waga (w kontekście sieci neuronowych – waga połączeń między neuronami)

WiFi: ang. Wireless Fidelity (zestaw standardów bezprzewodowych sieci komputerowych)

WFC: ang. Wave Function Collapse (kolaps funkcji falowej); samoorganizujący się algorytm generatywny opracowany przez Maxa Gumina

1. Wstęp

1.1. Problem badawczy

Architektura stoi na progu przełomowych zmian. Głębokie uczenie maszynowe, które od połowy poprzedniej dekady zrewolucjonizowało już wiele dziedzin, zaczyna teraz znacząco wpływać na praktykę architektoniczną. Komputery nie służą już jedynie do przeprowadzania obliczeń matematycznych i automatyzacji łatwych do sparametryzowania procesów. Sieci neuronowe przenikają do coraz większej liczby obszarów twórczych, takich jak sztuki wizualne, literatura czy muzyka, które do tej pory były domeną ludzi. Współcześni architekci używają komputerów i narzędzi cyfrowych na bardzo zaawansowanym poziomie. Nie dziwi więc, że według raportu Royal Institute of British Architects, już na początku 2024 roku 41% brytyjskich biur architektonicznych korzystało w praktyce projektowej z narzędzi opartych na głębokim uczeniu maszynowym (RIBA, 2024).

Ze względu na możliwości nowoczesnych sieci neuronowych, należy się spodziewać, że coraz więcej procesów twórczych będzie można zautomatyzować. Sieci neuronowe potrafią rozwiązywać złożone problemy nieliniowe, przetwarzać dane multimodalne, dobrze skalują się do zadań o dużym stopniu skomplikowania i skutecznie radzą sobie z zadaniami generatywnymi. Dodatkowo, możliwość uczenia się na ogromnych zbiorach danych pozwala im na uwzględnienie różnorodnych kontekstów przestrzennych i kulturalnych. W procesie integracji algorytmów neuronowych z praktyką architektoniczną konieczne jest jednak zwrócenie uwagi na kontrolę nad hiperparametrami, czyli zbiorem zmiennych sterujących zachowaniem wykorzystanych sieci, oraz na poziom abstrakcji rozwiązywanych problemów. Obecnie stosowane algorytmy zazwyczaj nie były tworzone z myślą o projektowaniu architektonicznym i są bardzo trudne w sterowaniu. Na dzisiejszym poziomie zaawansowania dostępnych technologii nie można powiedzieć sieci neuronowej „zaprojektuj dom” lub „zaplanuj miasto” i oczekiwać sensownych rezultatów. Cały proces projektowy składa się z dziesiątek bardziej szczegółowych zagadnień projektowych, które lepiej rozwiązywać osobno po kolei.

Jednym z takich kluczowych zagadnień jest kompozycja przestrzenna. Analizy kompozycyjne są przeprowadzane we wszystkich skalach projektowych, a także w praktyce planistycznej i administracyjnej. Automatyzacja czasochłonnych i powtarzalnych procesów związanych z przetwarzaniem kontekstów kompozycyjnych może zwiększyć efektywność pracy, skrócić

czas realizacji projektów i poszerzyć perspektywy inspiracyjne. Konteksty przestrzenne są bardzo złożone, a analizy kompozycyjne logicznie rozmyte i trudne do oprogramowania metodami klasycznymi. W takiej kategorii problemów zazwyczaj stosowane metody automatyzacji (np. oparte na ilościowych analizach GIS) nie są wystarczające. Sieci neuronowe jednak radzą sobie z takimi zadaniami bardzo dobrze. W niniejszej pracy przyglądam się właśnie temu aspektowi projektowania. Po wstępnym rozpoznaniu ogólnych możliwości przetwarzania przez sieci neuronowe złożonych zasad kompozycyjnych, bardziej szczegółowo zajmuję się problemem punktów głównych w kompozycji przestrzennej. Wybór tego konkretnego zagadnienia pozwala na precyzyjne zademonstrowanie potencjału głębokiego uczenia maszynowego w praktyce architektonicznej.

1.2. Struktura pracy

Dysertacja podzielona jest na dwie główne części: teoretyczną i eksperymentalną. W części teoretycznej zaczynam od rozpoznania obecnego stanu wiedzy dotyczącego projektowania architektonicznego wspomagane komputerowo, identyfikuję pierwsze próby wprowadzenia uczenia maszynowego do architektury i badam stopniowe przesunięcie paradygmatu od architektury parametrycznej do hiperparametrycznej. Następnie przez kwerendę literatury najnowszych zastosowań sieci neuronowych w projektowaniu urbanistycznym, architektonicznym i w skali detalu rozpoznaję możliwości współczesnych sieci neuronowych i identyfikuję potencjalne luki badawcze. W dalszej części przyglądam się klasycznym teoriom kompozycji przestrzennej, identyfikuję elementy kompozycyjne szczególnie nadające się do przetwarzania maszynowego i zawężam tematykę do punktów głównych kompozycji przestrzennej.

W części eksperymentalnej przeprowadzam dwa eksperymenty. Na etapie eksperymentu wstępnego testuję algorytmy głębokiego uczenia maszynowego różnego typu w przetwarzaniu syntetycznie stworzonych kontekstów kompozycyjnych. Najlepiej radząca sobie z problemem splotowa sieć neuronowa jest następnie testowana w trzech scenariuszach o zróżnicowanym poziomie trudności. W każdym scenariuszu sieć ma za zadanie odczytać zadaną w zbiorze treningowym złożoną zasadę kompozycyjną i na jej podstawie wygenerować samodzielnie nową kompozycję lub jej fragment. W oparciu o doświadczenia zebrane na przykładach syntetycznych w eksperymencie wstępnym, w eksperymencie głównym buduję prototyp służący do przetwarzania rzeczywistych kontekstów przestrzennych w tkance miejskiej. Na tym etapie opracowuję metodę kodowania kontekstów przestrzennych do postaci nadającej

się do przetwarzania przez generatywne sieci przeciwstawne. Wytrenowana w eksperymencie głównym sieć neuronowa, służąca do rozpoznawania głównych punktów kompozycyjnych i powiązanych z nimi osi naprowadzających, jest oceniona metodami jakościowymi i ilościowymi.

W podsumowaniu pracy prezentuję wnioski płynące z badań teoretycznych i przeprowadzonych eksperymentów, przedstawiam ograniczenia wynikające ze stanu wiedzy i wykorzystanych metod, oraz omawiam wpływ architektury hiperparametrycznej, wykorzystującej głębokie uczenie maszynowe, na rozwój poddziedziny projektowania architektonicznego wspomagane komputerowo.

1.3. Metodyka pracy

Architektura jest dziedziną łączącą naukę, inżynierię i sztukę. Z jednej strony próbuje naukowo rozwiązywać pytania o naturę rzeczy, wykorzystuje metody badań socjologicznych i środowiskowych, odwołuje się do ludzkiej biologii i psychologii oraz stara się opisywać siły sterujące rozwojem ulic, miast i regionów. Z drugiej strony inżyniersko tworzy narzędzia i technologie, bada przez projektowanie i eksperymenty, aktywnie wpływa na rzeczywistość. Z trzeciej strony z kolei operuje na dziełach sztuki, eksploruje rzeczy nieoczywiste, pyta o przekonania, twórczo interpretuje rzeczywistość i zamiast odpowiedzi poszukuje jeszcze głębszych pytań. Jakich metod powinno się używać w badaniach architektonicznych? Jakie są ich podstawy filozoficzne? W swoim podręczniku *Metody i Techniki Badawcze w Architekturze* Elżbieta Niezabitowska nazywa architekturę dziedziną przedparadygmatyczną, w której dopiero tworzą się jej podstawy naukowe. Na tym etapie nie ma niekwestionowanych teorii, symboli, metapojęć, wartości i wzorców rozwiązywania problemów naukowych (Niezabitowska, 2014). Stan dziedziny stawia badaczy architektury w obliczu pewnego anarchizmu metodologicznego, który można podsumować hasłem Paula Feyerabenda: „nic świętego” [ang. „anything goes” (Feyerabend, 2001)]. Zależnie od punktu widzenia sytuacja ta stanowi pewien problem, ponieważ utrudnia wybór oczywistych metod badawczych i uniemożliwia podążanie utartymi i sprawdzonymi szlakami. Z drugiej strony jednak, pragmatystycznie otwiera perspektywy na nowe, mieszane ścieżki eksploracji tematów badawczych (Creswell, 2013) i umożliwia tworzenie nieoczywistych powiązań interdyscyplinarnych.

Badania architektoniczne często oparte są na dogmatycznie stawianych tezach, popartych anegdotycznym wyborem przykładów potwierdzających głoszoną ideę (Niezabitowska, 2014). Zgodnie z założeniami racjonalizmu krytycznego Karla Poppera uważam, że twierdzenia naukowe nie mogą być weryfikowane przez ich indukcyjne potwierdzenie, tylko przez ich dedukcyjną falsyfikację (Popper, 1977). Niniejsza praca nie ma charakteru naukowo empirycznego i opiera się głównie na wnioskowaniu indukcyjnym. Z tego powodu przeprowadzone badania zostały oparte nie na hipotezach, a na jakościowych i ilościowych pytaniach badawczych odwołujących się do postawionych celów badawczych. Podobnie jak sama dziedzina, niniejsza praca ma charakter metodycznie mieszany. W przeprowadzonych badaniach wykorzystałem metody badań teoretycznych (analiza stanu wiedzy, zastosowanie modelowania matematycznego, algorytmizacja procesów), badań rozwojowych i stosowanych (opracowanie koncepcji praktycznego wykorzystania teorii w projektowanym prototypie), badań eksperymentalnych (testowanie opracowanych narzędzi w scenariuszach testowych, zapisywanie i analiza wyników), badań symulacyjnych (użycie modeli komputerowych do przeprowadzania symulacji mających odzwierciedlać rzeczywistość), badań ilościowych (statystyczna analiza danych, opracowanie autorskich metryk i ich wykorzystanie w ewaluacji stworzonych algorytmów) oraz badań jakościowych (studia przypadków, metoda ekspercka, analityczno-syntetyczne przetwarzanie danych). Tematyka doktoratu ma charakter interdyscyplinarny. Z jednej strony opiera się na architektonicznych teoriach kompozycji przestrzennej i artystycznych conceptach wrażeń estetycznych. Z drugiej, czerpie z inżynierskich osiągnięć dziedzin głębokiego uczenia maszynowego, analizy danych i programowania. W pracy odwołuję się do szeregu teorii architektonicznych. Najważniejszymi punktami wyjścia są: teoria formy architektonicznej Juliusza Żórawskiego (Żórawski, 1962), teoria struktury przestrzeni urbanistycznej (Lynch, 1960; Wejchert, 1984; Pluta, 2001), język wzorców Alexandra (Alexander, i in., 1977), ale także architektura informacyjna (Słyk, 2012), czy w szerszym znaczeniu, projektowanie architektoniczne wspomagane komputerowo (Mitchell, 1975).

1.4. Cele badawcze

1.4.1. Cele badania stanu wiedzy

- Ulokowanie głębokiego uczenia maszynowego na tle teorii architektury informacyjnej, architektury parametrycznej i projektowania architektonicznego wspomagane komputerowo.
- Rozpoznanie dotychczasowych zastosowań sieci neuronowych w skali urbanistycznej, architektonicznej i w detalu.
- Rozpoznanie kategorii problemów architektonicznych komplementarnych do możliwości głębokiego uczenia maszynowego.
- Uporządkowanie stanu wiedzy dotyczącego możliwości wykorzystania sieci neuronowych w architekturze na podstawie ich historii i zastosowań w dziedzinach pokrewnych.
- Identyfikacja luki badawczej między możliwościami wykorzystania sieci neuronowych, a ich rzeczywistym zastosowaniem w badaniach i praktyce architektonicznej.

1.4.2. Cele dotyczące zidentyfikowanej luki badawczej

- Ulokowanie problemu neuronowego przetwarzania kontekstów kompozycyjnych na tle polskiego prawa, teorii formy architektonicznej, teorii struktury przestrzeni urbanistycznej i języka wzorców.
- Identyfikacja elementów kompozycyjnych możliwych do przetwarzania maszynowego i ich wybór na potrzeby demonstracji w części eksperymentalnej.
- Rozpoznanie problematyki punktów głównych i punktów formalnie podkreślonych i ich znaczenia w kompozycji przestrzennej.
- Rozpoznanie nie-neuronowych, wspomaganych komputerowo metod przetwarzania głównych punktów kompozycyjnych.

1.4.3. Cele eksperymentu wstępnego

- Wstępna demonstracja możliwości przetwarzania złożonych zasad kompozycyjnych przy pomocy sieci neuronowych.
- Odizolowanie problemu kompozycji przestrzennej od ogólnego problemu syntetycznego przetwarzania obrazów i danych wielowymiarowych.

- Opracowanie i przetestowanie procedur potrzebnych do przeprowadzenia eksperymentu głównego.
- Jakościowa i ilościowa ocena przeprowadzonej demonstracji.

1.4.4. Cele eksperymentu głównego

- Demonstracja zdolności algorytmów głębokiego uczenia maszynowego w przetwarzaniu rzeczywistych, urbanistycznych kontekstów przestrzennych.
- Demonstracja głównych założeń projektowania hiperparametrycznego w praktyce.
- Opracowanie prototypowego narzędzia służącego ekstrakcji punktów głównych z miejskich kompozycji przestrzennych.
- Jakościowa i ilościowa analiza skuteczności prototypu.

1.5. Pytania badawcze

1.5.1. Pytanie główne

- Jakie są możliwości i ograniczenia wykorzystania algorytmów głębokiego uczenia maszynowego do przetwarzania kontekstów kompozycyjnych w architekturze i urbanistyce?

1.5.2. Pytania szczegółowe

- Jak głębokie uczenie maszynowe wpisuje się w teorię architektury informacyjnej, architektury parametrycznej oraz komputerowego wspomagania projektowania architektonicznego?
- Jakie są dotychczasowe zastosowania sieci neuronowych w architekturze i urbanistyce w różnych skalach (urbanistycznej, architektonicznej i detalu)?
- Jakie kategorie problemów architektonicznych mogą być skutecznie rozwiązywane przy pomocy głębokiego uczenia maszynowego?
- Jakie są możliwości i ograniczenia wykorzystania sieci neuronowych w architekturze, na podstawie ich historii i zastosowań w dziedzinach pokrewnych?
- Jakie elementy kompozycyjne mogą być efektywnie przetwarzane przez algorytmy głębokiego uczenia maszynowego?
- Czy jest możliwe stworzenie narzędzia do ekstrakcji punktów głównych z miejskich kompozycji przestrzennych przy pomocy sieci neuronowych?

- Jakie są możliwości i efektywność opracowanego prototypu?
- Jakie są perspektywy przyszłego rozwoju narzędzi opartych na sieciach neuronowych w kontekście wspomagania projektowania architektonicznego i urbanistycznego?

2. Stan wiedzy

2.1. Od architektury parametrycznej do hiperparametrycznej

Warsztat architekta od co najmniej 30 lat jest nierozłącznie związany z komputerami. Już w połowie lat 70. William J. Mitchell zauważył potencjał wykorzystania nowoczesnych narzędzi cyfrowych w projektowaniu i zdefiniował podstawy teoretyczne poddziedziny, którą obecnie nazywamy projektowaniem architektonicznym wspomagany komputerowo (ang. CAAD – Computer-Aided Architectural Design) (Mitchell, 1975; Mitchell, 1977). W późniejszych latach w ramach CAAD został sformułowany paradygmat architektury informacyjnej opisujący zbiór trendów i podstaw filozoficznych związanych z projektowaniem w erze informacyjnej (Schmitt, 1999; Saggio, 2013; Słyk, 2012). Zgodnie z jej klasycznym znaczeniem architektura informacyjna zwróciła się w kierunku programowania i wykorzystania komputerów, z nadzieją na zwiększenie możliwości przechowywania i przetwarzania danych, usprawnienie komunikacji i wzmocnienie ludzkich zdolności logicznego myślenia (Schmitt, 1999). Rolą architekta projektującego dla społeczeństwa informacyjnego było programowanie parametrów procesów twórczych na pograniczu architektury, urbanistyki, technik informacyjnych, wytwarzania i użytkowania (Słyk, 2012). Architektura informacyjna sięgnęła po techniki heurystyczne charakterystyczne dla klasycznego programowania algorytmów opartych na wykrytych przez architekta faktach, rozpoznanych wzorcach je łączących i hipotezach nadających kierunek działaniom. W takim rozumieniu projektowania procesów architekt stawiany był w roli demiurga odpowiedzialnego za definicję poszczególnych parametrów rozwiązania napotkanego problemu. Jak sygnalizuje stopniowe przesunięcie paradygmatu w dziedzinach informatycznych, takie dedukcyjne podejście do parametryzowania złożonych algorytmów zdaje się ustępować miejsca metodom indukcyjnym i stochastycznym. W coraz większej sferze problemów programistycznych odchodzi się od heurystyki na rzecz głębokiego uczenia maszynowego. Obecnie pokazuje się komputerom, jak rozwiązywać problemy, zamiast je programować (Hinton i Ng, 2017). Skąd ta zmiana? Historyk architektury Mario Carpo z dużą dozą nieufności do tego, jaki wpływ na teorię architektury mają narzędzia oparte o sieci neuronowe, zwraca uwagę na przełom

w sposobie myślenia o praktyce architektonicznej i nazywa ten proces „drugim cyfrowym zwrotem”:

„(...) Ten nowy typ nauki, nieodłączny od większości dzisiejszych zastosowań SI¹ jest w gruncie rzeczy tym, co nazywam drugim cyfrowym zwrotem (...). W naszym tradycyjnym sposobie myślenia fakty są mozolnie zbierane, sortowane, porównywane i selekcjonowane, potem generalizowane i formalizowane: kulminacją tego procesu jest teoria, często skompresowana do formy matematycznych wzorów przydatnych do przewidywania przyszłości w porównywalnych sytuacjach. Komputery tego nie robią. (...) Tradycyjni rzemieślnicy, niebędący inżynierami i niekorzystający z matematyki, działali metodą prób i błędów: złoży krzesło, a jeśli się rozpadnie, złoży kolejne i kolejne, aż któreś w końcu wytrzyma. (...) Z tego powodu z czasem zaczęliśmy wierzyć inżynierom bardziej niż rzemieślnikom. Ale dzisiejsze narzędzia komputerowe działają jak rzemieślnicy, nie jak inżynierowie. (...) Tworzenie stało się kwestią czucia, a nie myślenia: po prostu to zrób [wyróżnienie w oryginale] (...). Porzucenie tradycyjnego rozumowania przyczynowego w wielu przypadkach działa nieźle.”
[tłumaczenie własne] (Carpo, 2017, s. 163).

Czy jednak rzeczywiście korzystanie z głębokiego uczenia maszynowego jest „porzuceniem rozumowania przyczynowego”? Yann LeCun, jeden z ojców współczesnego uczenia maszynowego, uważa inaczej. Statystyczny charakter rozwiązań opartych na głębokim uczeniu maszynowym nie wyklucza możliwości, że algorytmy podejmujące interakcję ze światem rzeczywistym, wewnątrznie korzystają z zaawansowanych modeli mechanistycznych wyuczonych na podstawie przeanalizowanych danych. Innymi słowy, o ile próbkowanie danych treningowych dla głębokiego uczenia maszynowego jest statystyczne, to same dane mogą odzwierciedlać deterministyczne zależności rządzące światem (LeCun i Fridman, 2022).

Proste narzędzia potrafiące autonomicznie przetwarzać informacje dostępne są już od lat 50. i 60. dwudziestego wieku, jednak dopiero współczesne głębokie uczenie maszynowe ma szanse na precyzyjne i skuteczne przetwarzanie złożonych i chaotycznych zależności obserwowanych w rzeczywistości. Kryzys algorytmów opartych na zasadach heurystycznych, takich jak systemy eksperckie, wynika w dużej mierze z konieczności zbytniego uproszczenia

¹ SI – Sztuczna Inteligencja (ang. AI – Artificial Intelligence)

opisywanych problemów i rządzących nimi zmiennych. W przeciwnym wypadku opracowane modele byłyby zbyt złożone i niemożliwe do udźwignięcia bez skomplikowanych teorii opisujących ogrom możliwych interakcji między licznymi zmiennymi. Przykładem niech będą modele „mikroświatów”² opracowywanych przez Nicholasa Negroponte w ramach Architecture Machine Group. Uprozczone algorytmy mające w założeniu symulować odizolowane aspekty rzeczywistości, w praktyce generowały stwierdzenia, które odniesione do prawdziwego świata byłyby całkowicie fałszywe (Steenson, 2017). Mimo, że ograniczenia klasycznego programowania w architekturze były znane od dziesięcioleci, do niedawna brakowało dla niego praktycznych alternatyw. Zaledwie od niewiele ponad dekady dysponujemy wystarczającą mocą obliczeniową i dostatecznie niezawodnymi algorytmami, by efektywnie pozwolić maszynom samodzielnie uczyć się rozwiązywania praktycznych problemów (Krizhevsky, i in., 2012).

Gdy mówimy o głębokim uczeniu maszynowym, mamy na myśli uczenie maszynowe wykorzystujące wielowarstwowe sieci neuronowe. Sieci neuronowe wyróżniają się wśród innych technik uczenia maszynowego głównie ze względu na swoją wszechstronność. W 2010 roku Maithani, Arora i Jain podali listę czterech przesłanek uzasadniających możliwość wykorzystania sieci neuronowych w projektowaniu wspomaganym komputerowo:

- Sieci neuronowe mogą rozwiązywać złożone problemy nieliniowe.
- Przyjmują multimodalne dane wejściowe.
- Nie mają uprzedzeń i przewidywań dotyczących dystrybucji danych wejściowych.
- Mogą przetwarzać jednocześnie wiele danych wejściowych i ich pochodnych (Maithani, i in., 2010).

Jak przy każdej nowości wdrażanej w ramach danej dyscypliny powinniśmy być ostrożni przy wprowadzaniu głębokiego uczenia maszynowego w praktykę architektoniczną. Jak pisze

² Ang. Micro-world. Przykładową demonstracją modelu mikroświata sterującego fizyczną cząstką prawdziwego świata był eksperymentalny projekt „SEEK” (nazywany także „Bricks World”). W ramach projektu model cyfrowy miał za zadanie symulować potrzeby myszokoczków zamieszkujących terrarium wypełnione możliwymi do przesuwania bloczkami. Bloczki miały być przenoszone przez robotyczne ramię zgodnie z przewidywaniami algorytmu w celu dostosowania się do potrzeb gryzoni. Eksperyment zakończył się porażką. Myszokoczki zachowywały się niezgodnie z przewidywaniami komputera, niszczyły bloczki i atakowały się nawzajem (Steenson, 2017).

Carpo, ważne, byśmy to my sterowali sztuczną inteligencją, a nie sztuczna inteligencja nami (Carpo, 2017). Algorytmy uczenia maszynowego są często nieprzewidywalne, trudne w interpretacji i ocenie w oderwaniu od wyznaczników wyprowadzonych heurystycznie. Skuteczność współczesnych systemów jest bardzo silnie powiązana z jakością danych treningowych, a same modele są wrażliwe na niespotkane wcześniej przypadki brzegowe i muszą być precyzyjnie dostrajane. Mimo wszystko, sukcesy głębokiego uczenia maszynowego w dziedzinach pokrewnych dają podstawy do optymistycznego patrzenia na możliwości implementacji podobnych algorytmów w architekturze. Trudno nie ulec wrażeniu, że krok w kierunku uczenia maszynowego jest naturalną konsekwencją myślenia o architekturze informacyjnej jako o projektowaniu procesów: z programowaniem hiperparametrycznym zastępującym projektowanie parametryczne. Molly Wright Steenson obrazowo rysuje paralełę między programowaniem algorytmów uczenia maszynowego a architekturą:

„Ci, którzy projektują systemy sztucznej inteligencji nie tworzą rzeczowników - obiektów, budynków, przedmiotów – tworzą czasowniki. Planują architekturę algorytmów uczenia maszynowego – kroki, które program wykona by ukończyć zadanie. Układają na swoje miejsce zbiór warunków początkowych, dzięki którym programy programują się same. Zauważycie, że użyłam słów «projektować» i «architektura»³. To właśnie robią eksperci uczenia maszynowego. Budują modele statystyczne, które ewoluują same i się uczą. Wznoszą ramy algorytmów generatywnych. Tworzą architekturę.” [tłumaczenie własne] (Steenson, 2017, s. 224)

Wydaje się, że podobieństwa między uczeniem maszynowym a architekturą nie sprowadzają się jedynie do gier słownych i współwystępowania tych samych pojęć w obu dziedzinach. Architektoniczna intuicja przestrzenna przydaje się w zrozumieniu wielowymiarowych wektorów i ukrytych hiperprzestrzeni charakterystycznych dla głębokiego uczenia maszynowego. Dzięki wyobraźni przestrzennej skomplikowane wydające się operacje na macierzach można koncepcyjnie zastąpić zestawianiem wielowymiarowych strzałek, a pozornie zawiły⁴ rachunek różniczkowy opisujący propagację wsteczną umożliwiającą sieci

³ W angielskim oryginale – „to architect” w formie czasownikowej.

⁴ W rzeczywistości logika ukryta za rachunkiem różniczkowym propagacji wstecznej jest zaskakująco prosta i intuicyjna, ale stosowana do jej opisu notacja może skutecznie odstraszyć tych nieobytych z matematyką.

neuronowej samodzielne uczenie się, sprowadza się do schodzenia w dół po wielowymiarowych, abstrakcyjnych zboczach funkcji kosztu. Będąc architektem o wiele łatwiej jest zrozumieć przestrzennie koncept separacji liniowej odrębnych klas lub rzutowania wielowymiarowych przestrzeni stanów na powierzchnie dwuwymiarowe przy pomocy narzędzi PCA⁵ czy t-SNE⁶. Według tezy głoszonej przez Bruno Zevi'ego, na najogólniejszym poziomie architektura może być rozumiana jako sztuka kształtowania przestrzeni:

„[Esencja architektury] nie leży w ograniczeniach materialnych nałożonych na wolność przestrzenną, tylko w sposobie, w jaki przestrzeń zorganizowana jest w znaczącą formę poprzez proces ograniczenia.” [tłumaczenie własne] (Zevi, 1993, s. 49)

W projektowaniu hiperparametrycznym architekt nie kształtuje wprawdzie klasycznej, trójwymiarowej przestrzeni euklidesowej, a ogranicza abstrakcyjną, wielowymiarową przestrzeń charakteryzującą badany problem⁷ poprzez definicję hiperparametrów sterujących zachowaniem sieci neuronowej. Znaczenie formie reprezentowanej w tej przestrzeni nadaje algorytm uczenia maszynowego, definiując parametry rozwiązania problemu. W *Źródłach Architektury Informacyjnej* Jan Słyk poszukiwał nietrywialnego wykorzystania geometrii nieeuklidesowych w architekturze. W analizowanych przykładach odnalazł jedynie luźne inspiracje, które w praktyce nie wpływały na rzeczywistą architekturę będącą wynikiem tych inspiracji:

„Pomimo licznych nawiązań i deklaracji, trudno pozbyć się wrażenia, że współczesna architektura korzysta z geometrii nieeuklidesowych w sposób bardzo powierzchowny. (...) Möbius House i Muzeum Mercedesa nie powstałyby prawdopodobnie bez swobody

⁵ PCA – Analiza Głównych Składowych (ang. Principal Component Analysis) – metoda obrotu układu współrzędnych danych wielowymiarowych maksymalizująca wariację tych danych po rzutowaniu na przestrzeń o mniejszej liczbie wymiarów (Pearson, 1901).

⁶ t-SNE – ang. t-distributed Stochastic Neighbor Embedding. Algorytm wizualizacji wielowymiarowych danych w przestrzeni o mniejszej liczbie wymiarów przy próbie zachowania struktury sąsiedztwa danych wejściowych, czyli podobieństwa między punktami, podczas przekształcania danych na niższe wymiary.

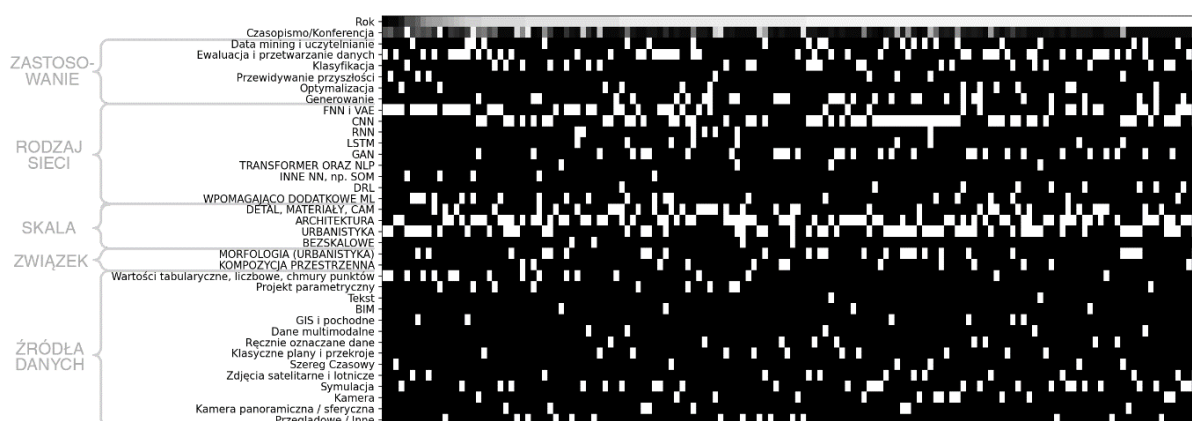
⁷ Technicznie rzecz biorąc hiperprzestrzeń opisująca możliwe stany w głębokim uczeniu maszynowym może spełniać definicję wielowymiarowej przestrzeni euklidesowej. Jednak w potocznym, architektonicznym znaczeniu, mówiąc o przestrzeni euklidesowej, mamy na myśli otaczającą nas klasyczną przestrzeń trójwymiarową.

wyobrażeń, jakie stymuluje współczesna matematyka. (...) Jednak niezależnie od nazw i wyjaśnień, oba te budynki osadzone są w tradycyjnym, pitagorejsko-kartezjańskim kontekście. Dotyczy to zarówno zagadnień projektowych, realizacji, jak i odbioru użytkowego. Konstruuując przedmioty o średnich (w skali doświadczeń fizyki) rozmiarach, podporządkowane prawom ziemskiej grawitacji, działamy w środowisku, które świetnie opisuje geometria euklidesowa. (...) Warto jedynie dostrzec, że środowisko twórczości rozszerza się. (...) Światy przestrzenne Internetu pozbawione są grawitacji. Sterowalne mechanizmy projekcji, hiperłącza i parametryczność ruchu istotnie wpływają na percepcję. W tym kontekście, zdobycze współczesnej matematyki mogą okazać się czynnikiem wpływającym na twórczość szerzej, niż w sensie estetycznej inspiracji.” (Słyk, 2012, s. 58)

W przypadku głębokiego uczenia maszynowego każda jego architektoniczna aplikacja ma u swoich podstaw wielowymiarową kompozycję uporządkowaną w ukrytej hiperprzestrzeni stanów. Hiperprzestrzeń ta jest możliwa do wyobrażenia i zwizualizowania (Olah, i in., 2017), a co najważniejsze, może przetwarzać konteksty przestrzenne, funkcjonalne czy historyczne dotyczące rzeczywistego świata na fizyczną architekturę czy urbanistykę. Ze względu na rozwiniętą wyobraźnię przestrzenną i wysokie kwalifikacje cyfrowe, współcześni architekci coraz częściej sięgają po narzędzia oparte na głębokim uczeniu maszynowym. Architektura wspomagana komputerowo wydaje się stopniowo ewoluować od algorytmów heurystycznych opisanych parametrycznie na niskim poziomie abstrakcji do algorytmów stochastycznych sterowanych na wysokim poziomie abstrakcji przez hiperparametry i odpowiednio sformułowane dane wejściowe (ang. prompt). Przyjrzyjmy się więc tym algorytmom, opartym na nich narzędziom i ich architektonicznym zastosowaniom.

2.2. Sieci neuronowe w literaturze dotyczącej komputerowego wspomagania projektowania

Aby rozpoznać najnowsze zastosowania sieci neuronowych w architekturze i zidentyfikować potencjalne luki badawcze, przeanalizowałem 149 publikacji naukowych z ostatnich kilku lat⁸ dotyczących implementacji sieci neuronowych w projektowaniu wspomaganym komputerowo. Źródła obejmowały przekrój artykułów z czasopism naukowych i publikacji pokonferencyjnych do połowy 2021 roku.



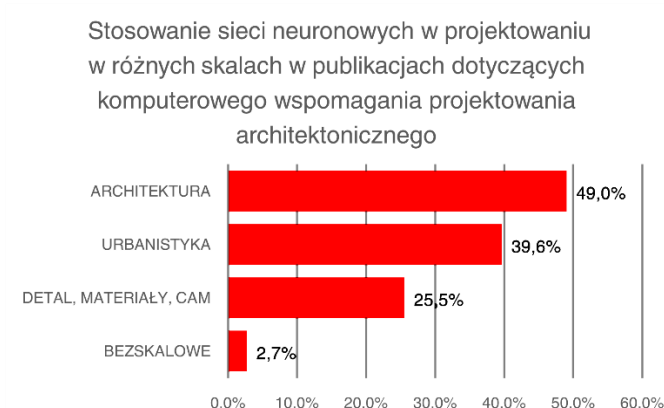
Rys. 1. Wizualizacja 37 cech (oś pionowa) wyodrębnionych ze 149 publikacji (oś pozioma) dotyczących zastosowania sieci neuronowych w projektowaniu wspomaganym komputerowo. Białe kropki oznaczają powiązanie danej cechy z publikacją. Wyodrębnione cechy pozwoliły na organizację zebranych źródeł i przeprowadzenie ich analizy ilościowej.

Opracowanie własne.

W celu ich organizacji źródła zostały zebrane w tabeli umożliwiającej przeprowadzenie analizy jakościowej i ilościowej. Na potrzeby analizy jakościowej w tabeli zapisałem główne wnioski płynące z proponowanych przez autorów implementacji sieci neuronowych w CAAD. Dla celów analizy ilościowej z artykułów wydobyłem 37 cech i zapisałem je metodą kodowania

⁸ Kwerenda zasadnicza została objęta okresem do połowy 2021 roku, w którym to momencie zamknąłem główną część badań literaturowych. W ramach dalszych prac nad doktoratem bibliografia była na bieżąco uzupełniana i aktualizowana o najnowsze wyniki badań i dodatkowe źródła, jednak nie zostały one ujęte w kwerendzie zasadniczej.

„multi-hot”⁹ (dla 35 cech możliwych do przedstawienia binarnego) i w postaci skalarnych wartości znormalizowanych dla pozostałych cech, takich jak rok i wydawnictwo (Rys. 1). Cechy binarne opisujące zebrane źródła obejmowały grupy zastosowań sieci neuronowych (Załącznik 1), rodzaje implementowanych sieci neuronowych, skalę ich zastosowania (Załącznik 2), grupy źródeł danych przetwarzanych przez sieć neuronową, oraz tematykę opracowań (Załącznik 3).



Rys. 2. Stosowanie sieci neuronowych w projektowaniu w różnych skalach w publikacjach dotyczących komputerowego wspomagania projektowania architektonicznego. Opracowanie własne.



Rys. 3. Wizualizacja skal, w których stosowane są sieci neuronowe, wykonana przy pomocy algorytmu t-SNE na podstawie 37 wyodrębnionych cech badanych artykułów. Każdy punkt reprezentuje odrębną publikację, kolor punktu odpowiada skali zastosowania sieci neuronowej. Opracowanie własne.

Po zakodowaniu, zebrane cechy reprezentujące analizowane źródła stanowiły chmurę punktów w 37-wymiarowej przestrzeni (po jednym wymiarze na wyodrębnioną cechę). Badanie ilościowe źródeł polegało na ich analizie statystycznej (Rys. 2) oraz wizualizacji korelacji między nimi przy pomocy algorytmu nienadzorowanego uczenia maszynowego t-SNE,

⁹ Kodowanie multi-hot polega na przypisaniu poszczególnym cechom wartości 0 lub 1 i zestawieniu ich razem w postaci wielowymiarowego wektora. Kodowanie to różni się od metody jeden z wielu (ang. one-hot encoding) tym, że w multi-hot wartość 1 może być przypisana do wielu cech jednocześnie.

umożliwiającego przedstawienie poszczególnych grup publikacji w bardziej czytelnej formie wykresu dwuwymiarowego zachowującego wzajemne podobieństwo między punktami. Na wizualizacji t-SNE główne składowe (osie wykresu) są skorelowane ze skalą zastosowań i typem wykorzystanych w publikacjach sieci neuronowych. Każdy punkt na wykresie odpowiada jednej z publikacji przeanalizowanej w kwerendzie zasadniczej (Rys. 3).

Ze statystycznej analizy przeprowadzonej kwerendy wynika, że w projektowaniu architektonicznym wspomagany komputerowo sieci neuronowe są wykorzystywane we wszystkich skalach projektowych:

- Skali urbanistycznej (39,6% analizowanych publikacji).
- Skali architektonicznej (49,0% analizowanych publikacji).
- Skali detalu, w tym wytwarzania wspomagane komputerowo CAM¹⁰ i materiałów (25,5%).

Niewielka część zebranych źródeł dotyczyła także zastosowań bezskalowych (2,7% analizowanych publikacji).

Na najogólniejszym poziomie wszystkie zastosowania sieci neuronowych można sprowadzić do zadania opracowania funkcji mapującej dane wejściowe na dane wyjściowe w pewien pożądaný sposób. Na trochę bardziej szczegółowym poziomie (ale wciąż na tyle ogólnym, aby umożliwić efektywne grupowanie) podzieliłem zastosowania sieci neuronowych w analizowanych skalach projektowych na pięć głównych kategorii:

- Ewaluacja i przetwarzanie danych, polegające na przypisaniu danym wejściowym pewnej wartości oceniającej lub na bezpośrednim przetworzeniu danych z jednej formy na drugą (30,9% analizowanych publikacji).
- Działalność generatywna, polegająca na tworzeniu rozwiązań projektowych lub sugestii projektowych bezpośrednio przez sieć neuronową (22,1% analizowanych publikacji).
- Klasyfikacja, polegająca na podziale danych wejściowych na kategorie lub na ich semantycznej segmentacji (17,4% analizowanych publikacji).
- Ekstrakcja i uczytelnianie, polegające na pozyskiwaniu danych użytecznych z obszernych, nieustrukturyzowanych baz (ang. data mining), lub na przedstawieniu

¹⁰ CAM – ang. Computer-Aided Manufacturing, wytwarzanie wspomagane komputerowo

danych nieczytelnych i nienadających się do dalszego przetwarzania w czytelniejszej formie pośredniej (15,4% analizowanych publikacji).

- Optymalizacja, polegająca na poszukiwaniu najlepszych rozwiązań danego procesu ze względu na zadane kryteria (8,7%).
- Przewidywanie przyszłości na podstawie sytuacji obecnej i historycznie zarejestrowanych procesów (5,4%).

W niektórych przypadkach brzegowych zastosowanie sieci neuronowej może wpisywać się w kilka kategorii jednocześnie. Przykładowo, czasem trudno jest jednoznacznie rozróżnić, czy dana operacja jest przetworzeniem danych (bezpośrednim) czy ich uczytelnieniem (pośrednim). Dodatkowo, zależnie od kontekstu i poziomu analizy poszczególne grupy zastosowań można rozumieć wieloznacznie. Na przykład patrząc na sieć neuronową z perspektywy technicznej, każdy model będzie służył optymalizacji funkcji kosztu względem wag i progów połączeń między neuronami. Z tych powodów zdecydowałem się dokonać kategoryzacji zastosowań bez precyzyjnego rozgraniczania, skupiając się na zastosowaniach rozumianych z perspektywy projektanta (architekta lub urbanisty) i nie traktując przyjętych kategorii w sposób rozłączny.

2.2.1. Skala urbanistyczna – zastosowania sieci neuronowych

Urbanistyka: ekstrakcja i uczytelnianie danych

W skali urbanistycznej najpopularniejszym w badanym okresie zastosowaniem sieci neuronowych jest ekstrakcja i uczytelnianie danych. Jednym z łatwiej dostępnych źródeł danych urbanistycznych są zdjęcia satelitarne, które ze względu na swoją obszerność i szczegółowość reprezentują ogromną liczbę potencjalnie użytecznych informacji. Ich ręczna ekstrakcja często jest jednak zadaniem bardzo czasochłonnym, lub, zależnie od zakresu, wręcz niemożliwym. Przykładowo, przy pomocy sieci neuronowych można automatycznie dzielić zdjęcia satelitarne miast na interesujące nas sektory. Zespół badaczy z politechniki w Singapurze wykorzystał otwarte, darmowe zdjęcia satelitarne systemu Sentinel 2 jako bazę danych umożliwiającą rozpoznawanie poszczególnych stref wykorzystania terenu w okolicach amerykańskich lotnisk. Tak podzielone zdjęcia satelitarne mogą być przydatne w monitorowaniu zmian w sposobie użytkowania gruntów, strat wywołanych pożarami, czy postępów rekultywacji gruntów (Meeran i Joyce, 2020). Próby wykorzystania sieci neuronowych do identyfikacji zmian w zabudowie miejskiej są podejmowane już od wczesnych lat dwutysięcznych. Liu i Lathrop w 2002 roku zaprogramowali prostą sieć neuronową

rozpoznającą nowozurbanizowane tereny na podstawie kilkunastu wybranych cech wyodrębnionych ze zdjęć satelitarnych. Nawet prosta, dwuwarstwowa sieć neuronowa operująca na ograniczonej liczbie cech osiągała wyższą skuteczność niż wcześniej stosowane metody wykrywania różnic w parach zdjęć satelitarnych (Liu i Lathrop Jr, 2002). Idąc krok dalej, Shota Iino i in. opracowali neuronowy system detekcji krótkoterminowych zmian w tkance miejskiej. System porównuje ze sobą zdjęcia satelitarne uzupełnione o numeryczne modele pokrycia terenu (ang. DSM – Digital Surface Model) w interwałach rocznych i ocenia prawdopodobieństwo, że na danym fragmencie obszaru użytkowanie gruntów uległo zmianie (Iino, i in., 2018). Głębokie uczenie maszynowe aplikowane jest także do uczytelniania zdjęć satelitarnych do postaci diagramatycznych schwarzplanów. Naukowcy z uniwersytetu w Osace zaproponowali metodę trenowania sieci neuronowych rysujących obrysy zabudowy przy pomocy syntetycznych par widok-schwarzplan renderowanych z fotorealistycznych, trójwymiarowych modeli miast. Takie rozszerzenie bazy danych treningowych (ang. data augmentation) pozwala niskim kosztem zwiększyć skuteczność trenowanych algorytmów (Ikeno, i in., 2020). Wychodząc z założenia, że jakość przestrzeni zbudowanej w mieście jest skorelowana ze stanem zdrowia mieszkańców, można zestawiać zdjęcia satelitarne miast z dostępnymi danymi medycznymi, w celu rozpoznania typów tkanki miejskiej powiązanych z wybranymi zaburzeniami zdrowotnymi. Sama korelacja cech nie oznacza, że są one powiązane przyczynowo-skutkowo, obie mogą być niezależnym objawem współwystępujących przyczyn. Korelację tę można jednak wykorzystać do mierzenia cechy łatwiejszej do przetworzenia maszynowego (zdjęcia satelitarne przestrzeni miejskiej) w celu identyfikacji potencjalnego występowania cechy trudniejszej do zmierzenia (problemy zdrowotne mieszkańców). Pewne sukcesy na tym polu wraz z zespołem osiągnął David Newton w zakresie przewidywania wskaźnika otyłości na badanym obszarze (Newton, i in., 2020), analizowania poszczególnych cech tkanki miejskiej skorelowanych z konkretnymi problemami zdrowotnymi (Newton, 2021) czy generowania widoków satelitarnych nieistniejących fragmentów miast w celu zwizualizowania, jakie typy krajobrazów miejskich są skorelowane z występowaniem zaburzeń nerwowych (Newton, 2020). Sieci neuronowe mogą być także użyte do generowania podziałów na grupy typologiczne na podstawie zdjęć satelitarnych. Przy pomocy sieci neuronowych typu autoenkoder można przetworzyć wiele przykładów interesujących nas obszarów (np. kampusów uniwersyteckich) na skompresowaną formę wektorową. Korzystając z takiej reprezentacji zabudowy Peiwen Li i Wenbo Zhu dokonali grupowania morfologicznego kampusów opartego na podobieństwie między poszczególnymi

wektorami przy pomocy algorytmu centroidów (ang. k-means clustering) i klastrowania hierarchicznego [ang. hierarchical clustering (Li i Zhu, 2020)].

Pokrewnym do zdjęć satelitarnych źródłem danych są zdjęcia z samolotów i dronów bezzałogowych. Głębokie uczenie maszynowe pozwala na ekstrakcję trajektorii ruchu pieszego z materiału wideo zarejestrowanego przez dron. Trajektorie poruszania się pieszych mogą być później wykorzystane przy projektowaniu i przebudowie ulic i placów miejskich (Wu, 2021). Z samych trajektorii (np. uzyskanych bezpośrednio z systemów śledzenia GPS) także da się uzyskać dodatkowe dane przydatne w dalszych analizach. Katarzyna Siła-Nowicka i in. proponują wykorzystanie sieci neuronowych do przetwarzania trajektorii GPS na wzorce aktywności miejskiej dotyczącej konkretnych budynków lub ogólne wzorce ruchu w mieście dotyczące poszczególnych trybów transportu (Siła-Nowicka, i in., 2016). Qayyum i in. zastosowali sieci neuronowe do ekstrakcji cech z wejściowych zdjęć wykonanych przez drona. Ekstrakcja istotnych cech opierała się na redukcji przez sieć wymiarów danych wejściowych i zdecydowanie poprawiała skuteczność późniejszej klasyfikacji elementów znajdujących się na satelitarnych zdjęciach miast (Qayyum, i in., 2019). Od momentu wprowadzenia przez Google usługi Street View, urbaniści zaczęli zwracać uwagę na łatwo dostępne, sferyczne zdjęcia przestrzeni publicznych jako na potencjalne źródło informacji miejskich. Dzięki sieciom neuronowym zdjęcia sferyczne pozyskane ze Street View mogą posłużyć do automatycznej ekstrakcji kolorów elewacji na danym obszarze (Zhang, i in., 2020) lub nawet wydobycia *genius loci* badanej ulicy (łac. duch miejsca). Zaproponowany przez Kyle’a Steinfelda system GAN Loci pobiera z serwerów Street View kilkaset zdjęć sferycznych znajdujących się w bezpośredniej okolicy badanego punktu i wykorzystuje je do wytrenowania przeciwstawnej sieci neuronowej. Sieć następnie generuje nieistniejący, fantastyczny krajobraz ilustrujący *genius loci* danego miejsca (Rys. 4) i zawierający elementy krajobrazowe charakterystyczne dla danej okolicy (Steinfeld, 2019).



Rys. 4. Nieistniejący, syntetyczny krajobraz wygenerowany przez opracowany przez Kyle'a Steinfelda neuronowy algorytm GAN Loci wizualizujący „genius loci” dzielnicy Blijdorp w Rotterdamie (Steinfeld, 2019).

Zdjęcia sferyczne Street View, mimo satysfakcjonującego pokrycia większości dużych miast, są aktualizowane bardzo rzadko, co kilka lat, przez co zawarte w nich informacje są zazwyczaj nieaktualne. Receptą na ten problem może być zbieranie materiału filmowego poprzez umieszczenie kamer wideo na pojazdach służb miejskich, takich jak śmieciarki, służby parkingowe czy straż miejską. Zebrane w ten sposób dane są zbyt obszerne do analizy manualnej, jednak mogą być przetwarzane przez sieci neuronowe. Analiza taka umożliwia przykładowo rozpoznawanie problematycznych obiektów, jak porzucone śmieci czy uszkodzone pojazdy, i nanoszenie ich na mapę w celu efektywniejszego zarządzania miastem (Sukel, i in., 2020a; Sukel, i in., 2020b). System taki może być także rozszerzony o dodatkowe, multimodalne źródła danych, jak raporty mieszkańców, dane geolokalizacyjne i historyczne szeregi czasowe w celu dokładniejszego rozpoznawania „mikrowydarzeń” w przestrzeni miejskiej (Sukel, i in., 2019). W temacie danych multimodalnych, głębokie sieci neuronowe sparowane z dodatkowymi technikami uczenia maszynowego (jak PCE czy t-SNE) mogą być wykorzystane do uczytelnienia wielowymiarowych i zróżnicowanych informacji pochodzących z wielu źródeł miejskich w formie możliwej do immersyjnego zwizualizowania w środowisku wirtualnej rzeczywistości (Kampani i Varoudis, 2020). Wartościowe informacje o przestrzeni publicznej da się pozyskać także z języka naturalnego zapisanego w formie tekstowej w mediach społecznościowych. Maszynowa analiza wydźwięku emocjonalnego wypowiedzi (ang. sentiment analysis) pozwala na zmapowanie emocji mieszkańców

na przestrzeni miejskiej w celu uczytelnienia emocjonalnych kontekstów przestrzennych (Kim i Rosenwasser, 2020).

Urbanistyka: ewaluacja i przetwarzanie danych

Drugim najpopularniejszym zastosowaniem głębokiego uczenia maszynowego w skali miejskiej jest ewaluacja danych wejściowych lub ich bezpośrednie przetworzenie na pożądaną formę docelową. W procesach urbanistycznych często przydatne jest przypisanie badanym zjawiskom pewnej wartości numerycznej w celu przeprowadzenia ich analizy ilościowej zarówno na potrzeby projektowania urbanistycznego, jak i zarządzania miastami.

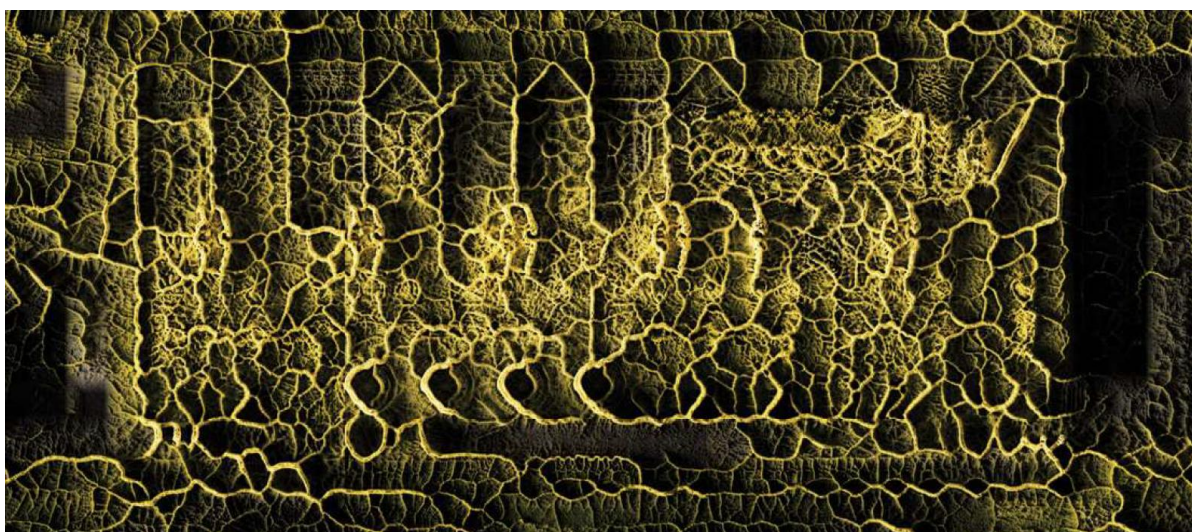
Przydatnym źródłem danych umożliwiającym ewaluację miejskich ulic jest baza sferycznych zdjęć Street View firmy Google. Yazıcıoğlu i Dino zaproponowali metodę semantycznej segmentacji zdjęć sferycznych wzdłuż ulicy i późniejszego przetworzenia otrzymanych diagramów na przewidywany poziom komfortu termicznego w danym punkcie. Dane treningowe dla sieci neuronowych zostały wygenerowane przy pomocy komputerowej symulacji nasłonecznienia przeprowadzonej na urbanistycznym modelu 3D (Yazıcıoğlu i Dino, 2021). Qi i in. opracowali system przetwarzający zdjęcia Street View na liczbowy wynik „witalności” danego miejsca. Sieć neuronowa została wytrenowana na podstawie zbioru ręcznie ocenianych względem witalności lokalizacji badanych metodą wizji lokalnej. Tak zebrane dane pozwoliły wytrenować sieć neuronową oceniającą witalność dowolnego fragmentu miasta. Co więcej, na podstawie analizy reprezentacji danego zdjęcia w przestrzeni ukrytej ostatniej warstwy sieci neuronowej, autorzy byli w stanie zidentyfikować dwanaście klas mających największy pozytywny wpływ na ocenę witalności miejsca. Według sieci najsilniej na witalność obszaru wpływała obecność na zdjęciach przechodniów, witryn sklepowych, chodników oraz, co zaskakujące, placów budowy (Qi, i in., 2020). Podobną metodę oceny „preferencyjności” przestrzeni zaproponowali Kinugawa i Takizawa z Uniwersytetu w Osace. W tym przypadku zdjęcia sferyczne uzupełnione były o maszynowo generowaną mapę głębi obrazu (wspomagana przez segmentację semantyczną w celu dokładniejszego rozpoznawania nieba) i następnie poddano je przetworzeniu na metrykę preferencyjności danego miejsca. Dane do treningu zebrano dzięki ankiecie przeprowadzonej na studentach architektury oceniających treningowe zdjęcia sferyczne na podstawie ich projekcji w rzeczywistości wirtualnej (Kinugawa i Takizawa, 2019). Pokrewny, dawniejszy system został zaprezentowany przez Takizawę i Furutę dwa lata wcześniej, jednak zamiast rzeczywistych przestrzeni miejskich dotyczył ewaluacji wirtualnych zdjęć sferycznych

renderowanych w 3D przez silnik Unity (Takizawa i Furuta, 2017). Możliwość przeprowadzania przy pomocy sieci neuronowych ewaluacji niezrealizowanych projektów syntetycznych jest szczególnie przydatna przy projektowaniu generatywnym i parametrycznym. Przy tej kategorii problemów projektant często musi dokonać wyboru najlepszych propozycji z obszernego zbioru możliwych rozwiązań. Sieci neuronowe mogą szybko ocenić wygenerowane projekty pod interesującymi projektanta aspektami, umożliwiając ich dalszą selekcję. Przykładem takiego zastosowania jest system generujący kwartały urbanistyczne opracowany przez Z. Hana, W. Yana i G. Liu. Autorzy zastosowali głębokie uczenie ze wzmocnieniem (ang. DRL: Deep Reinforcement Learning), w którym sterowany przez sieć neuronową agent generował w danej lokalizacji kubaturowy projekt zagospodarowania kwartału i jednocześnie oceniał wygenerowane rozwiązania pod względem estetyki i nasłonecznienia. Dzięki przypisanym metrykom możliwa była selekcja najlepszego projektu spośród dwóch tysięcy wygenerowanych możliwości (Han, i in., 2020).

Ewaluacji i przetwarzaniu można poddać także zdjęcia lotnicze i satelitarne. Za przykład może posłużyć technika ewaluacji stanu technicznego dróg opisana przez chińskich naukowców w 2020 roku. Pan i in. wykorzystali zdjęcia z autonomicznego drona (ang. UAV: Unmanned Aerial Vehicle) w celu identyfikacji zarysowań i dziur w drogach oraz oceny stanu uszkodzeń powierzchni asfaltowych – od niewielkich uszkodzeń we wczesnym stadium po poważnie uszkodzone powierzchnie wymagające interwencji (Pan, i in., 2021). Nie wszystkie zastosowania sieci neuronowych są jednak tak pragmatyczne. Zdarzają się aplikacje mniej jednoznaczne, o głębszym wydźwięku artystycznym. Claudia Pasquero i Marco Poletto z Innsbruck University i ecoLogicStudio opracowali algorytm GAN_Physarum umożliwiający, według autorów, transfer inteligentnych cech biologicznych zaczerpniętych z kolonii bezkomórkowego śluzowca do tkanki miejskiej lub projektu architektonicznego. Zaproponowany proces „urbanistycznej remetalizacji” jest *de facto* neuronowym transferem stylu (Gatys, i in., 2016) wytrenowanym na próbkach *Physarum polycephalum*: nieposiadającego układu nerwowego śluzowca wykazującego zdolności do rozwiązywania zaskakująco złożonych problemów (także obliczeniowych) i zdolnego do samoorganizacji. Wygenerowane przez GAN_Physarum abstrakcyjne, wieloskalowe wariacje na temat urbanistyki (Rys. 5) i architektury (Rys. 6) skłaniają do refleksji nad podobieństwami między przejawami inteligencji ekologicznej a algorytmiczną naturą współczesnych założeń miejskich (Pasquero i Poletto, 2020).



Rys. 5. Algorytm GAN_Physarum (oparty na neuronowym transferze stylu) opracowany przez Pasquero i Poletto zastosowany do przetworzenia satelitarnych zdjęć tkanki urbanistycznej w skali 1 x 1 km (po lewej) i 10 x 10 km (po prawej) (Pasquero i Poletto, 2020).



Rys. 6. Algorytm GAN_Physarum (oparty na neuronowym transferze stylu) opracowany przez Pasquero i Poletto zastosowany do przetworzenia Centre Pompidou sfotografowanego z lotu ptaka (Pasquero i Poletto, 2020).

W temacie ogólnie rozumianych rozważań ekologicznych, sieci neuronowe mogą być przydatne w analizowaniu zależności między morfologią przestrzeni zbudowanej i zieleni miejskiej, a aktywnością fizyczną mieszkańców na świeżym powietrzu. Zestawiając ze sobą

dwie bazy danych, GIS-owe mapy miejskie systemu Mapbox oraz mapy aktywności fizycznej, Strava- Yunjuan Sun, Lei Jiang i Hao Zheng wytrenowali sieć neuronową przekształcającą dany na wejściu plan fragmentu miasta na przewidywaną aktywność fizyczną mieszkańców na tym obszarze. System taki może służyć do przewidywania wzorców aktywności na niezagospodarowanych terenach projektowanych, ale także do analizy korelacji między cechami danego obszaru a ich wpływem na zachowanie mieszkańców. Przykładowo, autorzy zauważyli, że zazielenione, kameralne aleje osadzone w krajobrazie o wiele silniej skorelowane są z bieganiem i spacerowaniem niż z jazdą na rowerze, podczas gdy proste aleje bez częstych zakrętów sprzyjają jeździe na rowerze. Przeprowadzając drobne interwencje urbanistyczne na badanym obszarze autorzy sprawdzili także ich wpływ na przewidywane zmiany we wzorcach aktywności wywołane wprowadzonymi zmianami. Z porównania map ciepła sprzed interwencji i po niej, można przykładowo dojść do wniosku, że poszerzenie ulicy zwiększy intensywność ruchu rowerowego, a zwiększenie powierzchni zabudowy obniży intensywność wszystkich form aktywności fizycznej (Sun, i in., 2020).

Większość wymienionych powyżej zastosowań sieci neuronowych jako dane wejściowe wykorzystywała informacje zapisane w formie pewnego obrazu: zdjęcia, diagramu, mapy rastrowej czy wizualizacji 3D. Zasadniczo nie ma jednak ograniczeń co do typu danych przetwarzanych przez sieć neuronową, o ile mogą one być zapisane w postaci numerycznej. Jedno z wcześniejszych przestrzenno-urbanistycznych zastosowań sieci neuronowych z 1993 roku dotyczyło przetwarzania danych tabularycznych. Stan Openshaw z uniwersytetu w Leeds w latach 90. badał różne możliwości wykorzystania uczenia maszynowego i innowacyjnych algorytmów komputerowych w przetwarzaniu danych geograficznych. W trakcie tych poszukiwań przyjrzał się także prostym sieciom neuronowym i podjął próbę modelowania przy ich pomocy interakcji przestrzennych na przykładzie tabeli opisujących wzorce przemieszczania się do pracy i z powrotem mieszkańców miast w północno-wschodniej Anglii. Modelowane dane opisywały poszczególne środki transportu i powiązane z nimi koszty przemieszczania się. Sieci neuronowe radziły sobie z tym zadaniem na poziomie konkurencyjnym do stosowanych wtedy alternatywnych modeli klasycznych i wykazywały zdolność do generalizowania wyuczonych zależności na niespotkane w trakcie treningu sytuacje (Openshaw, 1993). Innym przykładem ewaluacji zmiennej zależnej od zapisanych liczbowo zmiennych niezależnych jest przewidywanie cen nieruchomości na podstawie ręcznie wyodrębnionych cech obiektu. Cechy takie mogą obejmować zmienne mikroekonomiczne opisujące konkretny budynek (powierzchnia, lokalizacja, rodzaj dachu, liczba sypialni,

odległości od usług etc.) oraz zmienne makroekonomiczne opisujące ogólną sytuację gospodarczą kraju (PKB, Climate Change Performance Index, stopa bezrobocia etc.). Zapewniwszy sieci neuronowej odpowiednią liczbę przykładów treningowych można skutecznie przewidywać ceny rynkowe badanej nieruchomości (Lam, i in., 2008; Abidoye i Chan, 2017). Przetwarzane w ramach jednego zastosowania dane nie muszą być także homogeniczne i mogą pochodzić ze zróżnicowanych źródeł. Li i in. opracowali złożony system oceniający indeks „przyjazności ulic miejskich dla ruchu pieszego” (ang. WoUS Index: Walkability of Urban Streets Index) na podstawie wektorowych map GISowych pozyskanych z Open Street Maps i Google Maps, zdjęć sferycznych Street View (Google, 2023), intensywności ruchu samochodowego ocenianego na podstawie wideo z monitoringu i danych środowiskowych zbieranych przez dedykowane sensory. W zaproponowanym systemie poszczególne źródła danych przetwarzane są przez różne, dedykowane sieci neuronowe i dodatkowe algorytmy komputerowe i przeliczane są względem poszczególnych wag na indeks przyjazności dla ruchu pieszego (Li, i in., 2020). Xinghua Gao wraz z Pardisem Pishdad-Bozorgim wykorzystali konglomerat danych wejściowych składających się z cech wydobywanych z modeli BIMowych, dokumentacji projektowych i wykonawczych, danych z systemu zarządzania miejscem pracy (IWMS: ang. Integrated Workplace Management System) oraz wskaźników wykorzystania mediów do ewaluacji kosztów cyklu życia obiektów użyteczności publicznej. Autorzy przetestowali równolegle kilka najpopularniejszych technik uczenia maszynowego. W przypadku głębokich sieci neuronowych zwrócili uwagę na ich potencjał w zdolnościach przybliżania skomplikowanych funkcji zależnych od wielu zróżnicowanych zmiennych (Gao i Pishdad-Bozorgi, 2019) .

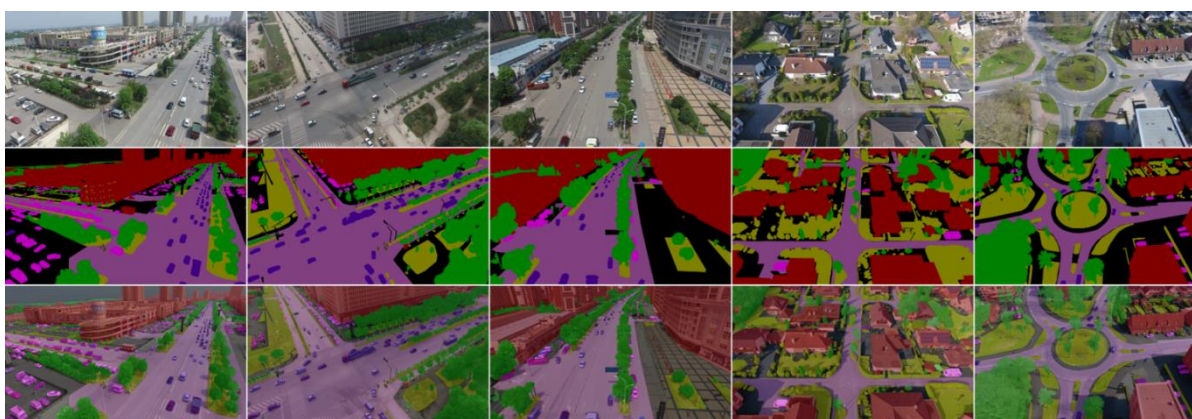
Urbanistyka: klasyfikacja

Kolejną grupą zastosowań sieci neuronowych w urbanistyce jest klasyfikacja, czyli rozpoznawanie w danych wejściowych pewnych klas i ich podział na grupy. Najczęściej klasyfikacji poddaje się dane reprezentowane w formie obrazów: fotografii wykonanych z perspektywy człowieka, zdjęć z monitoringu, diagramów, zdjęć sferycznych czy fotografii lotniczych. Bardzo ciekawym zastosowaniem sieci neuronowych w klasyfikacji jest opracowana przez Briana Ho metoda rozpoznawania elementów kompozycji urbanistycznej z *Obrazu Miasta* Kevina Lyncha (Lynch, 1960). W swojej publikacji pt. *Making a New City Image* Ho zaproponował system klasyfikacji fragmentów tkanki miejskiej oparty na czarno-białych fotografiach wykonanych z perspektywy człowieka uzupełnionych

o diagramatyczny obraz planu otaczającej zabudowy (schwarzplan). Do treningu sieci neuronowej autor wykorzystał oryginalne, oznaczone przez Lyncha fotografie pochodzące z jego roboczego archiwum. Skuteczność algorytmu została oceniona poprzez kolażowe dodawanie kolejnych obiektów na testowanych krajobrazach miejskich – zarówno na poddanemu fotomontażowi zdjęciu, jak i na odpowiadającym mu schwarzplanie. W miarę dodawania budynków do krajobrazu przedstawiającego „niedzielnicę”, algorytm klasyfikacyjny w pewnym momencie zaczyna rozpoznawać obszar jako „dzielnice” (Ho, 2020). Podobny system autorstwa Stephena Lawa i in. klasyfikuje wejściowe zdjęcia sferyczne Street View względem typu sfotografowanych fasad na cztery kategorie: fasady puste (pełne mury, płoty, etc.), aktywne jednostronnie, aktywne dwustronnie oraz zdjęcia bez fasad miejskich. Autorzy nie porzucili jedynie na wytrenowaniu skutecznego algorytmu, ale wykorzystali opracowany system do przeprowadzenia dalszych analiz. Sprawdzono między innymi korelację między aktywnością fasad a ręcznie oznaczanym przez ankietowanych współczynnikiem „sceniczności krajobrazu”. Według wyników na ograniczonej próbie ankietowanych aktywne fasady zdają się sprzyjać „sceniczności” okolicy. Sprawdzono także korelację między aktywnością fasad a cenami poszczególnych nieruchomości. Okazało się, że aktywność fasad w bezpośrednim otoczeniu nieruchomości nie jest istotnym czynnikiem wpływającym na cenę, jednak aktywność fasad mierzona jako średnia dla całego sąsiedztwa jest już czynnikiem silnie skorelowanym z ceną (dzielnice z wyższym wskaźnikiem aktywności fasad zazwyczaj mają też wyższe ceny nieruchomości). Przy pomocy wizualizacji gradientów aktywacji poszczególnych warstw sieci odpowiadających za rozróżnianie poszczególnych klas, autorom udało się także zwizualizować mapy „uwagi”, którą sieć neuronowa skupia na konkretnych obszarach zdjęcia przy podejmowaniu decyzji. Mapy te pozwalają na wizualizację poszczególnych rodzajów fasad bezpośrednio na wejściowych zdjęciach sferycznych (Law, i in., 2018). Dosyć szeroko stosuje się także klasyfikację obiektów widocznych na zdjęciach lotniczych i satelitarnych. Na potrzeby projektowania i zarządzania algorytmy takie mogą być przydatne do automatycznego rozpoznawania drzew, budynków, dróg czy większych struktur trudniejszych do uchwycenia z perspektywy człowieka (Pourazar, i in., 2021).

Szczególnym rodzajem problemów klasyfikacyjnych jest segmentacja semantyczna (ang. semantic segmentation) nazywana także klasyfikacją na poziomie piksela (pixel-level classification) (Long, i in., 2015). Segmentacja semantyczna polega na podziale wejściowego obrazu (lub innych danych wektorowych) na strefy odpowiadające poszczególnym kategoriom

(Rys. 7). Możliwe jest więc przykładowo wyszczególnienie dokładnych obrysów poszczególnych obiektów znajdujących się na zdjęciu. Neuronową klasyfikację na poziomie piksela wykorzystuje między innymi wspomniany już wcześniej zespół z Osaki, kierowany przez Tomohirę Fukudę i Nobuyoshi Yabuki. Fukuda i in. używają segmentacji semantycznej do przetwarzania scen miejskich przedstawionych na fotografiach z perspektywy człowieka oraz zdjęć sferycznych na potrzeby urbanistycznych wizualizacji w rzeczywistości mieszanej (ang. MR: Mixed Reality). Przy pomocy segmentacji neuronowej można generować maski oddzielające pierwszy plan zabudowy od jej tła w celu zwizualizowania modelu przesłanianych obiektów projektowanych (Nakabayashi, i in., 2021), lub maski oddzielające konkretny budynek od otoczenia w celu zautomatyzowanej wizualizacji krajobrazu miejskiego po jego wyburzeniu (Kikuchi, i in., 2021; Fukuda, i in., 2017).



Rys. 7. Przykładowe zdjęcia z drona pobrane z bazy danych UAVid stworzonej przez Lyu, Vosselmana, Xia i in. służącej do trenowania neuronowych algorytmów semantycznej segmentacji. Poszczególne klasy obiektów widoczne na zdjęciu oznaczone są na wizualizacjach różnymi kolorami wyznaczającymi ich obrysy (Lyu, i in., 2018).

Klasyfikacji i segmentacji semantycznej można poddawać także inne niż zdjęcia formaty danych. We wczesnych latach dwutysięcznych Kauko, Hooimeijer i Hakfoort wykorzystali algorytm oparty na sieci neuronowej typu SOM (ang. Self-Organizing Map) do przeprowadzenia klasyfikacji domów na sprzedaż względem „podrynków” rynku nieruchomości w Helsinkach. Przez „podrynki” autorzy rozumieli grupy zbliżonych cenowo i jakościowo nieruchomości oddzielonych od innych grup gwałtownymi, nieliniowymi rozbieżnościami w stosunku ceny do jakości. Opracowany algorytm dokonywał klasyfikacji na podstawie dziesięciu wyodrębnionych zmiennych niezależnych opisujących daną nieruchomość. Analiza procesu klasyfikacji przeprowadzanej przez sieć neuronową pozwoliła

autorom na identyfikację tych zmiennych, które miały kluczowy wpływ na segmentację rynku nieruchomości. Kluczowymi zmiennymi dla kontekstu Helsinek okazały się lokalizacja i rodzaj nieruchomości. Paradoksalnie sama cena miała na podział rynku o wiele mniejszy wpływ (Kauko, i in., 2002). Bardziej współczesnym przykładem wykorzystania danych multimodalnych w klasyfikacji jest metoda strefowania miasta według obszarów funkcjonalnych, opracowana przez naukowców z uniwersytetu w Wuhan. Zróżnicowane dane opisujące badany obszar, obejmujące odległości od centrów handlowych, pochylenie terenu, zagęszczenie placówek medycznych, odległości od dróg, etc., zostały przedstawione w formie gradientowych map pokrywających badany teren. Wytrenowana na tych danych sieć neuronowa potrafiła skutecznie dzielić miasto na strefy funkcjonalne i, co więcej, mogła być przydatna przy symulowaniu zmian w strefowaniu wywołanych przez zmiany wprowadzone w tkance miejskiej opisywanej przez dane wejściowe (Zhai, i in., 2020).

Urbanistyka: przewidywanie przyszłości

Jednym z ciekawszych zastosowań sieci neuronowych są próby przewidywania przyszłości w relacji do zjawisk cyklicznych i procesów ciągłych. Na podstawie danych historycznych oraz informacji o stanie obecnym, w teorii, możliwe jest formułowanie prognoz o przewidywanych zmianach urbanistycznych w różnych przedziałach czasowych. W jednym z wcześniejszych omawianych tu zastosowań sieci neuronowych w modelowaniu zjawisk w przestrzeni geograficznej, Manfred Fischer i Sucharita Gopal zaproponowali metodę przewidywania przyszłego obciążenia sieci telekomunikacyjnej w Austrii opartą na historycznych danych dotyczących wykorzystania sieci. Mimo że opracowanie to nie miało charakteru *stricte* urbanistycznego, to na podstawie wyników autorzy już w 1994 roku zauważyli potencjał drzemiący w sieciach neuronowych i przewidzieli wpływ jaki wywrze głębokie uczenie maszynowe na przetwarzanie złożonych danych przestrzennych:

„Wyniki analiz na chaotycznych, rzeczywistych danych telekomunikacyjnych o ograniczonej długości zapisu wykazały wyższość modeli opartych na sieciach neuronowych nad klasycznymi metodami regresji funkcji. (...) Sieci neuronowe odegrają ważną rolę w geografii i badaniach regionalnych nie tylko w modelowaniu interakcji przestrzennych, lecz także w sferze rozpoznawczych analiz przestrzennych. Co do zasady sieci neuronowe dobrze radzą sobie z problemami reprezentowanymi przez obszerne zbiory danych wybrakowanych, niedokładnych i opisujących niejednoznaczne

zjawiska, które trudno modelować przy pomocy konwencjonalnych technik statystycznych.” [tłumaczenie własne] (Fischer i Gopal, 1994, s. 21)

W 2004 roku Anthony Gar-On Yeh i Li Xia na przykładzie gwałtownie rozrastającego się miasta Dongguan w Chinach próbowali przewidywać prawdopodobieństwo rozbudowy miasta na testowanych obszarach. Na podstawie siedmiu zmiennych wyodrębnionych ze zdjęć satelitarnych i systemów GIS-owych prosta, trójwarstwowa sieć neuronowa wytrenowana na przykładach z dwóch momentów w czasie potrafiła według autorów generować wiarygodne propozycje przy założeniu, że sieć transportu w mieście nie ulegnie zmianie (Yeh i Xia, 2004). Przy zastosowaniu podobnej metodyki i porównywalnych danych wejściowych (cztery cechy wyodrębnione na podstawie zdjęć satelitarnych systemu LISS III) w 2010 roku Maithani i in. opracowali metodę strefowania miasta względem potencjału rozrostu urbanistycznego. Metoda została przetestowana na indyjskim mieście Dehradun w celu prognozowania jego przyszłej rozbudowy (Maithani, i in., 2010).

Obecnie do przewidywania przyszłości przy pomocy głębokiego uczenia maszynowego najczęściej wykorzystuje się szeregi czasowe, czyli zbiory obserwacji uporządkowane według czasu ich uzyskania (PWN, 2022). Boulila i in. wykorzystali segmentację semantyczną miejskich zdjęć satelitarnych do stworzenia obrazów przedstawiających same tereny zabudowane ulokowane na czarnym tle. Otrzymane z różnych okresów mapy zostały ustawione w szeregi czasowe, na podstawie których wytrenowano sieć neuronową przewidującą zagęszczanie się zabudowy w najbliższej przyszłości. Skuteczność rozwiązania przetestowano na trzech różnych miastach w Arabii Saudyjskiej. Opracowany system może służyć identyfikacji obszarów o dużym ryzyku niekontrolowanego rozrostu zabudowy, co z kolei może być przydatne w administracyjnym ograniczaniu rozlewania się tkanki miejskiej (Boulila, i in., 2021). Ze względu na swoją okresowość w cyklach dziennych, tygodniowych i rocznych ruch samochodowy w miastach wydaje się dobrym kandydatem do modelowania przez sieci neuronowe. Przy pomocy głębokiego uczenia maszynowego zatory drogowe w mieście można skutecznie przewidywać na podstawie szeregów czasowych kolorystycznych diagramów odpowiadających natężeniu ruchu samochodowego w mieście (Ranjan, i in., 2020) lub na podstawie szeregów czasowych zbioru trajektorii GPS przedstawionych w postaci wektorowej (Guo, i in., 2019). Prognozowanie natężenia ruchu samochodowego może być przydatne dla operatorów systemów komunikacyjnych, zarządców i projektantów miast oraz samych mieszkańców. Przy pomocy głębokiego uczenia maszynowego można analizować także ruch

pieszy. W tym przypadku dużym problemem jest niska dostępność danych pomiarowych umożliwiających wytrenowanie algorytmów uczenia maszynowego. Nikol Kirova i Areti Markopoulou z IAAC¹¹ proponują uzbrojenie traktów pieszych w technologię „inteligentnych chodników” miejskich opartych na materiałach sensorycznych. Chodniki takie mogłyby bardzo precyzyjnie mierzyć lokalne natężenie ruchu pieszego. Oprócz prezentacji niewielkiego prototypu inteligentnego chodnika autorzy przeprowadzili także komputerową symulację wielkoskalowego systemu. Na podstawie symulowanego ruchu pieszego wytrenowano też sieć neuronową zdolną klasyfikować bieżące, indywidualne trasy ruchu poszczególnych pieszych i przewidywać przepływy ruchu pieszego w niedalekiej przyszłości (Kirova i Markopoulou, 2020).

Urbanistyka: generowanie rozwiązań projektowych

W przypadku neuronowych algorytmów generatywnych głównym problemem jest ujęcie zadania projektowego w ramy umożliwiające efektywne wykorzystanie głębokiego uczenia maszynowego. Konieczne jest odpowiednie sformułowanie treningowych danych wejściowych i wyjściowych tak, by uchwycić istotę funkcji twórczej aproksymowanej przez sieć neuronową. Jedną z większych przeszkód jest praktyczna możliwość zapewnienia algorytmowi wystarczającej liczby przykładów treningowych.

Najprostszą metodą wydaje się wykorzystanie zebranych automatycznie kontekstów istniejących i próba nauczania sieci odtwarzania wzorcowych rozwiązań w nowych sytuacjach. Przykładem takiego podejścia jest algorytm zaprojektowany przez naukowców z Uniwersytetu Jiaotong w Pekinie. System ten służy generowaniu propozycji obrysów zabudowy w zadanym kwartale. Zastosowaną do tego celu sieć neuronową wytrenowano na 167 przykładach schwarzplanów już zrealizowanych kwartałów znajdujących się w sąsiedztwie obszaru projektowego (Pan, i in., 2021). Podobny system, ale rozwinięty o możliwość przetwarzania oprócz schwarzplanów także funkcji projektowanej zabudowy, zaproponował Runjia Tian z Harvardzkiej Szkoły Designu. Do treningu autor użył istniejących przykładów zabudowy w Bostonie i stanie Massachusetts pobranych z otwartych baz danych GIS. Opracowana metoda umożliwia przeprowadzenie procesu „transferu miejskości”, czyli generowania propozycji zabudowy wzorowanej na stylu jednego kontekstu miejskiego w innym kontekście. Tian

¹¹ IAAC – Institute for Advanced Architecture of Catalonia

zaprezentował ten proces generując przykład zabudowy na Manhattanie w stylu zabudowy bostońskiej (Tian, 2020). Najistotniejszym problemem powyższych metod jest ograniczona kontrola nad cechami wygenerowanych rozwiązań. Przy treningu sieci projektant zmuszony jest korzystać z już istniejących i zrealizowanych obiektów, które niekoniecznie mają pożądaną charakterystykę. Kontrola nad zasadami projektowymi sprowadza się do selekcji wzorcowych rozwiązań służących sieci neuronowej za inspirację. Sformalizowanym przykładem takiej kontroli nad projektem jest metoda selekcji danych treningowych wykorzystana przez naukowców z uniwersytetu w Kantonie. Przy problemie generowania diagramów funkcjonalnych i obrysów zabudowy kampusów uniwersyteckich i szkół podstawowych selekcja przykładów istniejących oparta była na filtrach zbudowanych z czterech (w przypadku kampusów) lub siedmiu (w przypadku szkół podstawowych) warunków, ograniczających liczbę wzorców treningowych tylko do tych, które spełniały pożądane cechy projektowe i zrealizowane były w podobnych kontekstach. Przykładowo, brane były pod uwagę jedynie szkoły podstawowe z wyodrębnionym placem zabaw, znajdujące się w konkretnej strefie klimatycznej, zbudowane na płaskim terenie, etc... Według autorów ograniczenie liczby przykładów treningowych tylko do tych spełniających założone warunki pozwoliło na osiągnięcie lepszych i bardziej precyzyjnych rezultatów (Liu, i in., 2020).

Alternatywą do korzystania przy treningu z założeń istniejących jest generowanie przykładów treningowych przy pomocy algorytmów heurystycznych lub trenowanie sieci w symulacji metodą głębokiego uczenia ze wzmacnianiem. System stworzony przez Nirvika Sałę, Johna Haymakera i Dennisa Sheldena opublikowany w formie wtyczki do środowiska Grasshopper, wykorzystuje głębokie uczenie maszynowe do sterowania algorytmem generującym podziały przestrzenne w wielu skalach (ang. SAP – Space Allocation Problem). Zamiast uczyć się na zadanych przykładach, system stara się generować strefowanie przestrzeni maksymalizujące poziom spełnienia matematycznie sformułowanych wymagań projektowych (Saha, i in., 2020). W podobny sposób działa algorytm autorstwa Yanana Songa i Philipa F. Yuana generujący formę zabudowy kwartału. W tym przypadku głębokie uczenie maszynowe modyfikuje formę zabudowy w środowisku symulacyjnym oceniającym efektywność modelu względem podmuchów wiatru wywołanych przez geometrię kwartału (Song i Yuan, 2021).

W przypadku, gdy nie jest możliwa selekcja wystarczającej liczby przykładów treningowych na podstawie rozwiązań istniejących, ani nie da się wytrenować sieci na podstawie symulacji, rozwiązaniem może być ręczne lub półautomatyczne przygotowanie danych treningowych

odzwierciedlających pożądaną funkcję projektową. De Yu z Kolegium Uniwersyteckiego w Londynie opracował system generujący proponowane ramy funkcjonalne kwartałów urbanistycznych podporządkowanych idei ogrodnictwa miejskiego. Do wytrenowania sieci neuronowej będącej osią procesu, Yu wykorzystał ręcznie przygotowane, reprezentatywne przykłady trójwymiarowych podziałów funkcjonalnych pasujących do konkretnych kontekstów przedstawionych w formie rzutów i przekrojów (Yu, 2020). Podejście takie zapewnia pełną kontrolę nad przykładami treningowymi, jednak wymaga dużego nakładu pracy.

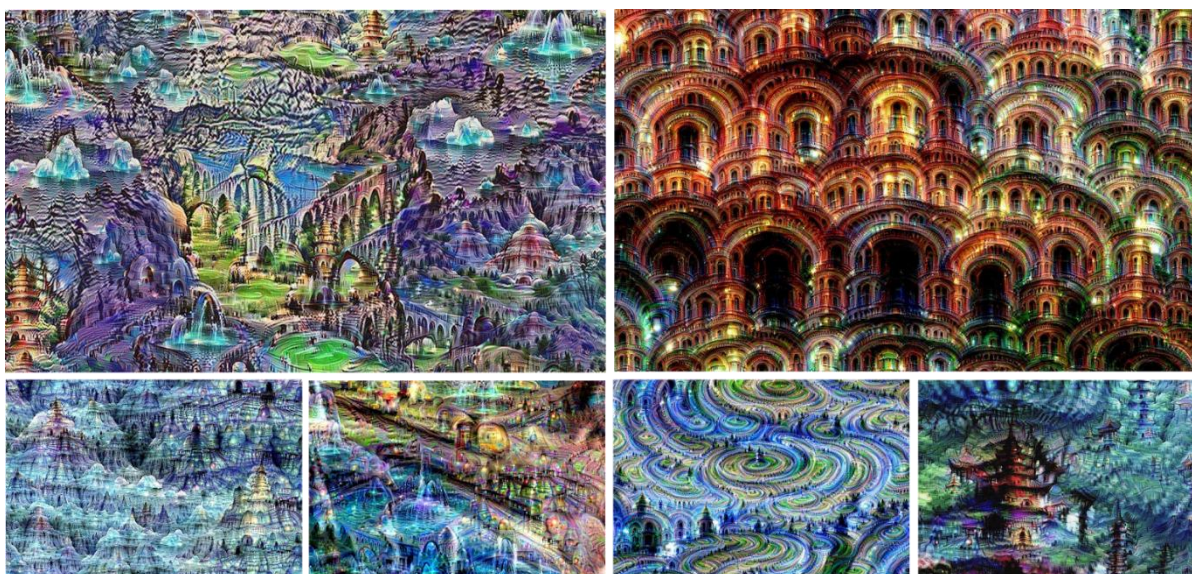
Urbanistyka: optymalizacja

W zidentyfikowanych źródłach bezpośrednia optymalizacja jest najrzadszym z zastosowań sieci neuronowych w urbanistyce. Należy jednak zastrzec, że, jak wspomniano na początku tego rozdziału, praktycznie każdy proces treningu sieci neuronowych to w technicznym znaczeniu proces optymalizacyjny, którego celem jest minimalizacja funkcji kosztu. Podobnie, przy zastosowaniu głębokiego uczenia ze wzmacnianiem, bezpośrednim celem algorytmu jest maksymalizacja wyniku obliczanego na podstawie interakcji sterowanego przez sieć neuronową agenta ze środowiskiem symulacyjnym. W tym podrozdziale nie chodzi jednak o wewnętrzną, pośrednią optymalizację parametrów sieci neuronowych, tylko o wykorzystanie algorytmów głębokiego uczenia maszynowego w docelowej optymalizacji procesów lub zjawisk urbanistycznych. W tym znaczeniu optymalizacji można poddać projekt zagospodarowania kwartału, starając się metodą głębokiego uczenia ze wzmacnianiem maksymalizować estetykę i nasłonecznienie. Metodę taką zaproponowali w swojej pracy Han, Yan i Liu (Han, i in., 2020). Innym przykładem jest opracowana przez naukowców z uniwersytetu w Alicante w Hiszpanii neuronowa metoda relaksacji i uproszczenia siatki przestrzennej, maksymalizującej odległości między węzłami przy jednoczesnym zachowaniu założonych ograniczeń projektowych i obrysu obszaru. W projektowaniu urbanistycznym system taki może być wykorzystany w planowaniu wolnostojącej zabudowy jednorodzinnej w sposób zachowujący jednorodność gęstości zabudowy na całym obszarze. Przy węzłowych założeniach już istniejących, nałożenie na nie zoptymalizowanej siatki o określonej liczbie węzłów pozwala na przeprowadzenie różnorodnych działań urbanistycznych równomiernie obsługujących wszystkie węzły siatki pierwotnej. Przykładowo, możliwe jest pokrycie dzielnicy bezprzewodową siecią nadajników telekomunikacyjnych zapewniającą jednorodny zasięg wszystkim mieszkańcom (Oliver, i in., 2011).

2.2.2. Skala architektoniczna – zastosowania sieci neuronowych

Architektura: generowanie rozwiązań projektowych

W skali architektonicznej sieci neuronowe najczęściej wykorzystywane są jako algorytmy generatywne. Potencjał sieci neuronowych do radzenia sobie z zadaniami twórczymi został zauważony przez architektów w okolicach 2015 roku, gdy opracowany przez Google algorytm DeepDream uświadomił światu, że sieci neuronowe stosowane w klasyfikacji obrazów tworzą złożone, wewnętrzne „wyobrażenia” na temat wyuczonych klas. Wyobrażenia te można zwizualizować poprzez iteracyjne wzmacnianie na danym obrazie cech, które zwiększają pewność przewidywania wybranej warstwy sieci (Mordvintsev, i in., 2015).

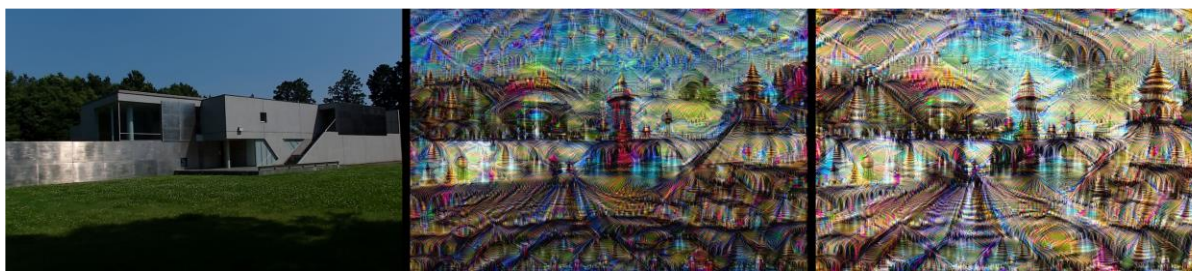


Rys. 8. Fantastyczne krajobrazy „wyśnione” przez opracowany przez Google algorytm DeepDream, zaaplikowany do sieci neuronowej służącej klasyfikacji obrazów. Krajobrazy zostały wygenerowane na podstawie czystego szumu poprzez iteracyjne wzmacnianie na obrazie wejściowym cech zwiększających siłę aktywacji wybranych neuronów i warstw sieci neuronowej (Mordvintsev, i in., 2015).

Zabieg ten pozwala nie tylko na uczytelnienie samego procesu klasyfikacji, ale także na generowanie nowych, fantastycznych obrazów „wyśnionych” przez sieć neuronową na podstawie wykorzystanego zestawu treningowego i zadanego obrazu startowego (Rys. 8).

Niewiele później generatywne zdolności sieci neuronowych zostały ujarzmione w pokrewnych do DeepDream algorytmach neuronowego transferu stylu, pozwalających na kontrolowane i precyzyjne przetwarzanie stylu jednego obrazu tak, by przypominał stylistycznie inny obraz

(Gatys, i in., 2016). Już w 2016 roku badacze z Keio University w Japonii przeprowadzili wstępną eksplorację architektonicznych zastosowań DeepDream i transferu stylu. DeepDream został wykorzystany do generowania nowej warstwy detali architektonicznych nałożonych na zadany obraz wejściowy, przedstawiający dom jednorodzinny (Rys. 9) oraz do generowania własnych założeń architektonicznych od zera na podstawie szumu i obrazów treningowych przedstawiających „miejsca”¹². Neuronowy transfer stylu został przetestowany w przetwarzaniu wizualizacji architektonicznej oraz w wizualizacji zmiany materiałów konstrukcyjnych i wykończeniowych wykorzystanych przy danym obiekcie architektonicznym. Efekty tych początkowych prób były obiecujące, jednak dało się w nich zauważyć brak kontroli nad generowanymi rozwiązaniami, które niewiele odbiegały od bazowych wizualizacji DeepDream i neuronowego transferu stylu zaprezentowanych przez Google oraz Gatysa, Eckera i Bethge (Silvestre, i in., 2016).

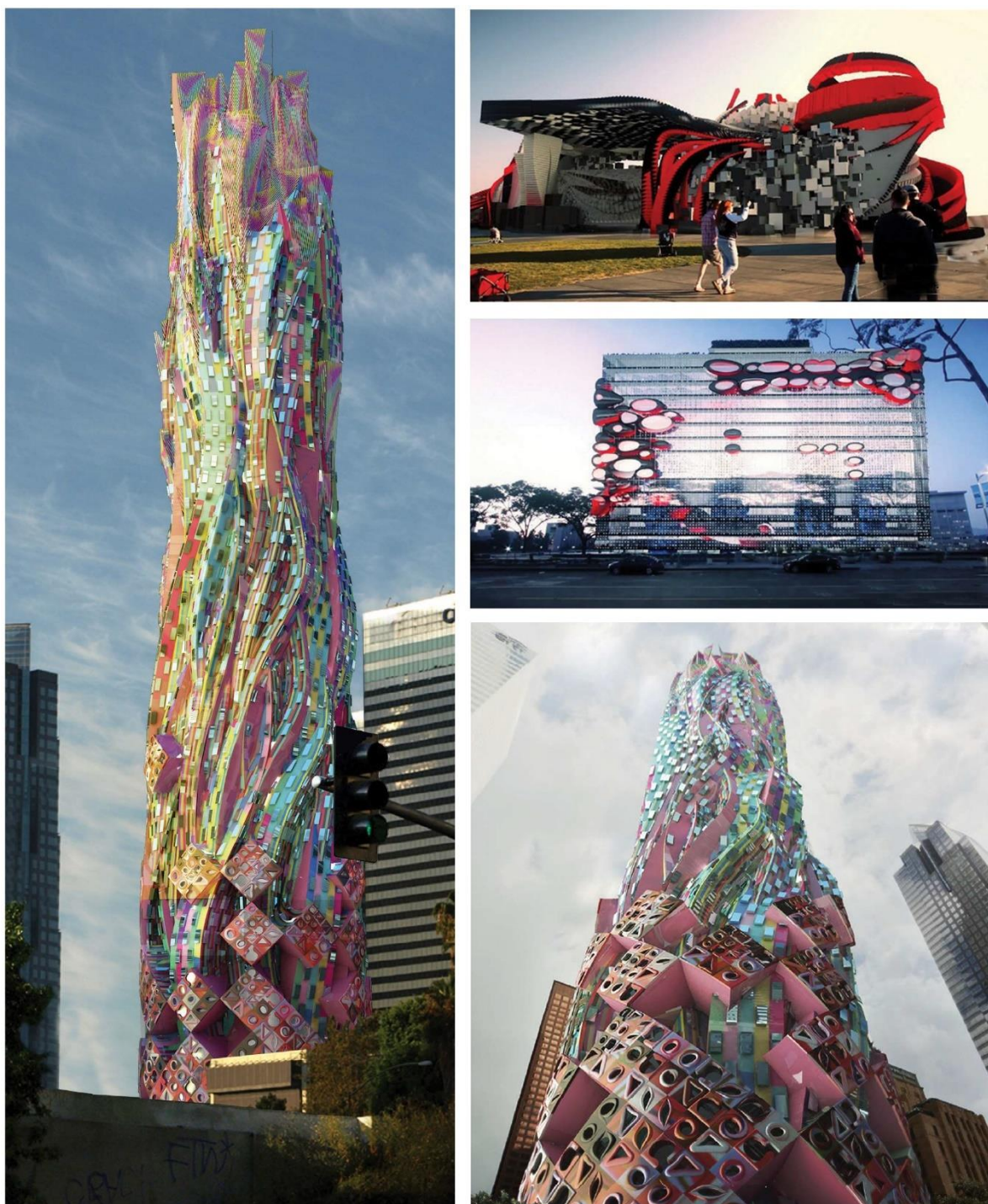


Rys. 9. Sekwencja generowania warstwy nowych detali architektonicznych na zadany zdjęciu domu jednorodzinnego. Metoda zaproponowana przez Silvestra, Ikedę i Guenę z Keio University, oparta na opracowanym przez Google algorytmie DeepDream (Silvestre, i in., 2016).

Obecnie architekci coraz lepiej radzą sobie z kontrolowaniem transferu stylu i algorytmów typu DeepDream w przetwarzaniu zdjęć i wizualizacji architektonicznych oraz znajdują dla nich zastosowania wykraczające poza ich podstawowe funkcje. Matias del Campo, Alexandra Carlson i Sandra Manninger zebrali własny zbiór danych treningowych dla sieci neuronowej przetwarzanej przez algorytm DeepDream. Stworzony zestaw danych „Architecture Parts Dataset” składał się z wielu przykładów konkretnych elementów architektonicznych, takich jak fontanny, ławki, łuki, itp. W założeniu, sieć wytrenowana na takich danych treningowych i przetworzona przez DeepDream powinna lepiej radzić sobie z generowaniem wyobrażeń

¹² MIT Places Database (Zhou, i in., 2014).

nowych rozwiązań architektonicznych. Opracowany system potrafił, przykładowo, na zadanym w formie wizualizacji kontekście architektonicznym proponować lokalizację i formę fontann, skalniaków czy innych detali (del Campo, i in., 2020). Korzystając z transferu stylu jako punktu wyjścia do procesu projektowego, Güvenç Özel i Benjamin Ennemoser osiągnęli imponujące wizualnie efekty w przekształcaniu istniejących obiektów architektonicznych na fantastyczne wariacje na ich temat (Rys. 10). Autorzy opracowali metodę proceduralnego tworzenia modeli 3D o wysokiej szczegółowości z wygenerowanych przez sieć neuronową obrazów dwuwymiarowych. Dzięki fotogrametrycznej rekonstrukcji otoczenia oryginalnych budynków możliwe jest także kompozytowe osadzenie wygenerowanych modeli w istniejących kontekstach, umożliwiające rendering szczegółowych wizualizacji (Özel i Ennemoser, 2019).



Rys. 10. Wizualizacje stworzone przez Güvença Özela i Benjamin Ennemosera na podstawie neuronowego transferu stylu. Wygenerowane przez sieć neuronową dwuwymiarowe obrazy zostały przekształcone proceduralnie w modele trójwymiarowe i osadzone kompozytowo w oryginalnych kontekstach (Özel i Ennemoser, 2019).

Przy zastosowaniach generatywnych sieci neuronowe mogą korzystać także z danych wejściowych innych niż zdjęcia perspektywiczne. Dostyc często podejmowane są próby generowania architektury na podstawie rzutów i przekrojów przedstawionych w formie obrazów rastrowych. Kontynuując swoje eksperymenty z transferem stylu i testując możliwości generatywnych sieci przeciwnych [ang. GAN – Generative Adversarial Network (Goodfellow, i in., 2014)] del Campo, Carlson i Manninger wykorzystali głębokie uczenie maszynowe do tworzenia nowych rzutów architektonicznych. Zaczynając od transferu stylu, autorzy zademonstrowali możliwość przetworzenia rzutów w stylu barokowym na rzuty w stylu modernistycznym, co poskutkowało wygenerowaniem abstrakcyjnych planów łączących cechy obu stylów. Przetwarzając rzuty dalej, przy pomocy sieci przeciwnych, wygenerowano obrazy na tyle oryginalne, że rozróżnienie przestrzeni od elementów ją ograniczających przestało być możliwe, a same rzuty przestały być rozpoznawalne jako rysunki architektoniczne (Rys. 11). Wygenerowane reprezentacje zostały nazwane przez autorów „architekturą ery postludzkiej” (del Campo, i in., 2019).



Rys. 11. Defamiliaryzacja rzutu architektonicznego będąca skutkiem prac Matiasa del Campo, Sandry Carlson i Alexandry Manninger. Po lewej rezultat transferu stylu między rzutem modernistycznym a barokowym (del Campo, i in., 2019), w centrum z lewej rezultat transferu stylu między rzutem barokowym a modernistycznym (del Campo, i in., 2020), po prawej abstrakcyjne „rzuty” stworzone przy pomocy generatywnych sieci przeciwnych (del Campo, i in., 2019).

W swoich pracach del Campo i in. wychodzili z założenia, że sieci neuronowe zdolne są odczytywać w planach architektonicznych elementy symboliczno-strukturalne (takie jak okna, ściany, pustki wewnątrz etc.) oraz stylistyczne (jak grubość ścian, krzywizny, symetrię etc.) i następnie potrafią przetwarzać te elementy w celu generowania nowej, oryginalnej architektury. Teza ta została poparta poprzez stworzenie przy pomocy sieci przeciwnych „architektury modernistycznej w stylu architektury barokowej”, w której symetria, masywność

i bujność założeń barokowych ścierają się symbiotycznie z asymetrią, otwartością i racjonalnością rzutów modernistycznych (del Campo, i in., 2020).

Do generowania rzutów architektonicznych przy użyciu sieci neuronowych podszedł także David Newton, wspomniany wcześniej w kontekście urbanistycznych analiz korelacji między zdrowiem mieszkańców miast a morfologią tkanki miejskiej (Newton, i in., 2020; Newton, 2021; Newton, 2020). Zauważył on, że architekci często nie dysponują odpowiednio dużym archiwum wzorcowych projektów, które mogłyby zostać wykorzystane przy treningu sieci neuronowych. Zazwyczaj stworzenie dużego zestawu rzutów treningowych, mogących być wykorzystanymi przy generowaniu konkretnego projektu, jest bardzo czasochłonne i kosztowne. Newton przetestował więc kilka metod zwiększenia ograniczonej liczby danych treningowych (ang. Data Augmentation) poprzez losowe obracanie rzutów o niewielki kąt lub nakładanie na nie losowego szumu o niewielkim natężeniu. Najlepsze efekty przyniosło nakładanie na rzuty treningowe szumu (Newton, 2019). Wydaje się, że jeszcze oprócz syntetycznego zwiększenia liczby obrazów treningowych warto byłoby w takim przypadku skorzystać dodatkowo z uczenia transferowego (ang. Transfer Learning), czyli wytrenować algorytm na obszernym zbiorze zróżnicowanych rzutów architektonicznych i później kontynuować trening tylko na rzutach wzorcowych dla konkretnego projektu, w celu dostrojenia sieci do zadanego problemu (ang. Fine-Tuning). W czasach popularności metod BIM-owskich obszernym źródłem danych treningowych dla głębokiego uczenia maszynowego mogą być wielobranżowe modele BIM. Zespół z japońskiej korporacji Takenaka sprawdził możliwości automatycznego generowania instalacji przeciwpożarowych na podstawie zadanych rysunków architektonicznych. Rzuty służące treningowi sieci i generowaniu instalacji zostały wyeksportowane w formie rastrowych obrazów z BIM-owskiego modelu branży architektonicznej. Archiwum historycznych projektów instalacyjnych, do którego miała dostęp korporacja, było wystarczająco obszerne do wytrenowania skutecznej sieci neuronowej zdolnej do sensownego generowania instalacji przeciwpożarowych, pokrywających wszystkie niezbędne do obsłużenia obszary budynku (Sato, i in., 2020). Przenosząc się z płaszczyzny poziomej do płaszczyzny pionowej, Mohammad, Boerkrem i Ellinger zajęli się problemem generowania elewacji budynków. Przeciwna sieć neuronowa została przez nich wykorzystana do eksploracji przestrzeni projektowej w poszukiwaniu możliwych do zrealizowania fasad hybrydowych, łączących w sobie cechy dwóch różnych typów elewacji. W pierwszym kroku przy pomocy algorytmów genetycznych parametrycznie wymodelowano dwa zestawy elewacji o różniących się między sobą charakterystykach. W jednej grupie

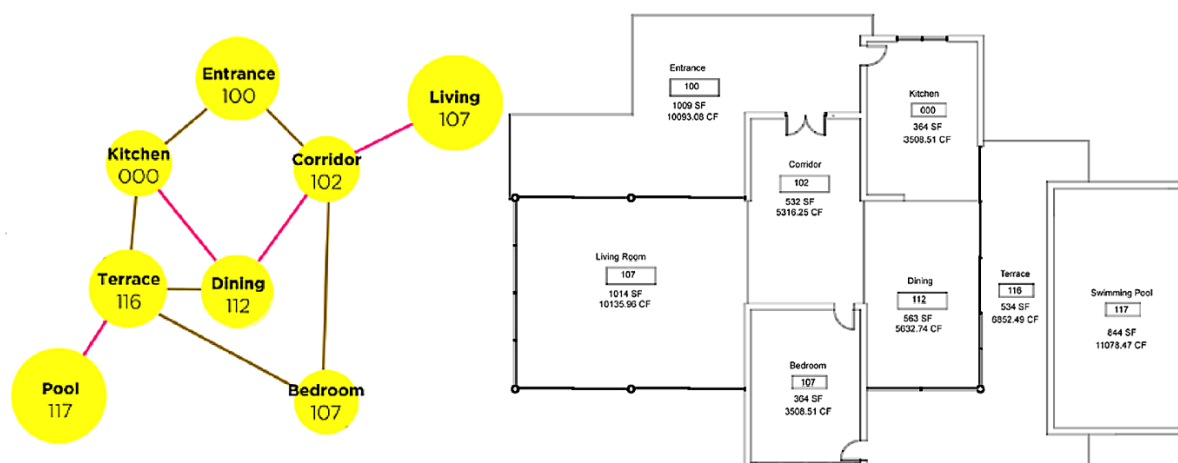
znajdowały się elewacje ciągłe, uformowane z wielokrzywiznowych płaszczyzn, w drugiej, elewacje z silnie akcentowanymi podziałami, uformowane z prostopadłościennych modułów. Sieć neuronowa została wytrenowana na obu zbiorach jednocześnie i w efekcie nauczyła się generować hybrydowe elewacje łączące cechy obu grup jednocześnie (Mohammad, i in., 2019). Większość popularnych algorytmów głębokiego uczenia maszynowego, implementowanych w ogólnodostępnych bibliotekach programistycznych, domyślnie służy przetwarzaniu obrazów dwuwymiarowych. Z tego powodu wiele z ich architektonicznych zastosowań także operuje na rysunkach i innych reprezentacjach dwuwymiarowych. Chuan Liu i in. zaproponowali prostą metodę przeniesienia twórczych algorytmów opartych na transferze stylu w trzeci wymiar, poprzez generowanie kratownicowej struktury nośnej budynku wysokościowego osobno dla kolejnych kondygnacji na podstawie ich obrysów. Stworzone w ten sposób kratownice zostały połączone z kratownicami przyległych kondygnacji przy pomocy parametrycznie generowanych skośnych słupów (Liu, i in., 2020). Głównym problemem takiego prostego rozwiązania był jednak brak logicznej sekwencyjności wygenerowanych kondygnacji i relacji między poszczególnymi piętrami. Z tego powodu słupy łączące kondygnacje były generowane losowo, a to dało dosyć chaotyczne rezultaty.

Sieci neuronowe są jednak bardzo elastyczne co do formatu danych wejściowych. Nic nie stoi na przeszkodzie, by zamiast danych przedstawionych w formie obrazów, przetwarzać projekty architektoniczne zapisane przy pomocy innych form reprezentacji danych. Szczególnie obiecujące są numerycznie reprezentowane projekty parametryczne. Przykładowo, projekt parametryczny wygenerowany w środowisku Dynamo lub Grasshopper może być reprezentowany przez sekwencyjny szereg operacji przetwarzających strumień danych początkowych w celu wygenerowania formy docelowej projektu. Tak reprezentowany projekt parametryczny jest zbliżony swoją formą do struktury zdania lub algorytmu (np. „weź punkt A, potraktuj go jako środek koła o promieniu r , wytłocz koło prostopadle do jego płaszczyzny na odległość h tworząc walec...”). Varvara Toulkeridou zauważyła podobieństwo tak reprezentowanego projektu do zdania i postanowiła wykorzystać sieci neuronowe stosowane zazwyczaj do przetwarzania języka naturalnego (ang. NLP: Natural Language Processing) do eksploracji morfologicznej projektu parametrycznego. Sieć neuronowa została wykorzystana do sugerowania alternatywnych dróg przepływów danych przez różnorodne, dostępne w środowisku operacje. Przykładowo – zadając sieci początek diagramu opisującego przekształcenia geometryczne projektu, można wygenerować jego kontynuację w celu sprawdzenia możliwych do uzyskania efektów końcowych. Opracowany przez Toulkeridou

system był w stanie tworzyć poprawne syntaktycznie i semantycznie projekty parametryczne, skutkujące wygenerowaniem nowych geometrii (Toulkeridou, 2019). Alternatywną metodą parametryzacji projektu architektonicznego jest przedstawienie go w formie grafowej, w której program funkcjonalny budynku reprezentowany jest przez węzły oznaczające poszczególne funkcje i krawędzie łączące funkcje ze sobą. Taka formalizacja projektu z jednej strony pozwala na złożone przedstawienie programu funkcjonalnego przekładającego się na formę obiektu, a z drugiej strony jest bardzo prosta do wymodelowania matematycznego przy pomocy teorii grafów. W głębokim uczeniu maszynowym do przetwarzania takich danych używa się grafowych sieci neuronowych oraz sieci Kohonena (ang. SOM: Self-Organizing Map). W 2018 roku Imdat As, Siddharth Pal i Prithwish Basu skorzystali z bazy projektów domów jednorodzinnych platformy Arcbazar i dokonali wyboru modeli dostępnych w formacie BIM-owskim. Modele zostały automatycznie przekształcone do formy grafów funkcjonalnych z przypisanymi powierzchniami pomieszczeń. Projektom nadano także manualnie oszacowane, liczbowe wartości „komfortu życia” i „komfortu spania”. Oprócz wykorzystania sieci do generowania nowych programów funkcjonalnych, autorom udało się także zidentyfikować podgrafy funkcjonalne pozytywnie skorelowane z metrykami komfortu. Przykładowo, według wytrenowanej sieci neuronowej na komfort spania pozytywnie wpływa układ składający się z kilku osobnych sypialni połączonych z łatwo dostępnymi schowkami, garderobami i łazienką. Proponowane przez sieć diagramy mogą łatwo być przetworzone przez algorytm parametryczny na trójwymiarowy, kubaturowy model koncepcyjny (Rys. 12) (As, i in., 2018).

Podobny system został równolegle opracowany przez Eisenstadta, Langenhana i Althoffa. W tym przypadku także wykorzystano grafową reprezentację układów funkcjonalnych i wytrenowano algorytm głębokiego uczenia maszynowego w celu generowania nowych układów na potrzeby koncepcyjnej fazy projektowej. Model zaproponowany przez Eisenstadta i in. jest jednak bardziej zaawansowany ze względu na o wiele większy zestaw danych treningowych (1000 razy większy) i zdolność algorytmu do przewidywania także możliwych zmian w układzie funkcjonalnym wprowadzanych przez mieszkańców w trakcie użytkowania obiektu (Eisenstadt, i in., 2019). Przestrzenny model architektoniczny można przedstawić też w postaci chmury punktów. Jose Algeciras-Rodriguez, korzystając z takiej reprezentacji, zaproponował metodę tektonicznej eksploracji modeli trójwymiarowych poprzez przetwarzanie modeli wejściowych przy pomocy sieci Kohonena. Metoda ta pozwala

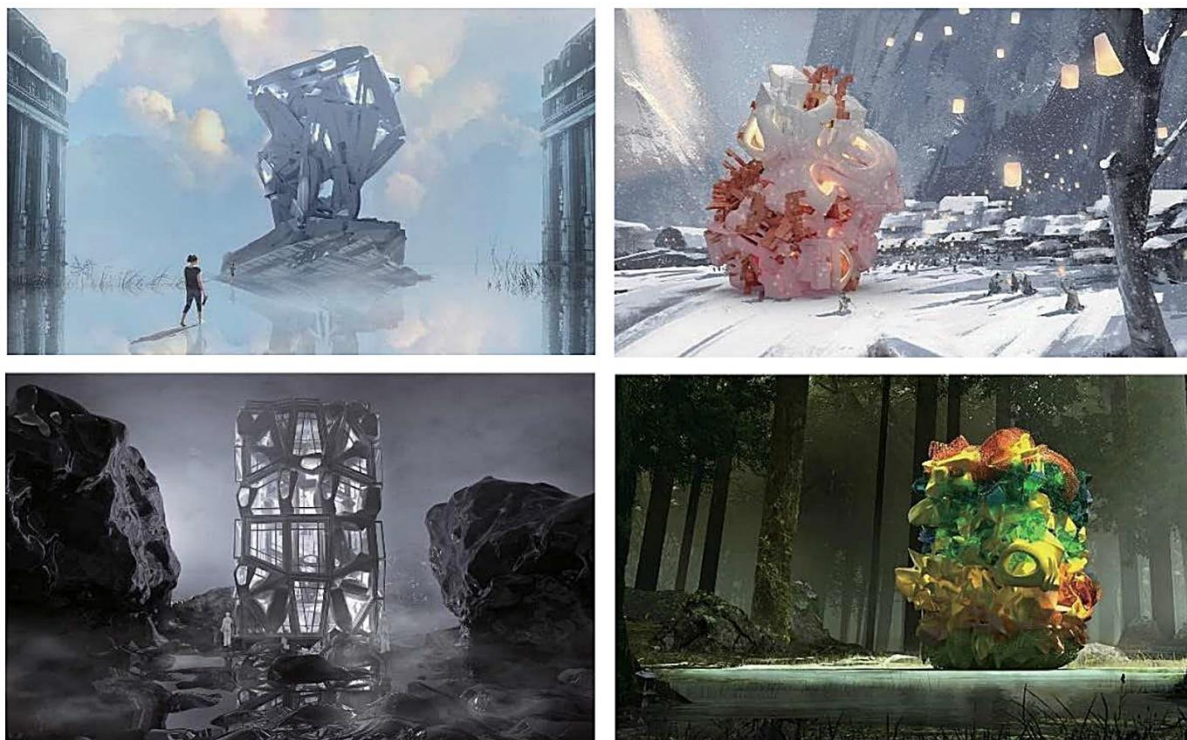
na generowanie hybrydowych form łączących cechy danych modeli referencyjnych. Zademonstrowane przez autora wyniki interpolacji między modelem *Wenus z Milo* a *Nike z Samotraki* są dosyć zaszumione i widać w nich pewien brak kontroli nad algorytmem, ale sam proces wydaje się interesujący i potencjalnie użyteczny (Algeciras-Rodriguez, 2018).



Rys. 12. Demonstracja skuteczności systemu opracowanego przez Asa i in. Po lewej wygenerowany przez sieć neuronową graf programu funkcjonalnego domu jednorodzinnego, po prawej automatycznie wygenerowany na podstawie grafu parametryczny, koncepcyjny model kubaturowy (As, i in., 2018).

Gdy brakuje istniejących zestawów danych, umożliwiających reprezentowanie pożądanej funkcji generatywnej w postaci zestawu treningowego, można skorzystać z półautomatycznych metod generowania przykładów lub sformułować problem przy pomocy symulacji głębokiego uczenia ze wzmacnianiem. Metody półautomatyczne pozwalają na zwiększenie kontroli nad stosowanymi algorytmami parametrycznymi. Liu, Liao i Srivastava wygenerowali przy pomocy nieneuronowego algorytmu zbiór zróżnicowanych brył zbudowanych z wymodelowanych przez autorów elementów modularnych. Wygenerowane bryły zostały później oznaczone ręcznie względem czterech cech: monumentalności, delikatności, solidności i dynamiczności. Na podstawie stworzonego, oznaczonego zestawu modeli wytrenowano sieć przeciwną generującą nowe parametry dla algorytmu wykorzystanego do stworzenia zestawu początkowego. Sieć definiowała parametry w ten sposób, by wynikowe bryły spełniały wymagane cechy (Rys. 13). Przy takim podejściu konieczne jest ręczne oznaczenie

niewielkiego zbioru wygenerowanych, przypadkowych modeli, ale w zamian zyskuje się kontrolę nad dalszymi rezultatami generowanymi przez algorytm (Liu, i in., 2019).



Rys. 13. Bryły wygenerowane przez system opracowany przez Liu, Liao i Srivastavę.

Przeciwnastawna sieć neuronowa generowała takie parametry dla algorytmu parametrycznego, by wynikowe bryły w jak największym stopniu spełniały wymagane cechy. Po lewej u góry – bryła z wysokim wynikiem „monumentalności”, po prawej u góry – „solidności”, po lewej u dołu – „delikatności” i po prawej u dołu – „dynamiczności” (Liu, i in., 2019).

Podejścia wykorzystujące głębokie uczenie ze wzmacnianiem są bardzo zróżnicowane pod względem stosowanych metod, inspiracji i poziomu abstrakcji. Największą zaletą uczenia sieci neuronowych poprzez symulacje interakcji agent – środowisko jest duża kontrola projektanta nad całością procesu generatywnego. Alessandro Mintrone i Elessio Erioli z uniwersytetu w Bolonii opracowali hybrydowy system generujący zwokselizowane, architektoniczne układy przestrzenne. System oparty był na głębokim uczeniu ze wzmacnianiem i algorytmie „Wave Function Collapse” autorstwa Maxa Gumina (Gumin, 2016). WFC jest algorytmem generatywnym do układów przestrzennych tworzącym złożone, nieoczywiste kompozycje przestrzenne w dwóch lub trzech wymiarach na podstawie przykładowych, możliwych do wykorzystania fragmentów. W pracy Mintrone’a i Erioli WFC decyduje o lokalizacji

i sposobie łączenia trójwymiarowych modułów, a DRL generuje wypełnienie modułu w sposób najbardziej pasujący do kontekstu określonego przez optymalizację wskaźników gęstości globalnej i lokalnej, orientacji, łączności konstrukcyjnej i przestrzennej oraz ciągłości płaszczyzn. Skuteczność systemu zademonstrowano generując wielopoziomowy, stabilny konstrukcyjnie obiekt architektoniczny o interesujących, wzajemnie połączonych przestrzeniach wewnętrznych (Mintrone i Erioli, 2021). Bastian Wibranek i in. wykorzystali głębokie uczenie ze wzmacnianiem do tworzenia kompozycji przestrzennych złożonych z „bloków SL” [ang. SL-Block (Shih, 2016)]. Bloki SL to struktury, które można łączyć ze sobą w różne formy tak, że bloki wzajemnie się unieruchamiają. Dzięki temu wynikowa forma jest statyczna. Forma docelowa jest pod pełną kontrolą projektanta. Poprzez optymalizację procesu hierarchicznego i sekwencyjnego układania bloków względem podobieństwa do formy docelowej, możliwe jest wytrenowanie sterowanego przez sieć neuronową agenta tworzącego złożone formy architektoniczne stabilne nie tylko po wybudowaniu, ale i w trakcie montażu (Wibranek, i in., 2021). Możliwe jest także podejście do problemu od drugiej strony, w której to forma docelowa jest wynikiem procesu twórczego algorytmu, a elementy składowe są projektowane metodami klasycznymi przez architekta. Hosmer, Tigas, Reeves i He zaproponowali ramową metodę generowania megastruktur mieszkaniowych złożonych z zaprojektowanych klasycznie, modularnych jednostek mieszkalnych składających się z jeszcze mniejszych bloków konstrukcyjnych opartych o siatkę przestrzenną rozpiętą na modułach 3,5 x 3,5 x 3,5 m. Sieć neuronowa steruje algorytmem, próbując balansować między maksymalizacją metryk celów globalnych (jak minimalizacja kosztów materiałowych lub maksymalizacja współdzielonych przestrzeni wspólnych), a lokalnych (prywatność, otwartość na przestrzeń zewnętrzną), mieszcząc się jednocześnie w ograniczeniach projektowych, takich jak granice działki, założone wskaźniki zabudowy i konteksty zewnętrzne (Hosmer, i in., 2020). Najbardziej pragmatycznym z omawianych przykładów wykorzystania głębokiego uczenia ze wzmacnianiem w projektowaniu generatywnym jest „Akademia Agentów Przestrzennych” autorstwa Pedra Velosa i Ramesha Krishnamurtiego, służąca do projektowania w formie diagramów funkcjonalnych domów jednorodzinnych. „Akademia” stanowi symulację wieloagentową, w której każde pomieszczenie reprezentowane jest przez sieć neuronową. Pomieszczenia „walczą” ze sobą i jednocześnie „współpracują” w celu wygenerowania projektu domu jednorodzinnego maksymalizującego funkcję przystosowania, będącą wynikową metryk formy, powierzchni i wzajemnego układu funkcjonalnego pomieszczeń (Velo i Krishnamurti, 2020).

Architektura: ewaluacja i przetwarzanie danych

Drugim najpopularniejszym zastosowaniem sieci neuronowych w skali architektonicznej jest ewaluacja i przetwarzanie danych. W najogólniejszym znaczeniu chodzi tu o bezpośrednie przetworzenie danych jednego rodzaju na inną formę lub na przetworzenie ich na pojedynczą metrykę w celu oceny pewnego aspektu reprezentowanego przez dane. Zakres możliwych zastosowań jest bardzo szeroki i obejmuje zarówno pragmatyczne algorytmy służące rozwiązywaniu jednostkowych problemów, jak i ambitne, złożone systemy przetwarzające wiele kontekstów jednocześnie. Ciekawym przykładem praktycznego przetwarzania danych przy pomocy sieci neuronowej jest system HorizonNet autorstwa Chenga Suna i in. HorizonNet umożliwia przetworzenie panoramicznego zdjęcia, przedstawiającego wnętrze pomieszczenia, na jego oteksturowany model 3D. Sieć neuronowa odczytuje na zdjęciach panoramicznych krawędzie między sufitem, ścianami i podłogą i na tej podstawie generuje jednowymiarową reprezentację układu ścian. Reprezentacja taka jest wystarczająca do zrekonstruowania modelu trójwymiarowego całego pomieszczenia. Dzięki odpowiednio dużej grupie w zbiorze treningowym przykładów podchwytliwych sieć nauczyła się także prawidłowo generować niewidoczne, ukryte za załomami fragmenty pomieszczeń (Sun, i in., 2019). Przykładem bardziej złożonego systemu jest wieloskalowy projekt koncepcyjny „Emoting City” autorstwa Sayjela Vijaya Patela i in. W założeniu tym sieci neuronowe mają służyć przetwarzaniu emocji użytkowników przestrzeni zbudowanej na aktywnej, fizycznej odpowiedzi elementów architektonicznych. Pomysł został zademonstrowany na niewielkiej instalacji, w której zachowanie użytkownika przetwarzano przy pomocy sieci splotowych na ruch i oświetlenie zautomatyzowanych, ruchomych luster reagujących na emocje osoby korzystającej z instalacji. W pełnoskalowej implementacji „Emoting City” interfejsem sterowania byłoby samo zachowanie człowieka odczytane przez sensory i przetwarzane przez sieci neuronowe. Elementami aktywnymi, dostosowującymi przestrzeń do człowieka, mogłyby być roboty, ruchome powierzchnie lustrzane i przestrzenie autoplastyczne, potrafiące zmieniać swoją formę w odpowiedzi na odebrane bodźce sensoryczne (Patel, i in., 2020).

Zdolności sieci neuronowych do aproksymowania różnorodnych funkcji wykorzystywane są do przybliżania wyników kosztownych pod względem obliczeniowym symulacji komputerowych. Sieci neuronowe łatwo można wytrenować na rzeczywistych danych pomiarowych lub na precyzyjnych wynikach bardziej skomplikowanych symulacji. Wytrenowana sieć może generować wyniki zbliżone do rzeczywistych bez konieczności dokonywania kosztownych pomiarów lub przeprowadzania czasochłonnych symulacji

wymagających dużych zasobów obliczeniowych. W latach 90., gdy symulacje energetyczne budynków nie były jeszcze szeroko stosowane, sieci neuronowe były jedynymi dostępnymi algorytmami potrafiącymi przeprowadzać trafne symulacje pewnych złożonych procesów. W 1994 roku William Stevenson wytrenował na danych pomiarowych sieć neuronową przetwarzającą wejściowe szeregi czasowe opisujące nasłonecznienie budynku i temperaturę na zużycie energii elektrycznej oraz zużycie ciepłej i zimnej wody. Dzięki takiej sieci możliwe jest przewidywanie wydajności energetycznej budynku względem warunków zewnętrznych. Biorąc pod uwagę prostotę wykorzystanej sieci, osiągnięte wyniki były zaskakująco dokładne. Najgorzej sieć radziła sobie z przewidywaniem zużycia energii elektrycznej, ponieważ przez brak dostępu do informacji o dniu tygodnia pomiaru sieć nie była w stanie uchwycić weekendowych zmian w zapotrzebowaniu na energię elektryczną (Stevenson, 1994b). Ioannis Chatzikonstantinou i Sevil Sariyildiz przeprowadzili bardzo skrupulatne badanie zdolności różnych algorytmów uczenia maszynowego do przybliżania dwóch metryk komfortu wizualnego w przestrzeniach biurowych: autonomii¹³ światła dziennego DA (ang. Daylight Autonomy) i współczynnika prawdopodobieństwa olśnienia światłem dziennym DGP (ang. Daylight Glare Probability). Przetestowane algorytmy obejmowały jednokierunkowe sieci neuronowe FNN (ang. Feedforward Neural Network), maszyny wektorów nośnych SVM (ang. Support Vector Machine) oraz losowe lasy decyzyjne RF (ang. Random Forest). Dane do treningu sieci neuronowej i metod alternatywnych wygenerowano przy pomocy symulacji komputerowej. W przewidywaniu autonomii światła dziennego najlepiej radziła sobie maszyna wektorów nośnych, ale już przy bardziej skomplikowanej metryce prawdopodobieństwa wystąpienia olśnienia, najlepsze wyniki osiągnęła sieć neuronowa (Chatzikonstantinou i Sariyildiz, 2016). Asl, Das, Tsai, Molloy i Hauck stworzyli hybrydowy system „Energy Model Machine”, wykorzystujący sieć neuronową i drzewo decyzyjne (ang. Decision Tree) do ewaluacji wydajności energetycznej budynków. Model został wytrenowany na danych wygenerowanych symulacyjnie. System umożliwia przyspieszenie procesu projektowania iteratywnego dzięki ominięciu konieczności przeprowadzania pełnych symulacji przy każdym wygenerowanym układzie przestrzennym. Na zademonstrowanym przez autorów przykładzie parametrycznego projektu budynku biurowego implementacja algorytmu

¹³ W tym kontekście „autonomia” oznacza procent czasu, w którym poziom oświetlenia przekracza wartość docelową (Heschong, i in., 2012).

pozwoili na dokladniejsz eksploracj przestrzeni projektowej. Możliwa byla szybka ewaluacja energetyczna az siedmiu tysicy wariantow (Asl, i in., 2017). Podobny algorytm zaproponowali Singh, Schneider-Marin, Harter, Lang i Geyer. Stworzyli oni system pozwalajacy na ewaluacje wydajnosci srodowiskowej budynku we wczesnej fazie projektowania poprzez probabilistyczne szacowanie kosztow energii zuzytej w calym cyklu zycia obiektu. Podobnie jak przy propozycji Asla i in., glowna korzyscia plynaca z zastosowania algorytmu jest usprawnienie procesu projektowego (Singh, i in., 2020). Walter Mazuroski i in. wykorzystali sieci neuronowe do przyblizania wynikow szeregu symulacji typu BPS (ang. Building Performance Simulations), w tym przeplywu powietrza we wnetrzach, trojwymiarowego transferu ciepla oraz symulacje systemow instalacyjnych (ogrzewanie, wentylacja i klimatyzacja) (Mazuroski, i in., 2018). Podchodzacy do problemu fragmentarycznie, Adam Sebestyen i Jakub Tyc skupili sie na wydajnosci srodowiskowej samych elewacji i wytrenowali siec neuronowa przewidujaca, na podstawie zadanej geometrii, spodziewane zacienienie wytworzone przez projektowana elewacje i intensywnosc promieniowania slonecznego we wnetrzach (Sebestyen i Tyc, 2020). Ze wzgledu na swoja uniwersalnosc, sieci neuronowe moga byc takze stosowane przy przyblizaniu wynikow bardziej skomplikowanych symulacji, ulatwiajacych podejmowanie decyzji projektowych. Sherif Tarabishy, Stamatios Psarras, Marcin Kosicki i Martha Tsigkari opracowali neuronowy system upraszczajacy zlozone, czasochlonne symulacje sluzace ocenie lacznosci przestrzennej i wizualnej w projektowanej przestrzeni. Stworzony algorytm moze przetworzyc dowolny rzut architektoniczny na mapy intensywnosci polaczen miedzy pomieszczeniami oraz na diagramy lacznosci wizualnej. Analizy takie sa szczegolnie uzyteczne przy projektowaniu miejsc pracy w przestrzeniach biurowych. System dziala wystarczajaco sprawnie, by oceniać rzuty tworzone przez architekta w czasie rzeczywistym (Tarabishy, i in., 2020).

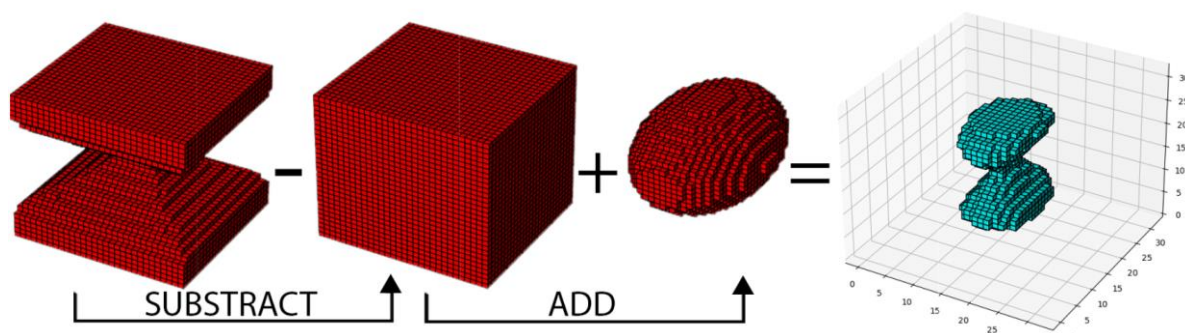
Ciekawe zastosowanie dla sieci neuronowych znalezi takze Nicholas, Chen, Borpujari, Bartoc i Refsgaard ze Szkoły Architektury Duńskiej Akademii Królewskiej. Stworzyli oni sekwencyjny system zlozony z kilku dzialajacych w tandemie sieci neuronowych, sluzacych ewaluacji skutkow modernizacji elewacji i dachow domow jednorodzinnych. Najpierw, przy pomocy transferu stylu, generowana jest wizualizacja pokrycia budynku nowymi, izolacyjnymi panelami wykończeniowymi. Potem, na podstawie wygenerowanych elewacji, inna siec neuronowa przeprowadza gradientowe symulacje energetyczne biorace pod uwage strefe klimatyczna, w ktorej znajduje sie dany budynek. Kolejna siec neuronowa przetwarza zebrane w ten sposob informacje w celu obliczenia wskaźnikow skalarnych przeprowadzonej

modernizacji: przewidywanego kosztu zakupu i instalacji paneli oraz zyski z ograniczenia emisji CO₂ do atmosfery wynikające ze zmniejszenia zapotrzebowania energetycznego budynku. W artykule opisującym opracowany system autorzy zwracają uwagę na korzyści wynikające z możliwości eksploracji przestrzeni ukrytej (ang. Latent Space) sieci neuronowej. W czasie treningu sieć neuronowa tworzy wewnętrzne reprezentacje wyuczonej dystrybucji danych treningowych w postaci hiperprzestrzeni, w których wymiarami są możliwe wartości aktywacji neuronów w jednej z warstw sieci¹⁴. Wszystkie możliwe do wygenerowania przez sieć dane wyjściowe są reprezentowane jako punkty lub wektory próbkowane z przestrzeni ukrytej. Nicholas i in. wykorzystali reprezentację w przestrzeni ukrytej do próbkowania z niej wizualizacji o różnym stopniu pokrycia elewacji i dachu zmodernizowanymi panelami. Po przeprowadzeniu redukcji wymiarów z przestrzeni wielowymiarowej na czterowymiarową przy pomocy analizy głównych składowych PCA (obrócenie układu współrzędnych i rzutowanie na przestrzeń mniej wymiarową), autorzy zidentyfikowali dwie główne osie składowe przestrzeni ukrytej: jedną odpowiadającą za liczbę paneli na ścianach, drugą – za liczbę paneli na dachu. Zidentyfikowane osie były wykorzystane do liniowej kontroli nad stopniem pokrycia budynków nowymi panelami (Nicholas, i in., 2021). Z tandemowych sieci neuronowych przetwarzających sekwencyjnie dane wejściowe skorzystali także Hamid Mehmood i Nitin K. Tripathi w opracowanym w 2011 roku systemie lokalizacyjnym działającym we wnętrzach budynków na podstawie siły sygnału Wi-Fi. Stworzony system potrafił w przybliżeniu oceniać położenie w budynku w trzech osiach z dokładnością średnią wynoszącą 1,9m, co zapewniało lepszą precyzję lokalizacji we wnętrzach niż system GPS (Mehmood i Tripathi, 2013).

Przetwarzanie danych wejściowych do ich reprezentacji w przestrzeni ukrytej jest bardzo potężnym narzędziem umożliwiającym nie tylko kontrolę nad samą siecią neuronową, lecz także przeprowadzanie operacji geometrycznych i przestrzennych, których nie dałoby się przeprowadzić bezpośrednio na przestrzeni cech danych wejściowych. Jaime de Miguel i in. zaproponowali metodę eksploracji przestrzeni projektowej opartej na reprezentacji w przestrzeni ukrytej. Zaczynając od dwóch początkowych, uproszczonych modeli architektonicznych reprezentowanych w formie mapy połączeń w zwokselizowanej przestrzeni

¹⁴ W klasycznych, w pełni połączonych sieciach jednokierunkowych, każdej warstwie sieci odpowiada osobna przestrzeń ukryta o liczbie wymiarów odpowiadającej liczbie neuronów.

trójwymiarowej, wygenerowano zbiór 3000 modeli będących pochodnymi modeli podstawowych poddanych pewnym przekształceniom (ang. Data Augmentation). Modele te zostały wykorzystane do wytrenowania sieci typu autoenkoder, która stworzyła w przestrzeni ukrytej dwuwymiarową reprezentację dystrybucji modeli treningowych. Próbkując punkty z tej przestrzeni, można generować nowe modele, znajdujące się na dwuwymiarowym spektrum między modelami początkowymi (de Miguel, i in., 2019). Zaproponowana metoda poszukiwania formy byłaby niemożliwa do zrealizowania bez głębokiego uczenia maszynowego. Podobny algorytm został opracowany przez Adama Sebestyena, Johannę Rock i Ursa Leonharda Hirschberga. W tym przypadku także wytrenowano autoenkoder na zbiorze 15000 prostych, wygenerowanych automatycznie brył geometrycznych. Pierwsza połowa modeli oparta była na poddanej przekształceniom kuli, druga na przekształconym sześcianie. Na wejściu sieci neuronowej modele reprezentowane były w formie zwokselizowanej. Autorzy nie tylko zademonstrowali możliwość płynnego przechodzenia między różnymi bryłami, lecz także zaprezentowali możliwość przeprowadzania semantycznych operacji arytmetycznych na geometriach poprzez dodawanie i odejmowanie wektorów reprezentacji w przestrzeni ukrytej (Rys. 14) Wyniki tych operacji dowodzą, że wytrenowana reprezentacja ukryta rozpięta jest na składowych o dosyć wysokim poziomie abstrakcji (Sebestyen, i in., 2021).



Rys. 14. Możliwość przeprowadzania semantycznych operacji arytmetycznych na wektorach w przestrzeni ukrytej zademonstrowana przez Sebestyena i in. Ściśnięty sześcian minus sześcian plus kula = ściśnięta kula (Sebestyen, i in., 2021).

Alternatywne podejście do eksploracji przestrzeni projektów trójwymiarowych brył architektonicznych zaproponowali Hang Zhang i Ye Huang z uniwersytetu w Pensylwanii. Zamiast korzystać z reprezentacji ukrytej zakodowanej przez autoenkoder, autorzy stworzyli własną metodę wykorzystującą przeciwstawne sieci generatywne. W metodzie tej model trójwymiarowy krojony jest najpierw na serię uszeregowanych przekrojów, na których trenuje

się sieć przeciwną. Generowane przez sieć przekroje mogą być z powrotem przekształcone w model trójwymiarowy. Dzięki zastosowanej metodzie mapowania 3D-2D-3D możliwe jest przetwarzanie geometrii trójwymiarowych przy pomocy klasycznych sieci przeciwnych operujących na obrazach dwuwymiarowych. Opracowany algorytm także pozwala na przeprowadzenie eksploracji formalnej. Możliwa jest przykładowo interpolacja między budynkami w dwóch różnych stylach architektonicznych skutkująca wygenerowaniem nowej formy, jak bryły znajdującej się między stylem opery w Sydney a stylem katedry Notre Dame w Paryżu (Zhang i Huang, 2020). Bardzo przyszłościowym kierunkiem badań wydaje się podejście zaproponowane przez Hanga Zhanga z Uniwersytetu Pensylwanii. Autor opracował własny system przetwarzania tekstu naturalnego opisującego szczegółowo projekt architektoniczny na macierze reprezentujące układ funkcjonalny obiektu. Macierze są następnie przetwarzane przez grafową, spłotową sieć neuronową na diagramatyczne, grafowe przedstawienie układu funkcjonalnego pomieszczeń i połączeń między nimi. Na tej podstawie parametrycznie generowany jest model budynku. Pomysł oparcia systemu na tekstowej reprezentacji architektury wydaje się bardzo trafiony ze względu na potencjał wykorzystania w ramach dalszych prac najskuteczniejszych i najpotężniejszych modeli głębokiego uczenia maszynowego, służących przetwarzaniu języka naturalnego (Zhang, 2020). Podobne nadzieje względem możliwości implementacji pochodnych najnowszych algorytmów NLP (ang. Natural Language Processing) budzi system „BIMToVec” autorstwa Mahankaliego, Johnsona i Andersona służący do przetwarzania modeli BIMowskich do zakodowanej semantycznie formy wektorowej (Mahankali, i in., 2018). Jeszcze inny system oparty na grafowej sieci spłotowej opracowali del Campo, Carlson i Manninger. W pierwszym kroku wytrenowano sieć neuronową do klasyfikacji prostych, niskopoligonowych, ręcznie tworzonych modeli przedstawiających kolumny i domy jednorodzinne. Modele oznaczono dodatkowo wartościami oceniającymi ich styl, estetykę i funkcjonalność. Wytrenowany algorytm z jednej strony może bezpośrednio oceniać klasę i zmienne wartościujące model, ale z drugiej strony, propagowany wstecz może służyć eksploracji przestrzeni projektowej i generowaniu nowych modeli spełniających zadane wymagania. Niestety, interpretacja osiągniętych przez del Campo i in. dosyć abstrakcyjnych wyników jest trudna ze względu na zastosowane przez autorów agresywne przekształcenia służące zwiększeniu liczby danych treningowych, które utrudniają rozpoznanie poszczególnych klas już w zbiorze treningowym (del Campo, i in., 2020).

Przetwarzaniem danych jest także wizualizacja projektów architektonicznych. Sieci neuronowe mogą być wykorzystane do tworzenia fotorealistycznych lub stylizowanych wizualizacji na podstawie prostych, linearnych szkiców. Podstawowe możliwości wykorzystania transferu stylu w wizualizacji architektonicznej zaprezentowali Yick Hin Edwin Chan i A. Benjamin Spaeth z Uniwersytetu Cardiff. Autorzy zdecydowali się wytrenować system na ręcznie rysowanych, perspektywicznych szkicach budynków wolnostojących zestawionych z odpowiadającymi im fotografiami. Przez ręczne rysowanie przykładów treningowych sieć została wytrenowana jedynie na 84 obrazach, przez co wykazywała objawy przetrenowania (ang. overfitting) i miała problem z generalizacją na niespotkane w treningu przykłady (Chan i Spaeth, 2020). O wiele bardziej funkcjonalny, analogiczny system został stworzony przez Kyle’a Steinfelda z Uniwersytetu Kalifornii w Berkeley. Napotkane przez Chana i Spaetha problemy zostały rozwiązane poprzez wyspecjalizowanie algorytmu i zautomatyzowanie generowania przykładów treningowych. Algorytm został wyspecjalizowany jedynie do widoków aksonometrycznych o stałym nachyleniu kamery i jednorodnym oświetleniu, co znacząco ujednoliciło przykłady treningowe i uprościło problem aplikacji transferu stylu. Pary treningowe zostały wygenerowane automatycznie poprzez komputerowe renderowanie wizualizacji zbioru modeli trójwymiarowych w formie fotorealistycznej (z teksturowaniem i cieniowaniem) oraz w formie linearnej stylizowanej na rysunek odręczny. Podejście zautomatyzowane pozwoliło na wygenerowanie znacznie większego zbioru treningowego, co umożliwiło modelowi nauczenie się radzenia sobie z o wiele bardziej zróżnicowanymi geometriami i teksturami. Opracowany algorytm został zaimplementowany w postaci wtyczki do programu Photoshop i wykorzystany w trakcie zajęć projektowych ze studentami architektury (Rys. 15). Wytrenowane przez sieć neuronową aksonometryczne wizualizacje posłużyły studentom jako punkty wyjścia do dalszych prac projektowych (Steinfeld, 2020).

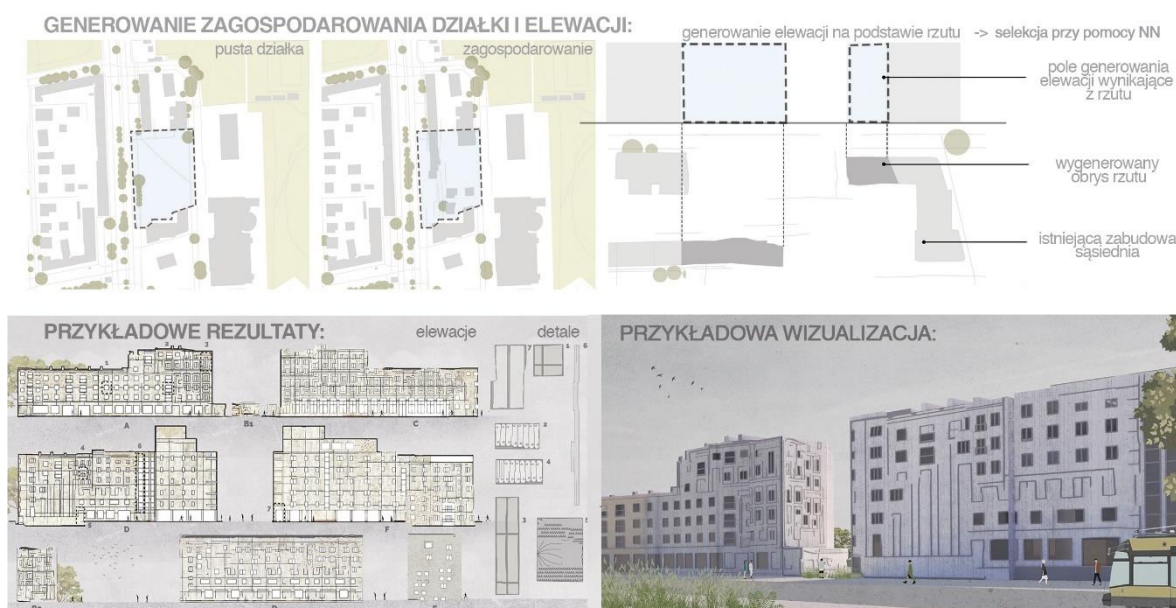


Rys. 15. System Sketch2Pix opracowany przez Kyle'a Steinfelda służący do przetwarzania linearnych szkiców aksonometrycznych na ich fotorealistyczne wizualizacje. Po lewej przykłady prostych modułów zwizualizowanych przez algorytm, po prawej dwa przykłady kolaży stworzonych przez studentów architektury korzystających z algorytmu Sketch2Pix w trakcie zajęć projektowych (Steinfeld, 2020).

Architektura: klasyfikacja

W większości dziedzin klasyfikacja, detekcja i segmentacja semantyczna są jednymi z popularniejszych zastosowań dla głębokiego uczenia maszynowego. Klasyfikacja jest stosowana także w problemach projektowych w skali architektonicznej. Najczęściej do klasyfikacji wykorzystywane są projekty architektoniczne przedstawione w formie rastrowych rzutów i przekrojów. Jednym z prostszych zastosowań jest sam problem rozpoznawania, czy dany rysunek architektoniczny przedstawia rzut czy przekrój. Algorytm zdolny do klasyfikacji typów rysunków technicznych mógłby być przydatny przy cyfryzacji papierowych archiwów architektonicznych. Zautomatyzowane sortowanie materiałów mogłoby zwiększyć efektywność procesu cyfryzacji dużych zbiorów rysunków. W 2019 roku Jennifer Mei Yee Ng i in. opracowali właśnie taki system wykorzystujący sieć neuronową do rozróżniania rzutów od przekrojów (Ng, i in., 2019). O krok dalej poszli Can Uzun i Meryem Birgül Çolakoğlu z Politechniki w Istambule, wykorzystując głębokie uczenie maszynowe nie tylko do klasyfikacji rzutów i przekrojów, ale i do ich detekcji, czyli dokładnej lokalizacji

na arkuszu i rozpoznawania ich obrysów. Detekcja pozwala na odpowiednie przycięcie arkusza, tak by rysunek przedstawiał jedynie rzut czy przekrój, bez elementów dodatkowych jak tabele czy opisy (Çolakoğlu i Uzun, 2019). Rzuty i przekroje można poddać także segmentacji semantycznej, czyli grupowaniu fragmentów obrazu wejściowego względem wybranych klas. Jednoznaczna segmentacja poszczególnych elementów rysunków architektonicznych, jak ściany, drzwi, okna i klatki schodowe, może być przydatna przykładowo przy automatycznej konwersji klasycznych, dwuwymiarowych rysunków CAD-owskich na modele trójwymiarowe lub BIM-owskie. System służący rozpoznawaniu elementów składowych rzutów architektonicznych przy pomocy segmentacji semantycznej został opracowany przez naukowców z Uniwersytetu Keio w Japonii (Xiao, i in., 2020). Na wyższym poziomie abstrakcji projektowej rzuty mogą być także dzielone względem funkcji poszczególnych pomieszczeń w budynku. Algorytmy takie zaproponowali między innymi Zheng i Huang (Zheng i Huang, 2018) oraz Brown i in. (Brown, i in., 2020). Oprócz rozpoznawania już zdefiniowanych w planie funkcji pomieszczeń algorytmy tego typu mogą służyć do sugerowania najlepszych funkcji przy projektach adaptacyjnych, a także pomóc przy badaniu cech rzutów skorelowanych z poszczególnymi funkcjami poprzez analizowanie aktywacji poszczególnych warstw sieci neuronowej (Brown, i in., 2020). Zespół z Politechniki Warszawskiej, którego byłem członkiem, wykorzystał klasyfikacyjne zdolności sieci neuronowych do ulepszenia zdolności generatywnych nieneuronowego algorytmu PatchMatch. W projekcie tym algorytm PatchMatch służący do bezszwowego uzupełniania brakujących fragmentów obrazów został przez nas wykorzystany do generowania zagospodarowania działki, projektów elewacji i detali plomby architektonicznej na podstawie otaczającej zabudowy (Rys. 16). Wygenerowane rezultaty były następnie klasyfikowane przez zestaw pięciu różnych sieci neuronowych. Pewność klasyfikacji danego elementu posłużyła do selekcji najlepszych przykładów wygenerowanych przez algorytm. Wyselekcjonowane przez sieci neuronowe rozwiązania zostały skompilowane do postaci projektu architektonicznego plomby architektonicznej (Dzieduszyński, i in., 2020).



Rys. 16. Projekt plomby architektonicznej wygenerowany przez algorytm PatchMatch wspomagany przez zespół pięciu klasyfikacyjnych sieci neuronowych dokonujących selekcji najlepszych elementów projektowych (Dzieduszyński, i in., 2020).

Klasyfikacji w skali architektonicznej można dokonywać także na danych wejściowych innego rodzaju. Zespół z German Research Center for Artificial Intelligence Uniwersytetu w Monachium przeprowadził szczegółowe badanie mające na celu sprawdzenie, które formy tensorowej, semantycznej, grafowej reprezentacji funkcjonalnych diagramów architektonicznych najlepiej sprawdzają się w problemach klasyfikacyjnych i które reprezentacje sprzyjają nabywaniu przez sieć neuronową zdolności do generalizowania. Przetestowano reprezentacje projektu w formie mapy wielowarstwowej (ang. multilayer map), w formie tekstu i w formie wektora w kodzie „jeden z n” (ang. one-hot vector). W testowanym problemie klasyfikacyjnym sieć neuronowa musiała klasyfikować poszczególne strefy funkcjonalne w mieszkaniach. Przy badanym zastosowaniu zdecydowanym zwycięzcą wśród badanych formatów okazało się kodowanie „jeden z n” (Eisenstadt, i in., 2021).

W dobie dostępności i popularności skanowania laserowego i fotogrametrii coraz częściej w projektowaniu wykorzystywane są chmury punktów. Stojanovic i in. wykorzystali splotowe sieci neuronowe do segmentacji semantycznej chmur punktów (np. rozpoznawanie obiektów we wnętrzach architektonicznych lub do ich „wzbogacania semantycznego” (ang. semantic enrichment) poprzez powiązanie poszczególnych punktów z odpowiadającymi im danymi, pochodzącymi z sensorów dokonujących pomiarów w badanej przestrzeni [np. pomiary

temperatury (Stojanovic, i in., 2019)]. Semantycznie pogrupowane chmury punktów mogą być w łatwiejszy sposób przetworzone na modele trójwymiarowe, służące do dalszych działań projektowych lub konserwatorskich (Croce, i in., 2021). W architektonicznej klasyfikacji typologicznej bardzo przydatne są także reprezentacje przestrzeni przedstawione w formie szeregu diagramów typu isovist¹⁵. Isovisty w przybliżeniu modelują percepcję przestrzeni przez znajdującego się w niej człowieka i mogą być skutecznie wykorzystane w klasyfikacji przez sieć neuronową przestrzeni wewnętrznej oraz zewnętrznej (Sedlmeier i Feld, 2018). Metoda ta pozwala na statystyczne próbkowanie percepcji przestrzeni w całym budynku i w jego otoczeniu, pozwalając na podejmowanie decyzji projektowych na podstawie danych statystycznych opisujących kompozycję przestrzenną projektu (Peng, i in., 2017). Użytecznym zbiorem danych mogą być także informacje zebrane przez sensory znajdujące się w telefonach komórkowych. Judit Tamas i Zsolt Toth przyjrzeni się problemowi nawigacji we wnętrzach architektonicznych i przetestowali różne algorytmy uczenia maszynowego w przewidywaniu klasy pomieszczenia na podstawie hybrydowych, multimodalnych danych sensorycznych zebranych przez system ILONA (Indoor Localisation and Navigation). Z testowanych algorytmów w klasyfikacji pomieszczeń najlepiej radziły sobie sieci neuronowe, „k najbliższych sąsiadów” (kNN, ang. k-Nearest Neighbors) oraz naiwne klasyfikatory bayesowskie (Tamas i Toth, 2018).

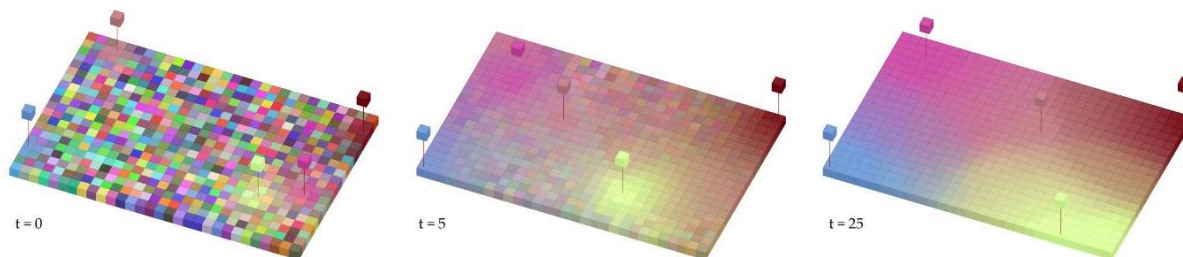
Klasyfikacji i segmentacji semantycznej można dokonywać także na fotografiach. Przykładem takiego zastosowania jest algorytm autorstwa Islama Zohiera, Ahmeda El Anabły’ego i Ahmeda S. Madaniego służący do oznaczania na zdjęciach minaretów, odróżniania ich od wież innego typu i segmentacji semantycznej względem stylu architektonicznego (Zohier, i in., 2020). W celu uzyskania zbioru danych treningowych dla sieci neuronowej dokonującej segmentacji semantycznej obiektów architektonicznych można wykorzystać obecnie dostępne narzędzia, służące do fotorealistycznego renderingu wizualizacji architektonicznych. Mohammad Alawadhi i Wei Yan zaproponowali ciekawe podejście do renderingu par treningowych z architektonicznych modeli BIM-owskich. Dzięki klasom

¹⁵ Diagramy isovist są dwuwymiarową lub trójwymiarową reprezentacją obwiedni przestrzeni dostępnej bez przeszkód z badanego punktu. Isovisty dwuwymiarowe stanowią wielokąt wyznaczony przez końce promieni wysyłanych z badanego punktu we wszystkich kierunkach, a kończących się na ograniczających przestrzeń przeszkodach (Tandy, 1967).

elementów architektonicznych wbudowanych w model BIM możliwy jest automatyczny rendering fotorealistycznej, perspektywicznej wizualizacji oraz odpowiadających jej masek semantycznych grupujących poszczególne kategorie na dodatkowych obrazach. Zautomatyzowane generowanie fotorealistycznych danych treningowych pozwala na efektywne wytrenowanie sieci neuronowej potrafiącej radzić sobie z problemem segmentacji semantycznej także na prawdziwych zdjęciach przedstawiających obiekty architektoniczne (Alawadhi i Yan, 2020).

Architektura: ekstrakcja i uczytnianie danych

W kategorii uczytniania i ekstrakcji danych architektonicznych sieci neuronowe pozwalają na domknięcie pętli sprzężenia zwrotnego w projektowaniu poprzez ekstrakcję informacji o rzeczywistym sposobie wykorzystywania przestrzeni zbudowanej przez użytkowników i przedstawienie ich projektantowi w uczytnionej, możliwej do interpretacji formie. Jens Jørgensen i in. wykorzystali splotowe sieci neuronowe do zbierania informacji o „sytuacjach behawioralnych” zachodzących w przestrzeni i rejestrowanych na surowym materiale wideo z monitoringu. Sytuację behawioralną w rozumieniu autorów stanowić może przykładowo jednoczesna detekcja w danym obszarze człowieka i krzesła oznaczająca ich wzajemną interakcję, czyli czynność siedzenia. Zebrane zachowania organizowane są przez algorytm w formie grafu integrującego użytkowników, obiekty, wyposażenie przestrzeni i ich wzajemne interakcje. System taki, dzięki implementacji neuronowego rozpoznawania obiektów bezpośrednio w module kamery, umożliwia zbieranie i przetwarzanie danych w sposób w pełni anonimowy (Jørgensen, i in., 2020). W projektowaniu parametrycznym przydatna może być także ekstrakcja parametrów istotnych z obszernej grupy parametrów pierwotnych. Algorytm autorstwa Johna Hardinga pozwala na analizę parametrów numerycznych definiujących model trójwymiarowy pod względem ich rzeczywistego wpływu na geometrię lub inne cechy modelu. Redukcja liczby wymiarów definiujących model (np. do formy powierzchni dwuwymiarowej, Rys. 17) pozwala na efektywniejszą eksplorację możliwych rozwiązań przy jednoczesnym zwiększeniu czytelności wpływu zmian wprowadzonych w parametrach na zmiany zachodzące w modelu (Harding, 2016).



Rys. 17. Proces treningu algorytmu służącego do redukcji liczby wymiarów definiujących projekt parametryczny stworzony przez Johna Hardinga. Na ilustracji przedstawiono proces redukcji parametrów z pięciu do dwóch, możliwych do zwizualizowania w postaci dwuwymiarowej płaszczyzny (Harding, 2016).

Sieci neuronowe pozwalają też na wydajniejsze przetwarzanie obszernych baz danych architektonicznych. System DANIEL opracowany przez Sharmę i in. służy do automatycznego przeszukiwania repozytoriów CAD-owskich rzutów architektonicznych w celu odnalezienia przykładów projektów, których układ przestrzenny i funkcjonalny jest zbliżony do zadanego rzutu. Możliwe zastosowania takiego systemu wydają się szerokie. Przykładowo, projekty znalezione przez DANIEL mogą służyć za inspirację do dalszych prac projektowych lub mogą być potencjalnymi pokrewnymi ofertami dla klientów na rynku nieruchomości. System może także być przydatny przy zarządzaniu obszernymi zbiorami projektów katalogowych, wyszukiwaniu historycznych projektów w dużych biurach architektonicznych, wykrywaniu plagiatów czy w rozstrzyganiu konkursów projektowych (Sharma, i in., 2017).

Architektura: optymalizacja

Ostatnią kategorią zastosowania sieci neuronowych w skali architektonicznej jest optymalizacja procesów. Danymi do treningu sieci neuronowych służących optymalizacji architektonicznej zazwyczaj są dane będące wynikami symulacji komputerowych, symulacji środowisk uczenia ze wzmacnianiem lub dane wygenerowane syntetycznie w sposób zautomatyzowany. Już w 1996 roku Jeffrey Ignatius Kindangen wykorzystał prostą, jednokierunkową sieć neuronową do optymalizacji naturalnej wentylacji w projektowanym budynku na podstawie danych symulacyjnych. Algorytm potrafił na uproszczonym modelu pomieszczenia modyfikować rozmiar i położenie okien w celu zapewnienia układu gwarantującego odpowiednie przewietrzanie pomieszczenia przy zadanym kącie natarcia wiatru (Kindangen, 1996). Sieci neuronowe mogą być wykorzystywane pomocniczo jako algorytmy optymalizacyjne w bardziej złożonych systemach generatywnych. Sevil Yazici

zastosował sieć neuronową do optymalizacji wydajności strukturalnej generowanego parametrycznie projektu dwupłaszczyznowych przekryć względem mierzonego symulacyjnie odkształcenia struktury poprzez dobór materiałów o odpowiednich właściwościach w różnych regionach struktury (Yazici, 2020). Jingyi Li i Hong Chen wytrenowali sieć neuronową w celu optymalizacji zmiennych projektowych w budynku biurowym względem jego wydajności energetycznej. Błędy przewidywanych przez algorytm wartości symulacji energetycznych w najgorszych przypadkach nie przekraczały 15%, a oparta na głębokim uczeniu maszynowym optymalizacja pozwoliła na znalezienie rozwiązań projektowych oferujących aż do 37,5% oszczędności energetycznych względem bazowego projektu zaproponowanego przez architekta (Li i Chen, 2021). Tyson Hosmer i Panagiotis Tigas w pracach nad zautomatyzowanymi, robotycznymi modułami konstrukcyjnymi w ustroju tensegrity¹⁶ wykorzystali sieci neuronowe jako algorytmy optymalizujące w symulacjach głębokiego uczenia ze wzmacnianiem. Algorytm sterujący ruchem i naprężeniem poszczególnych modułów tensegrity pozwala na tworzenie ustrojów zdolnych do samoorganizacji, dynamicznego i statycznego utrzymywania równowagi oraz rekonfiguracji przestrzennej. System wytrenowany jest w symulacji komputerowej, jednak skutecznie może być aplikowany także w fizycznym ustroju tensegrity. Dzięki zastosowaniu symulacji wieloagentowej liczne moduły mogą współpracować ze sobą, by osiągnąć zadany cel. Geometria jest optymalizowana przez sieć neuronową względem zadanej pozycji i parametrów strukturalnych ustroju (Hosmer i Tigas, 2019). Dzięki zdolnościom sieci neuronowych do przybliżania złożonych, nieliniowych funkcji, są one dobrymi kandydatami do rozwiązywania problemów optymalizacji wielokryterialnej, w której konieczna jest maksymalizacja kilku celów jednocześnie. W 2018 roku David Newton zaproponował system służący wielocelowej, jakościowej optymalizacji projektów architektonicznych (MOQO, ang. Multi-Objective Qualitative Optimization). W algorytmie tym poszczególne cele jakościowe (jak horyzontalność, modularność czy strzelistość obiektu) konwertowane były na podstawie badania ankietowego do postaci wartości numerycznej. Na podstawie tych danych wytrenowano sieć neuronową przypisującą zadany, zwokselizowanym modelom architektonicznym wynik dla każdego z celów jakościowych.

¹⁶ Tensegrity – (ang. tension + integrity; naprężenie + integralność) ustrój konstrukcyjny, w którym elementy przenoszące ściskanie są ze sobą połączone pośrednio poprzez ciągną przenoszące rozciąganie. Ustrój został opisany teoretycznie i opatentowany na początku lat 60. przez Buckminstera Fullera (Fuller, 1961; Fuller, 1962) i równolegle stosowany w instalacjach przestrzennych przez Kennetha Snelsona (Heartney, 2013).

Wynik ten służył następnie za metrykę dla potrzeb dalszej optymalizacji modelu (Newton, 2018).

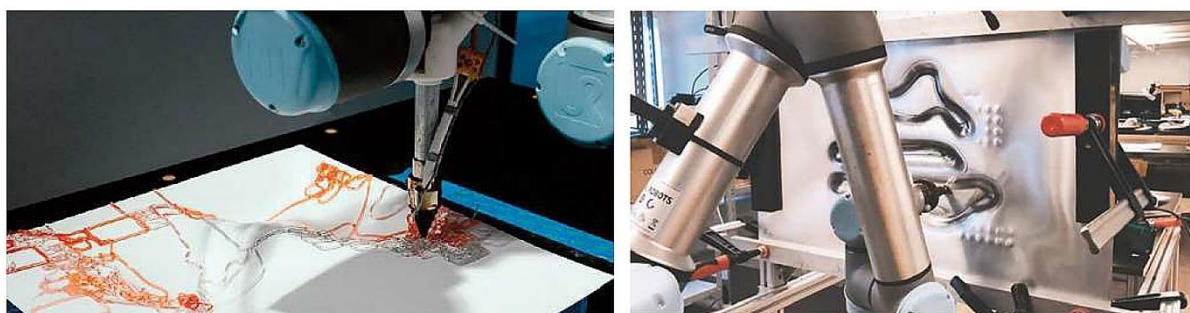
2.2.3. Detal, materiały i komputerowe wspomaganie wytwarzania (CAM) – zastosowania sieci neuronowych

Detal, materiały, CAM: ewaluacja i przetwarzanie danych

Głębokie uczenie maszynowe stosowane jest także w problemach dotyczących skali detalu architektonicznego, materiałów oraz wytwarzania wspomaganego komputerowo. Przy ewaluacji i przetwarzaniu danych sieci neuronowe mogą operować na zmiennych liczbowych przedstawionych w formie tabularycznej, ręcznie oznaczanych modelach parametrycznych, szeregach czasowych, danych symulacyjnych czy fotografiach. Zespół pod kierownictwem Any Silvy z Politechniki w Lizbonie od 2011 roku bada możliwości wykorzystania maszynowych metod stochastycznych i sieci neuronowych w modelowaniu przewidywanej żywotności materiałów elewacyjnych. W zaproponowanym podejściu dane wejściowe stanowi zbiór numerycznych zmiennych charakteryzujących stan materiałów elewacyjnych. W przypadku paneli kamiennych, zastosowanych do wykończenia elewacji, zmienne te opisują cechy materiałowe, jak typ wykończenia, wiek, rodzaj kamienia, wielkość paneli, a także cechy środowiskowe, jak ekspozycję i odległość od morza. Na podstawie tych danych algorytm ocenia przewidywane poziomy degradacji elewacji w skali od zera do stu lat (Silva, i in., 2011). Zespół opracował zbliżony system służący przewidywaniu żywotności powierzchni malowanych. Podobnie jak w przypadku elewacji kamiennych, cechy wejściowe dla sieci neuronowej opisywały wiek, orientację oraz dystans od morza, ale dodatkowo także wielkość i wysokość budynku, kolor i typ farby, odległość od źródeł zanieczyszczeń, ekspozycję na wilgoć, funkcję budynku i teksturę powierzchni. W przypadku powierzchni malowanych algorytm jest w stanie trafnie oceniać żywotność wykończenia na jedenaście lat w przód (Dias, i in., 2013).

Współcześnie bardzo aktualnym tematem są aktywne i inteligentne materiały architektoniczne. Głębokie uczenie maszynowe pozwala na zwiększenie kontroli nad wytwarzaniem materiałów tego typu i nad ich zachowaniem w trakcie eksploatacji. Daniil Koshelyuk i in. z IAAC w Barcelonie wykorzystali sieć neuronową do interpretacji zmian oporności prądu elektrycznego przepływającego przez membranę grafenową w celu przewidywania kąta ugięcia odkształconej powierzchni. Stworzony system pozwala na precyzyjne przybliżanie

prawdopodobnej formy geometrycznej aktywnego materiału poddanego działaniu sił zewnętrznych, umożliwiając efektywniejszy nadzór nad poddanym deformacji elementem (Koshelyuk, i in., 2019). Dan Luo, Jingsong Wang i Weiguo Xu stworzyli linearny elastomer o niejednorodnych przekrojach wzdłuż osi głównej. W zależności od wykorzystanej sekwencji przekrojów elastomer poddany obciążeniu przyjmuje różne formy geometryczne. Korzystając z ramienia robotycznego i danych symulacyjnych, autorzy elastomeru wytrenowali sieć neuronową przetwarzającą sekwencję przekrojów wzdłuż osi materiału na przewidywaną pod obciążeniem geometrię. Algorytm działa także w drugą stronę, umożliwiając przetworzenie zadanej geometrii docelowej na konieczną do uformowania sekwencję przekrojów materiału (Luo, i in., 2018). Gabriella Rossi i Paul Nicholas zastosowali sieci neuronowe do zwiększenia sztywności giętych, elewacyjnych paneli blaszanych. W ramach proponowanego procesu w pierwszej kolejności docelowa geometria panelu kodowana jest w formie obrazu, w którym poszczególne kanały opisują krzywiznę docelowego panelu i punkty mocowania do elewacji. Następnie na podstawie tych obrazów sieć neuronowa przewiduje naprężenie w panelu wywołane przez parcie wiatru i generuje ścieżki dla wydruku 3D wzmacniającego najbardziej obciążone regiony panelu (Rys. 18). Wzmocnienia mogą być także implementowane w postaci podłużnych żeber wytłaczanych przez tandem dwóch ramion robotycznych (po jednym na każdą stronę panelu) (Rossi i Nicholas, 2020).



Rys. 18. Wspomagane przez sieć neuronową metody fabrykacji giętych, blaszanych paneli elewacyjnych opracowane przez Gabriellę Rossi i Paula Nicholas. Po lewej – druk 3D wzmocnień wygenerowanych przez sieć neuronową w odpowiedzi na przewidywane obciążenia wywołane parciem wiatru. Po prawej – wytwarzanie podobnych wzmocnień w formie wytłoczonych w blasze podłużnych żeber (Rossi i Nicholas, 2020).

Ten sam zespół wytrenował sieć neuronową służącą do generowania instrukcji dla numerycznie sterowanego koła angielskiego w celu wytwarzania dwukrzywiznowych paneli blaszanych o zadanej geometrii. Sieć została wytrenowana na ręcznie przygotowanych parach docelowych geometrii i odpowiadających im ścieżek gięcia. Do tej pory koło angielskie zazwyczaj było traktowane jako narzędzie ręczne, przy pomocy którego ciężko jest osiągnąć powtarzalne, precyzyjne rezultaty. Sieci neuronowe pozwalają na dokładne przetwarzanie kształtu panelu na właściwe instrukcje dla koła angielskiego. Zaproponowana zautomatyzowana metoda jest jedną z tańszych opcji formowania elewacji wielokrzywiznowych (Rossi i Nicholas, 2018). Część z członków zespołu zajmującego się wspomaganiem przez głębokie uczenie maszynowe gięciem metalu zwróciła się w kierunku mniej sztywnych materiałów i przyjrzała się możliwościom wykorzystania sieci neuronowych w projektowaniu tkanych cyfrowo dzianin, służących za membrany osłaniające wnętrza przed słońcem. Stworzony przez Paula Nicholasa, Martina Tamke i in. prototyp przetwarza zadane warunki zacienienia na pliki produkcyjne dla maszyny dziewiarskiej. System umożliwia architektom bardziej intuicyjne projektowanie detalu membrany zacinającej poprzez definiowanie oczekiwanego efektu świetlnego w projektowanej przestrzeni bez konieczności zagłębiania się w szczegóły techniczne sztuki dziewiarskiej (Thomsen, i in., 2019). Sieci neuronowe mogą również przetwarzać projekty dzianych membran w celu osiągnięcia złożonych form geometrycznych. Poprzez modyfikację w pliku produkcyjnym poszczególnych typów ściegów w różnych rejonach płaszczyzny sieci neuronowe mogą kontrolować kształt, jaki osiągnie membrana po rozciągnięciu (Sinke, i in., 2021). Dechen Chen i in. przetestowali kilka różnych rodzajów sieci neuronowych w przewidywaniu geometrii wydruku 3D niepodpartego mostka (ang. bridging) w zależności od przekazanych drukarce 3D instrukcji w formie szeregów czasowych G-code. System został wytrenowany przy pomocy prostego algorytmu widzenia maszynowego, interpretującego geometrię krzywej, będącej efektem wydruku 3D drukarki sterowanej przez konkretną sekwencję instrukcji. W zadaniu przewidywania geometrii mostka na podstawie instrukcji najlepiej radziła sobie sieć typu LSTM (ang. Long Short-Term Memory, sieć z mechanizmem długiej pamięci krótkotrwałej). W zadaniu odwrotnym, polegającym na przewidywaniu koniecznych instrukcji na podstawie zadanej krzywej lepiej radziła sobie sieć jednokierunkowa (Chen, i in., 2019). Pradeep Devadass i in. użyli sieci neuronowych do mapowania relacji między parametrami wejściowego projektu mostu łukowego na instrukcje służące do wytworzenia drewnianych elementów składowych z uwzględnieniem ograniczeń procesu fabrykacji. Wygenerowane przez sieć polecenia sterują ramieniem

robotycznym przycinającym odpowiednio drewniane moduły na pile taśmowej. Sama geometria mostu optymalizowana jest przez osobny algorytm genetyczny (Devadass, i in., 2019).

Podobnie jak w skali architektonicznej, w skali detalu i w wytwarzaniu wspomagany komputerowo sieci neuronowe także są wykorzystywane do przybliżania wyników kosztownych czasowo i obliczeniowo symulacji komputerowych. Effima Giannapoulou i in. dzięki głębokiemu uczeniu maszynowemu opracowali cyfrową metodę projektowania cienkościennych, rozgałęzionych ustrojów powłokowych. Sieć neuronowa wytrenowana na parametrach wejściowych modeli parametrycznych została wykorzystana do przewidywania cech strukturalnych generowanego modelu. Algorytm działa o wiele szybciej niż pełna symulacja, jest wystarczająco precyzyjny do eksploracji przestrzeni projektowej na wczesnych etapach prac i potrafi skutecznie przetwarzać niespotkane w trakcie treningu wartości parametrów wejściowych (Giannopoulou, i in., 2019). W najmniejszej z analizowanych skal Runjia Tian, Yujie Wang i Onur Yüce Gün zaproponowali wykorzystanie sieci typu LSTM, zwykle stosowanych do przetwarzania języka naturalnego, do wspomagania generowania ergonomicznych podeszew w obuwiu na podstawie szeregów czasowych opisujących rozkład ciśnienia pod stopą w trakcie biegu i odpowiadające mu deformacje powierzchni podeszwy. Zastosowanie to wykracza w prawdzie poza zakres CAAD, jednak przykład ten bardzo dobrze pokazuje nieszablonowe możliwości przetwarzania danych projektowych przy pomocy głębokiego uczenia maszynowego. Sieć neuronowa została wykorzystana do konwersji wielowymiarowych danych wejściowych opisujących obciążenia podeszwy do formy sygnału jednowymiarowego. Na tej podstawie osobny algorytm generatywny definiował ażurową, elastyczną strukturę podeszwy, by w jak najbardziej ergonomiczny sposób przenosić siły dynamiczne w trakcie biegu. Podobna metodyka może być stosowana także w problemach projektowych innego typu, w którym podstawą do podejmowania decyzji projektowych są pochodzące z pomiarów szeregi czasowe (Tian i Gün, 2020). Ciekawą demonstrację zdolności sieci neuronowych do złożonego przetwarzania kontekstów projektowych na przykładzie detalu zaproponowali Jacek Markusiewicz i Ander Gortazar Balerdi. W czasie warsztatów projektowych ze studentami autorzy opracowali algorytm głębokiego uczenia maszynowego rozpoznającego, czy dany projekt parametryczny krzesła jest projektem oryginalnym, czy plagiatem krzesła DCW Cahrlesa i Raya Eamesa z lat 40. dwudziestego wieku. Stworzenie algorytmu zostało zainspirowane przez krzesło Loti, bardzo zbliżone do krzesła DCW, popularne w Kraju Basków w latach 50. i 60. W celu

uchwycenia rozmytej granicy między plagiatem a projektem oryginalnym zorganizowano parametryczne warsztaty projektowe. Wygenerowane modele zostały przy pomocy ankiety ocenione pod kątem „stopnia naśladownictwa”. Na podstawie zebranych danych wytrenowano sieć neuronową potrafiącą decydować, czy dane krzesło jest plagiatem, czy nie (Markusiewicz i Balerdi, 2020).

Detal, materiały, CAM: algorytmy generatywne

Rozwiązania głębokiego uczenia maszynowego są implementowane w różnorodnych algorytmach generatywnych. Wśród przykładów w mniejszych skalach Jose Luis Garcia del Castillo y López zademonstrował system sterowania ramieniem robotycznym, służącym do cyfrowej fabrykacji, zdolnym odczytywać intencje człowieka w celu dokończenia rozpoczętego przez niego procesu. W zaprezentowanej implementacji sieć neuronowa rozpoznawała fragment szkicu narysowanego przez człowieka i proponowała jego uzupełnienie. Jeśli użytkownik zaakceptował propozycję algorytmu, ramię robotyczne kończyło flamastrem rozpoczęty szkic (del Castillo y López, 2019). Kacper Radziszewski i Jan Cudzik wytrenowali sieć neuronową na przykładach kapiteli kolumn korynckich przedstawionych w postaci zrasteryzowanych map głębi. Wytrenowana sieć potrafiła generować nowe, oryginalne przykłady kapiteli inspirowanych przez kapitele korynckie (Cudzik i Radziszewski, 2018). Mostafa W. Alani i Bilal R. Al-Kaseem zebrali zestaw przykładów heksagonalnych rozet charakterystycznych dla architektury islamskiej z okresu od dziewiątego do piętnastego wieku. Zebrane dane posłużyły przeciwstawnej sieci neuronowej za zbiór treningowy. Algorytm został wykorzystany przez autorów do wygenerowania projektów współczesnych rozet opartych na przykładach historycznych (Alani i Al-Kaseem, 2021).

W trochę większej skali Zandavali i Garcia skorzystali z sieci przeciwstawnych przy rozwiązywaniu problemu generowania wzorów służących do pisania instrukcji dla ramienia robotycznego odpowiedzialnego za układanie cegieł. Opracowana metoda brała pod uwagę kolejność układania cegieł oraz siły oddziałujące na strukturę w trakcie budowy, dzięki czemu możliwe było kształtowanie także delikatnych nadwieszów (Zandavali i Garcia, 2019). Yetkin i Sorguç użyli głębokiego uczenia maszynowego do eksploracji przestrzeni projektowej możliwych do wygenerowania struktur kratownicowych. Sieć neuronowa została wytrenowana do tworzenia propozycji kratownic na podstawie zadanych wymagań dotyczących przemieszczenia pod obciążeniem i pożądanej masy konstrukcji (Yetkin i Sorguç, 2019). Zheng

i in. stworzyli algorytm do generowania kielichowych, żebrowych konstrukcji stropowych możliwych do fabrykacji przy pomocy druku 3D. Forma otrzymanych projektów stropów jest efektem neuronowej optymalizacji sił wynikających z ciężaru struktury (Zheng, i in., 2020).

Detal, materiały, CAM: optymalizacja

Niektóre z zastosowań wymienionych w poprzednich sekcjach mogłoby być traktowanych pośrednio jako zadania optymalizacyjne, jednak warto wspomnieć osobno o aplikacjach sieci neuronowych, w których optymalizacja procesów jest bezpośrednim celem końcowym wykorzystania głębokiego uczenia maszynowego. W badaniach Gabreilli Rossi i Paula Nicholasa nad algorytmami sterującymi dla robotów wyposażonych w sensory haptyczne, użyto sieci neuronowych do optymalizacji generowania ścieżek, w sposób umożliwiając robotowi reagowanie na przeszkody obecne w jego bezpośrednim otoczeniu. Celem optymalizacji było proponowanie takich instrukcji, które jednocześnie zapewniają najpłynniejszy ruch i minimalizują ryzyko kolizji z przeszkodami. Sieć neuronowa była więc zastosowana do optymalizacji wielocelowej. Sterowany na podobnej zasadzie robot uniwersalny może być wdrażany w procesach wytwarzania wspomagane komputerowo w chaotycznych środowiskach, w których system musi dostosowywać się np. do ludzi znajdujących się w przestrzeni roboczej i powinien odpowiednio reagować na sytuacje nieprzewidywalne (Rossi i Nicholas, 2019). Podobną rolę odgrywają sieci neuronowe w proponowanym przez Zhihao Fanga i in. ramowym projekcie wieloagentowego systemu służącego budowie struktur ceglanych przez rój latających dronów. W proponowanym systemie głębokie uczenie maszynowe również odpowiada za optymalizację wielocelową trajektorii lotu dronów, tak by minimalizować ryzyko kolizji i maksymalizować trajektorie zbliżające drony do ich indywidualnych celów (Fang, i in., 2020).

Detal, materiały, CAM: klasyfikacja

W projektowaniu architektonicznym wspomagany komputerowo w skali detalu sieci neuronowe nie są często wykorzystywane w klasyfikacji. W projekcie „Lace Wall” autorstwa Martina Tamke, Mateusza Zwierzyckiego, Andersa Holdena Deleurana i Yuliya’i Šinke Baranovskaya’ey sieć neuronowa została wykorzystana jako część składowa bardziej złożonego systemu przestrzennego (Rys. 19). Głębokie uczenie maszynowe posłużyło do eksploracji przestrzeni projektowej poprzez klasyfikację i grupowanie możliwych do uzyskania form instalacji względem obciążeń poszczególnych modułów w rozpatrywanych sytuacjach. W „Lace Wall” sieć neuronowa była odpowiedzialna za identyfikację modułów

poddanych działaniom warunków brzegowych, w których ze względu na deformacje konieczne było przeprowadzenie dalszej optymalizacji. Sklasyfikowane przez sieć neuronową moduły przetwarzane były ponownie przez algorytm genetyczny, aż do osiągnięcia optymalnego rozwiązania mieszczącego się w zadanych zakresach (Tamke, i in., 2017).



Rys. 19. Instalacja przestrzenna „Lace Wall” autorstwa Tamke, Zwierzyckiego i in. W „Lace Wall” sieci neuronowe zostały wykorzystane do klasyfikacji modułów, które ze względu na zbyt duże deformacje w skali mikro uniemożliwiały stworzenie docelowej formy w skali makro (Tamke, i in., 2017).

Detal, materiały, CAM: przewidywanie przyszłości

W trakcie badań literaturowych trafiłem na jedno zastosowanie w skali detalu urbanistycznego, w którym sieci neuronowe zostały wykorzystywane do krótkoterminowego przewidywania przyszłości. Jiaxu Wu i in. opracowali metodę przewidywania przyszłych trajektorii ruchu poszczególnych pieszych w przestrzeni miejskiej na podstawie monitoringu wideo. Możliwość dynamicznej ekstrapolacji ruchu pieszego w czasie rzeczywistym może być przydatna przy projektowaniu szczegółowych rozwiązań z zakresu Smart City i ulic aktywnych dynamicznie reagujących na zachowanie poszczególnych użytkowników w celu zwiększenia bezpieczeństwa lub komfortu korzystania z przestrzeni miejskiej (Wu, i in., 2019). W skali długoterminowej do przewidywania przyszłości można zaliczyć także omówiony wcześniej w kategorii ewaluacji i przetwarzania danych algorytm przewidujący żywotność materiałów

wykończeniowych fasad (Dias, i in., 2013). W publikacji tej autorzy przewidywali stopień zużycia elewacji na podstawie szeregu czynników obejmujących lokalizację, typ wykończenia, wiek czy dystans od morza.

2.3. Luka badawcza – neuronowe przetwarzanie kontekstów kompozycyjnych, przesłanki

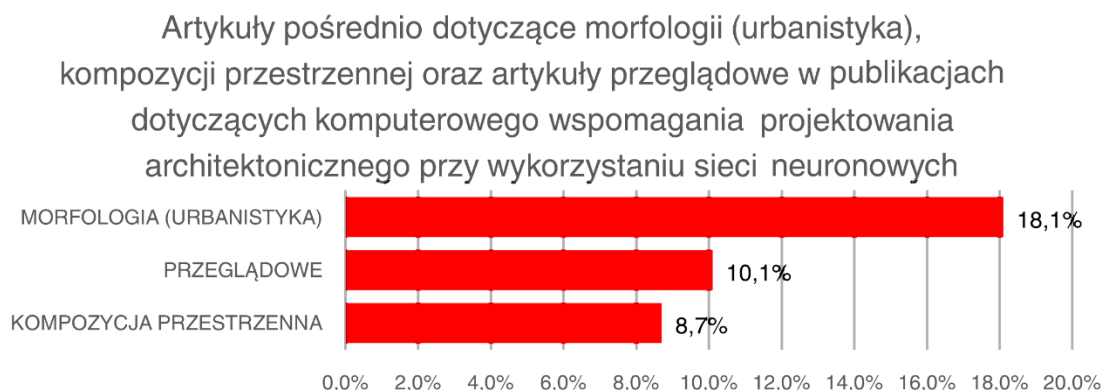
Jednym z zaskakujących wniosków płynących z kwerendy jest bardzo mała liczba publikacji dotyczących kompozycji przestrzennej. Analizy kompozycyjne przeprowadzane są praktycznie we wszystkich skalach projektowo-planistycznych. We wszystkich tych skalach dysponujemy też wieloma danymi typu big-data, zawierającymi w sobie wartościowe informacje o kompozycji przestrzennej. Zaliczają się do nich między innymi bazy projektów architektonicznych (bazy projektów archiwalnych¹⁷, projekty składane w urzędach, projekty konkursowe, bazy BIM-owskie), cyfrowe reprezentacje przestrzeni miast (bazy GIS, mapy, modele Digital Surface Model, zdjęcia satelitarne, geolokalizowane media społecznościowe, sensory smart-city), czy nawet zdigitalizowane dzieła sztuki (reprodukcje cyfrowe, skany fotogrametryczne, skany laserowe). Konteksty kompozycyjne często są rozległe, złożone, logicznie rozmyte i trudne do oprogramowania metodami klasycznymi. Analizy przestrzenne wymagają dużo czasu i trzeba je w pewnym stopniu przeprowadzać zarówno przy projektowaniu architektonicznym, planowaniu urbanistycznym i przy zadaniach administracyjnych. Badacze poszukują ostatnio możliwości automatyzacji części analiz przestrzennych. Anna Jachimowicz z Politechniki Warszawskiej zauważa, że ze względu na ciągły charakter planowania przestrzennego analizy nie powinny być przeprowadzane jednorazowo. Automatyzacja procesów analitycznych oferuje możliwość wielokrotnego przetwarzania zmieniających się uwarunkowań, co pozwala zaoszczędzić czas i zasoby biur projektowych oraz ułatwia uchwycenie procesów ciągłych, dla których pojedynczy przekrój danych w jednej chwili jest niewystarczający (Jachimowicz, 2022; Jachimowicz, 2023). Sieci neuronowe pozwalają na automatyzację części procesów, które dotychczas były wykonywane przez ludzi, umożliwiając przetworzenie dużych zbiorów danych w krótkim czasie.

¹⁷ Szczególnie ciekawą polską bazą projektów archiwalnych jest cyfrowa kolekcja Muzeum Architektury we Wrocławiu. Obejmuje ona kilka tysięcy zdigitalizowanych rysunków projektowych z okresu międzywojennego i powojennego w formie rastrowej (Muzeum Architektury we Wrocławiu, 2019).

Coraz częściej mówi się o zastępowaniu analiz eksperckich przez narzędzia cyfrowe. Czy jednak tak skomplikowany temat, jak kompozycja przestrzenna, może być uchwycony przez sieć neuronową? Sieci neuronowe już od lat biegle radzą sobie z klasyfikacją obrazu [np. AlexNet (Krizhevsky, i in., 2012)], jego syntetyzowaniem [Dall-E (Ramesh, i in., 2021), Dall-E 2 (Ramesh, i in., 2022)] czy uzupełnianiem brakujących fragmentów [ImageGPT (Chen, i in., 2020), Stable Diffusion (Rombach, i in., 2022)]. Zadania te wymagają praktycznego operowania zasadami kompozycyjnymi i pewnego „rozumienia” kompozycji przestrzennej jako całości. Mikołaj Małkiński i Jacek Mańdziuk przeprowadzili w 2022 roku badanie przeglądowe, w którym bardzo szeroko przeanalizowali skuteczność różnych algorytmów uczenia maszynowego (w tym sieci neuronowych) w abstrakcyjnym rozumowaniu wizualnym polegającym na rozwiązywaniu testów matryc Ravena¹⁸. Wiele z badanych sieci neuronowych osiągało wyniki zbliżone do tych osiąganych przez człowieka, a niektóre były od ludzi zdecydowanie lepsze (Małkiński i Mańdziuk, 2022). Skuteczność sieci neuronowych w tego typu testach jest silną przesłanką za postawieniem przypuszczenia, że współczesne sieci neuronowe są wystarczająco potężne, by móc wykorzystać je do rozwiązywania rzeczywistych problemów kompozycyjnych. Nawet jeśli sieci neuronowe nie będą w stanie zastąpić eksperta mogą być wykorzystane jako narzędzia wspomagające podejmowanie decyzji lub posłużyć w procesie projektowym jako inspiracja.

Ze 149 przeanalizowanych w kwerendzie prób wykorzystania sieci neuronowych w projektowaniu wspomaganym komputerowo 27 z nich dotyczyło morfologii i typologii miast, a jedynie 13 pośrednio odnosiło się do kompozycji przestrzennej (Rys. 20). W rozwiązaniach tych kompozycja przestrzenna traktowana była zazwyczaj jako element składowy większych problemów, a zagadnienia kompozycyjne nie były analizowane w oddzieleniu od innych zmiennych. Wyjątkiem jest praca Briana Ho opisana w publikacji *Making a New City Image*, ale zakres rozważań kompozycyjnych przeprowadzonych przez autora był niestety dosyć ubogi i sprowadził się jedynie do klasyfikacji „dzielnic” i „nie-dzielnic” w miastach na podstawie kolaży fotograficznych i odpowiadających im schwarzplanów (Ho, 2020).

¹⁸ Test matryc Ravena – ang. Raven’s Progressive Matrices, test inteligencji opracowany przez Johna C. Ravena w 1939 roku, polegający na przewidywaniu kompozycji wizualnej poprawnie wpisującej się w zadaną macierz kompozycji nadających kontekst (Raven, 2000).



Rys. 20. Artykuły pośrednio dotyczące morfologii urbanistycznej, kompozycji przestrzennej oraz artykuły przeglądowe w publikacjach dotyczących komputerowego wspomagania projektowania architektonicznego przy wykorzystaniu sieci neuronowych. Opracowanie własne.

Podsumowując, przeprowadzona kwerenda ujawnia, że pomimo potencjału sieci neuronowych w rozwiązywaniu problemów pokrewnych do kompozycji przestrzennej, istnieje zaskakująco mała liczba publikacji skupiających się bezpośrednio na tym zagadnieniu. Chociaż technologia głębokiego uczenia maszynowego zdaje się wystarczająco rozwinięta, by wspierać automatyzację analiz kompozycyjnych w różnych skalach projektowych i procesach urbanistycznych, temat ten pozostaje niedostatecznie eksplorowany w literaturze. Wyniki kwerendy sugerują potrzebę dalszych badań i rozwoju narzędzi cyfrowych w tej dziedzinie.

2.4. Punkty główne kompozycji przestrzennej

2.4.1. Elementy kompozycyjne i ich przetwarzanie maszynowe

Cyfrowe narzędzia służące rozpoznawaniu i przetwarzaniu kontekstów kompozycyjnych mogą być przydatne wszędzie tam, gdzie konieczne jest podejmowanie decyzji poprzez przesłanki wynikające z uwarunkowań kompozycyjnych. W myśl *Ustawy o planowaniu i zagospodarowaniu przestrzennym* uwarunkowania kompozycyjno-estetyczne bezpośrednio przekładają się na jakość przestrzeni, jej harmonię i są jednym z wyznaczników osiągnięcia ładu przestrzennego (Sejm RP, 2003). Ekstraktory zasad kompozycyjnych mogłyby ułatwić urzędom pracę przy sporządzaniu planów ogólnych gmin, planów rewitalizacji, tworzeniu miejscowych planów zagospodarowania przestrzennego, podejmowaniu decyzji o warunkach

zabudowy i formułowaniu dokumentów strategicznych. Uwarunkowania kompozycyjne mogą przekładać się na podział funkcjonalny miasta, a w szczególności na lokalizację ważnych funkcji publicznych. W skali urbanistycznej pracownie projektowe mogłyby wykorzystywać sieci neuronowe przy projektowaniu układów urbanistycznych wpisujących się w okoliczną zabudowę. Dzięki neuronowym ekstraktorom zasad kompozycyjnych deweloperom łatwiej byłoby identyfikować kompozycyjnie podkreślone lokalizacje nowych inwestycji, co przekłada się na ich atrakcyjność i wartość. Biura architektoniczne także zyskałyby możliwość zautomatyzowanego odczytywania uwarunkowań przestrzennych projektowanej inwestycji.

Dobrym punktem wyjścia do przeanalizowania możliwości aplikacji sieci neuronowych w problematyce kompozycji przestrzennej jest przyjrzenie się klasycznym elementom kompozycyjnym, opracowanym przez teoretyków oraz praktyków formy i kompozycji architektoniczno-urbanistycznej. Juliusz Żórawski w pracy *O Budowie Formy Architektonicznej* wyróżnia następujące elementy i aspekty kompozycji przekładające się na jej charakter, estetykę oraz na spoistość formalną:

- Części proste, z których składa się kompozycja. Odczytywane jako osobne elementy większych całości (s. 18).
- Grupy elementów, które same stanowią kompozycję punktów składowych, ale same stają się punktami w kompozycji bardziej złożonej (s. 28).
- Punkty główne oraz punkty ważne umieszczone w miejscach formalnie podkreślonych, przez co wybijają się spośród innych punktów (s. 48, 118).
- Dominanty, czyli przyciągające uwagę formy silne (s. 28).
- Liczbę elementów, które daje się percepcyjnie od siebie odróżnić i policzyć w liczbie od jednego do pięciu, a które powyżej tej liczby stanowią niepoliczalną grupę „wielu” (s. 27).
- Położenie elementu, który może być osadzony w kompozycji spójnie lub rzucony bardziej swobodnie (s. 90).
- Osiowość i powiązaną z nią symetrię osiową (s. 36, 79).
- Kąty ustanowione przez osie, które, jeśli są ortogonalne, sprzyjają spoistości formy (s. 79).
- Rytm, który musi mieć więcej niż 3 lub 4 elementy składowe (w zależności od otaczającego kontekstu), ponieważ inaczej odczytywany jest jedynie jako pewien układ osiowy (s. 62, 74, 110).

- Pole działania formalnego oznaczające obszar oddziaływania elementu w kompozycji (s. 103, 117).
- Wytyczną generalną, czyli główną zasadę kompozycyjną decydującą między innymi o swobodności lub spoistości układu (s. 120).

Oprócz powyższych Żórawski wyróżnia także powiązane z nimi rodzaje form przestrzennych:

- Formy oraz otoczenie stanowiące dla nich tło. Tło także może się składać z form, jednak w zestawieniu z formą silniejszą o mocnym konturze zostają one percepcyjnie zepchnięte do roli tła (s. 66).
- Formy spoiste (uporządkowane, czytelne, harmonijne) oraz formy swobodne (bezładne, z nieczytelną wytyczną generalną, organiczne) (s. 23).
- Silniejsze formy matki i podporządkowane im formy składowe (s. 30).
- Formy zależne od części i formy od nich niezależne (s. 31).
- Ograniczone formy skończone i otwarte formy nieskończone (Żórawski, 1962, s. 101).

W książce *Architecture Form, Space, & Order* Francis D.K. Ching trochę szerzej niż Żórawski wyróżnia więcej podstawowych elementów kompozycyjnych i podaje rodzaje złożonych z nich kompozycji. Ching proponuje następujący podział typologiczny kompozycji przestrzennych:

- Kompozycje oparte na siatkach regularnych (s. 66, 230).
- Kompozycje oparte na siatkach nieregularnych (s. 72).
- Kompozycje centralne (s. 198).
- Kompozycje radialne (s. 216).
- Kompozycje liniowe (s. 206).
- Kompozycje klastrowe (s. 222).
- Kompozycje hierarchiczne, w których jedne elementy są ważniejsze od innych (s. 212).

W obrębie tych typów kompozycyjnych Ching wyróżnia następujące elementy oraz zasady ich porządkowania:

- Punkty i centra, czyli percepcyjnie nierozdzielane elementy podstawowe odczytywane jako obiekty bezwymiarowe (s. 4, 216).
- Płaszczyzny odczytywane jako podstawowe elementy dwuwymiarowe (s. 14).
- Objętości odczytywane jako podstawowe elementy trójwymiarowe (s. 14).
- Dominanty podkreślone przez własną formę lub umiejscowienie (s. 206).

- *Data* (l. poj. *Datum*), czyli elementy odniesienia zbierające i organizujące kompozycję w całość podobnie do pięciolinii w zapisie nutowym (s. 366).
- Grupy elementów tworzące większe całości. Już dwa punkty zaczynają wyznaczać między sobą pochodną oś (s. 6).
- Linie, osie i naprowadzenia, czyli elementy podstawowe o jednym z wymiarów zdecydowanie większym od pozostałych lub grupy elementów ułożonych liniowo (s. 8, 62, 206, 242, 340).
- Symetrie osiowe lub punktowe (s. 222, 348).
- Rytm proste, niejednostajne lub progresywnie zmienne (s. 382).
- Podobieństwa części odczytywane jako pokrewieństwo fragmentów kompozycji (s. 222).
- Klastry obejmujące wiele wyróżniających się z otoczenia elementów podstawowych tworzących percepcyjnie czytelną całość (s. 68).
- Skale oznaczające rozmiary elementów lub wizualne postrzeganie ich rozmiarów zależne od położenia obserwatora (s. 293, 330, 333).
- Proporcje będące wzajemnymi relacjami między rozmiarami elementów (s. 29).
- Hierarchie według rozmiarów, form czy lokalizacji (s. 358).
- Transformacje, oznaczające przetworzenie kompozycji lub ich części zachowujące pierwotne znaczenie (s. 402)..
- Krawędzie, czyli liniowe granice formy (s. 82).
- Narożniki będące załamaniami krawędzi i punktowymi ograniczeniami formy (s. 82).
- Pełnie, czyli trójwymiarowe objętości ograniczające pustą przestrzeń (s. 96).
- Pustki, czyli przestrzenie między elementami pełnymi (s. 96).
- Pętle oznaczające ciągle obrysy biegnące przez kompozycję lub dookoła niej (s. 222).
- Ścieżki stanowiące sekwencyjny, linearny plan poruszania się użytkownika w przestrzeni (s. 264).
- Zakończenia osi, ścieżki lub rytmu (s. 278).
- Wejścia i otwarcia wyznaczające początek nowej ścieżki (Ching, 2007, s. 250, 283).

W *Elementach Kompozycji Urbanistycznej* Kazimierz Wejchert zwraca uwagę głównie na kompozycję tkanki miejskiej i wyróżnia:

- Elementy krystalizujące, wokół których narasta reszta kompozycji (s. 52).

- Ulice i drogi, czyli liniowe elementy przeznaczone do ruchu użytkowników przestrzeni (s. 56).
- Rejony wyróżniające się pod jakimś względem od innych obszarów (s. 59).
- Linie i pasma graniczne będące wyraźnymi ograniczeniami przestrzeni lub stanowiące przerwy między poszczególnymi rejonami (s. 64).
- Dominanty układu przestrzennego, czyli elementy o dużym znaczeniu (s. 67).
- Wybitne elementy krajobrazu będące swoistą wizytówką danej przestrzeni (s. 71).
- Punkty węzłowe, w których stykają się elementy przestrzenne o różnych strukturach (s. 71).
- Znaki szczególne unikalne dla konkretnej przestrzeni i wyróżniające ją (Wejchert, 1984, s. 74, 90).

W odróżnieniu od Wejcherta, Kevin Lynch pisze o mentalnych, dwuwymiarowych mapach przestrzeni miejskiej. W *The Image of the City* opisuje następujące elementy kompozycyjne:

- Drogi, czyli linie, wzdłuż których przemieszcza się użytkownik przestrzeni (s. 49).
- Węzły, czyli miejsca przecinania się elementów liniowych (s. 72).
- Charakterystyczne elementy krajobrazu, czyli punkty orientacyjne, na podstawie których użytkownicy określają swoje położenie (s. 78).
- Rejony, czyli elementy powierzchniowe (s. 66).
- Krawędzie rozumiane jako linie rozgraniczające obszary oraz bariery i nieciągłości przestrzeni (Lynch, 1960, s. 62).

Wymienione powyżej elementy kompozycyjne, zasady ich porządkowania oraz typy kompozycji przestrzennych wydają się możliwe do przedstawienia maszynowego. Modele sieci neuronowych wykazujące przestrzenne uprzedzenie dotyczące struktury danych (Mitchell, 2017), takich jak sieci CNN, mogą dobrze radzić sobie z przetwarzaniem elementów kompozycyjnych (Dzieduszyński, 2022b) charakteryzowanych przez ich relacje geometryczne (punkty główne, osie kompozycyjne, osie symetrii, rejony, klastry, rozmiary, proporcje, etc.). Modele sekwencyjne takie jak RNN, LSTM czy transformery można wykorzystać do przetwarzania elementów uporządkowanych sekwencyjnie w pewne logiczne ciągi (ścieżki, drogi, rytmy, hierarchie lub transformacje układu na osi czasu).

2.4.2. Punkty główne w teorii kompozycji przestrzennej

Jednym z ważniejszych elementów kompozycyjnych, na które zwracają uwagę teoretycy, są punkty główne i punkty formalnie podkreślone. Ich waga wynika z psychosomatycznych uwarunkowań człowieka, który przyciągany jest przez podkreślone kompozycyjnie, dominujące formy. Zjawisko to trafnie podsumowuje w swojej pracy o kształtowaniu panoram miejskich Karolina Sobczyńska:

„[W] (...) obserwatorze przestrzeni miasta, powstają wrażenia ogólne pochodzące od całości obrazu, ale też następuje rejestracja cząstkowa niektórych elementów, po których przesuwają się jego wzrok. Dominujące doznanie powstaje w momencie, kiedy wzrok człowieka zatrzymuje się na formie architektonicznej, która w jakiś sposób wyróżnia się od pozostałych form lub formie, która stanowi główny punkt kompozycji przestrzennej. (...) Tendencją człowieka obserwującego jakąś formę czy kompozycję z pewnej odległości jest poszukiwanie punktu głównego, dominanty układu.”
(Sobczyńska, 2021, s. 64)

Według Żórawskiego punkty główne to ważne punkty kompozycji przestrzennej umieszczone w miejscach formalnie podkreślonych. Wybijają się one na pierwszy plan spośród innych punktów i są potencjalnym miejscem dla umieszczenia dominanty. Oprócz punktów głównych są także punkty ważne, które także są formalnie podkreślone, jednak nie zawsze stanowią dla kompozycji pojedynczy punkt główny. W punkcie formalnie podkreślonym niekoniecznie musi znajdować się jakiś obiekt architektoniczny. Punkt główny może być pustką, przejściem, potencjałem dla pojawienia się czegoś lub kogoś. Żórawski odróżnia także od punktów głównych jądra kompozycji, których ważność nie wynika z kompozycji formalnej, a jest uwarunkowana przez ich znaczenie funkcjonalne. Przykładowo jądrem kompozycji w piramidzie jest ukryta komora grobowa, a podkreślony formalnie jest jej szczyt. Zespół jądra nie musi być formalnie podkreślony przez punkt główny, ale gdy tak się dzieje, sprzyja to spójności formy (Żórawski, 1962).

Ching także pośrednio nawiązuje do idei punktów głównych, jednak nie nazywa ich wprost. Mówi o dominantach ze względu na umiejscowienie, wyznaczonych przykładowo przez zakończenie założenia osiowego czy centralność punktu. Drugorzędnymi punktami formalnie podkreślonymi są także różne akcenty wizualne. Formalnie podkreślone bywają narożniki, jednak nie każdy narożnik ma wystarczającą wagę, by od razu uzyskać rangę punktu głównego.

Ching zwraca uwagę także na punkty wyznaczone przez węzły założeń sieciowych oraz zakończenia i rozpoczęcia ważnych dla użytkowników przestrzeni ścieżek (Ching, 2007).

Wejchert mówi o dwóch kategoriach elementów kompozycyjnych, wpisujących się w kategorię głównych punktów formalnie podkreślonych. Pierwszymi są dominanty układu przestrzennego wyróżniające się w skali architektonicznej ze względu na swoje umiejscowienie w kompozycji miejskiej, unikalność oraz topografię terenu (np. jako punkty najwyższe). Dominanty mogą także być podkreślone przez naprowadzenia widokowe, otwarcia szczelinowe, otwarcia wąskie lub kierunkowe. Druga kategoria to znaki szczególne, czyli punkty w mieście o nieco mniejszej wadze, które, mimo że nie są dominantami, zapisują się w mapie mentalnej użytkowników miast ze względu na swoją unikalność (mogą to być przykładowo atrakcyjne rzeźby czy elewacje). Wejchert w kontekście komunikacyjnym wyróżnia także punkty węzłowe umożliwiające zmianę trybu transportu, które mają bardzo ważną pozycję w mapach mentalnych użytkowników miast, jednak niekoniecznie muszą stanowić formalnie podkreślone punkty główne (Wejchert, 1984). Kevin Lynch zwraca uwagę na punkty orientacyjne (ang. landmarks), które mogą mieć różne skale, ale niezależnie od rozmiarów są dobrze zapamiętywalne ze względu na znaczącą lokalizację w kompozycji przestrzennej. Punkty orientacyjne często są podkreślone przez ich wysokość i widoczność, przez co stanowią formy wyróżniające się z tła (Lynch, 1960).

Podsumowując, w kontekście przestrzeni miejskiej punkty główne to ważne geometrycznie punkty znajdujące się w miejscach formalnie podkreślonych. Podporządkowują one sobie całość okolicznej kompozycji przestrzennej. W pojedynczym założeniu może występować jeden punkt główny, kilka równorzędnych i najważniejszych punktów formalnie podkreślonych lub kompozycja może być takich punktów pozbawiona. W takim przypadku wszystkie elementy kompozycyjne znajdujące się w polu percepcji są równie ważne (Maryńczuk, 2012). Punkty główne mogą mieć zarówno charakter pozytywowý (znacząca forma wyróżniająca się z tła) lub negatywowý (pustka będąca potencjałem do pojawienia się czegoś). Punkt taki może więc stanowić dominantę przestrzenną, ale także być placem miejskim – miejscem dla aktywności publicznych. Jego umiejscowienie może być podkreślone przez kompozycję miejską oraz topografię terenu. Przykładowo, podkreślenie takie może mieć formę przecięcia czy zakończenia osi, lub założenia radialnego, kierującego uwagę na punkt centralny. Ze względu na swoją lokalizację, widoczność i znaczącą skalę punkty główne mogą pełnić funkcję punktów orientacyjnych.

2.4.3. Punkty główne spontaniczne i projektowane

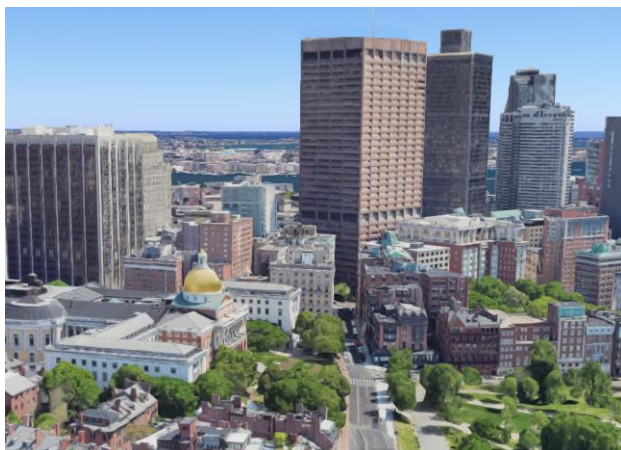
Punkty główne zazwyczaj krystalizują się w przestrzeni miejskiej spontanicznie na skutek nawarstwienia wielu, rozłożonych w czasie decyzji projektowych, planów i zdarzeń przypadkowych. Co więcej, poszczególne zespoły kompozycyjne zdefiniowane na różnych układach punktów formalnie podkreślonych wzajemnie się na siebie nakładają i nawarstwiają (Mironowicz i Ossowicz, 1997). Kompozycja przestrzenna jest fenomenem rozpiętym w czasie. Katarzyna Pluta zwraca uwagę, że „[r]ola kompozycji w kształtowaniu miasta polega na ochronie jego kulturowego oblicza z zastosowaniem historycznej kontynuacji” (Pluta, 2014, s. 79). Projektanci inspirowani zastaną topografią, układem komunikacyjnym, symbolicznym znaczeniem lokalizacji, czy wcześniejszą zabudową mogą intuicyjnie podkreślać kolejnymi interwencjami rangę danego miejsca zwiększając stopniowo jego wagę w kompozycji przestrzennej. Jak pisze w *Języku wzorców* Christopher Alexander o wzorcach formujących się w skali urbanistycznej:

„Te wzorce nigdy nie mogą być «zaprojektowane» czy «zbudowane» za jednym zamachem – wzrastają one po kawałku, są projektowane w sposób, w którym każde pojedyncze działanie zawsze pomaga w tworzeniu lub generowaniu wzorców globalnych(...)” [tłumaczenie własne] (Alexander, i in., 1977, s. XIX).

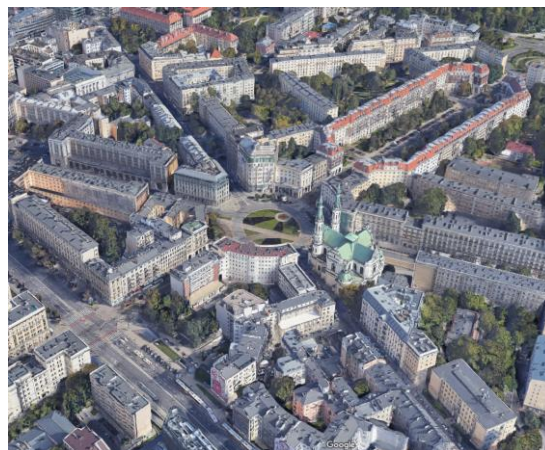
Spontaniczne wydarzenia formujące jedne punkty główne mogą doprowadzać jednocześnie do zatarcia pozycji innych punktów. Przykładem organicznie ewoluującego punktu głównego może być opisany przez Kevina Lyncha gmach State House w Bostonie. Miejsce to początkowo było podkreślone przez topografię terenu jako załamanie krawędzi wzgórza Beacon Hill. Dodatkowo jedna z głównych dróg bostońskich (obecnie Beacon Street), skręcając nagle wraz z układem terenu, wyznaczyła swoją ośią domagającą się zaakcentowania punkt. Ekspozycja widokowa przełożyła się na decyzję o zlokalizowaniu tam w 1798 roku zaprojektowanego przez Charlesa Bulfincha gmachu miejskiego. Powiązanie miejsca z ważną funkcją jeszcze bardziej zwiększyło jego rangę, przez co gmach był wielokrotnie rozbudowywany, a w 1874 roku jego szczyt zaakcentowano złotą kopułą. W 1960 roku, gdy Lynch prowadził swoje badania, złota kopuła wybijająca się na tle nieba, widoczna z każdego punktu Parku Boston Common i znajdująca się na zakończeniu kilku osi komunikacyjnych, zdecydowanie stanowiła punkt główny całego założenia urbanistycznego Beacon Hill (Lynch, 1960). W kolejnych latach jednak, z każdym ukończonym bostońskim wieżowcem, ranga State House stopniowo malała. Obecnie ginąca wśród betonowych i szklanych prostopadłościanów złota kopuła

stanowi punkt formalnie podkreślony jedynie w lokalnej skali narożnika parku Boston Common (Rys. 21).

O ile opisywane przez Alexandra wzorce społeczno-przestrzenne potrzebują czasu, by się w pełni uformować, to czysto formalnie rozumiane punkty główne mogą jednak także być tworzone przez celowe, pojedyncze decyzje projektowe. Świadomie projektując otwarcia widokowe i formując linie naprowadzające, architekt czy urbanista może tworzyć całościowe kompozycje rozpięte na planowanych punktach głównych. Przykładem może być ujazdowskie założenie placów gwiaździstych z lat 70. osiemnastego wieku, w którym centra placów stanowią równorzędne względem siebie węzły założenia sieciowego (Szwankowski, 1970). Osie wyznaczone wśród podwarszawskich pól (Zannoni, 1772; Aubert i Tardieu, 1788) stały się szkieletem pod jedno z najbardziej rozpoznawalnych warszawskich założeń urbanistycznych (Rys. 22).



Rys. 21. Złota kopuła State House w Bostonie ginąca na tle nowszych wieżowców, m.in. One Beacon Street z roku 1972 (fot. Google Maps, 2023)



Rys. 22. Widok na plac Zbawiciela będący punktem głównym założenia gwiaździstego (fot. Google Maps, 2023)

2.4.4. Punkty główne a skala kompozycji

Założenie ujazdowskie dobrze ilustruje kolejną cechę punktów głównych (i ogólnie kompozycji przestrzennych), mianowicie ich charakter fraktalny. W zależności od przyjętej skali analizowanej przestrzeni będziemy zwracać uwagę na punkty główne o różnych polach oddziaływania formalnego. W skali całej warszawskiej dzielnicy Śródmieście plac Zbawiciela stanowi jedynie jeden z kilku formalnie podkreślonych punktów węzłowych całego założenia

ujazdowskiego. Jeśli zbliżymy się do trójkąta wyznaczonego przez osie ulic Waryńskiego, Koszykowej i al. Armii Ludowej, to plac Zbawiciela przyjmie charakter pojedynczego punktu głównego jako centrum założenia gwiaździstego. Gdy zbliżymy się jeszcze bardziej do samego placu, nasza uwaga padnie już bezpośrednio na kościół Najświętszego Zbawiciela jako dominanty przestrzennej lub na sam centralny punkt placu wyznaczony przez przecięcie osi widokowych (jeśli kościół Zbawiciela nie pojawi się akurat w polu naszej percepcji)¹⁹. Punkty główne nie występują w pojedynczej, właściwej sobie skali. Uwidaczniają się względnie płynnie w zależności od przyjętego w danej chwili zakresu pola postrzegania i mogą funkcjonować w skali zarówno urbanistycznej, architektonicznej, jak i detalu. W skali wielkomiejskiej punkty główne będą budowane przez elementy takie jak: bramy, ciągi komunikacyjne, ulice wlotowe do miasta, stałe elementy przestrzenne planu miasta, założenia przestrzenne w dużej skali, układ komunikacyjny oraz główne elementy tożsamości miasta takie jak doliny rzeczne i skarpy (Pluta, 2001). W mniejszej skali pojedynczego placu punkty główne będą wyznaczane przez detal ścian, podłogi i stropu wnętrza urbanistycznego. Detale te uczytelniają i harmonizują kompozycję przestrzenną. Największe znaczenie mają detale podłogi urbanistycznej, takie jak: naprowadzające tafle wody, rysunki na posadzce, mała architektura, zróżnicowania materiałowe, poręcze, pochylnie i murki. W drugiej kolejności wpływ mają ściany wnętrza przez samą tektonikę pierzei, ale też szpalery zieleni, pergole, skarpy, elementy rytmiczne, bramy czy tablice reklamowe. W skali tej najmniejszą wagę ma płaszczyzna stropu dysponująca zadaszeniami, łącznikami architektonicznymi czy liniami energetycznymi (Pluta, 2014).

2.4.5. Znaczenia punktów głównych

Znaczenie funkcjonalne i społeczne

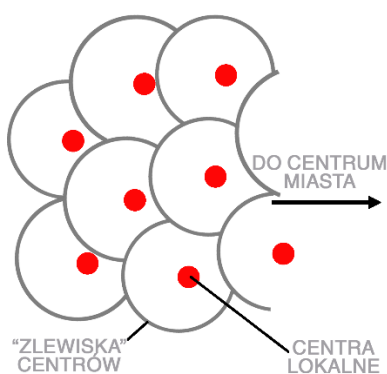
Punkty główne nie są jedynie *stricto* geometrycznym fenomenem zależnym od fizycznego zagospodarowania miasta. Elementy te zawsze będą miały swoje znaczenia także w wymiarach funkcjonalnym, społecznym, symbolicznym i ekonomicznym. Ze względu na ich wagę w kompozycji przestrzennej punkty główne naturalnie zyskują także na atrakcyjności w sferze

¹⁹ Napięcie między tymi dwoma kandydatami na punkt główny kompozycji uzyskało swoją symboliczną kulminację w momencie zlokalizowania w centrum placu Zbawiciela słynnej tęczy projektu artystki Julity Wójcik. Łuk tęczy jeszcze bardziej zaakcentował przecięcie osi ulic i odciągnął uwagę od kościoła Zbawiciela.

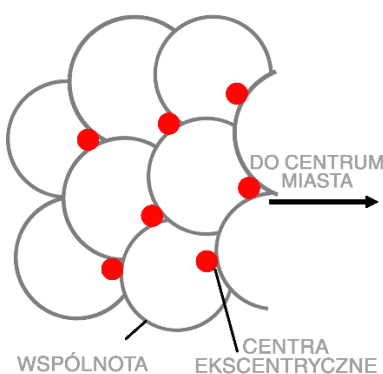
funkcjonalnej. Radialne przecięcie kilku osi multimodalnego transportu wyznacza punkt dla węzła przesiadkowego. Dominanta wysokościowa ze względu na swoje gabaryty pozwala na lokalizację w swoim wnętrzu atrakcyjnych funkcji o dużej intensywności. Naprowadzenia i ekspozycje widokowe zwiększają rangę miejsca w mapach mentalnych użytkowników przestrzeni, zwiększając wartość ekonomiczną miejsca i przekładając się na potrzebę lokowania funkcji społecznie ważnych. Niemiecki geograf i etnograf Friedrich Ratzel w swojej *Antropogeografii* postawił wręcz tezę, że żadne miasto nie może powstać bez istnienia pierwotnego punktu głównego wyznaczonego przez centralność, skrzyżowanie szlaków komunikacyjnych i dostępność topograficzną (Ratzel, 1882). Osada zlokalizowana w takim miejscu działa jako punkt ściągający kontakty, zarówno z otaczających terenów wiejskich, jak i z sąsiednich miast. Punkty główne mogą działać jak generatory rozwoju organizujące strukturę całych miast (Mironowicz, 2013).

W *Języku Wzorców* Christopher Alexander, wychodząc od społecznych założeń formowania się miast, mówi o dwóch skalach centrów w tkance urbanistycznej. W skali obszarów zamieszkałych przez 300 tysięcy mieszkańców identyfikuje intensywne centra miejskie będące jądrami dla całych regionów miejskich. Przy proponowanym przez Alexandra układzie palczastych zatok miejsko-wiejskich²⁰, centra takie mają tendencję do formowania się na przecięciu osi stykających się ze sobą podłużnych zatok miejskich (Alexander, i in., 1977, s. 59). W bardziej lokalnej skali obszarów zamieszkałych przez 7 tysięcy mieszkańców Alexander pisze o centrach wspólnotowych, wynikających ze struktury sąsiedzkiej miasta. Centra te są uwarunkowane przez codzienne migracje mieszkańców, presję ekonomiczną i układ morfologiczny klastrow sąsiedzkich. Tom Brennan w latach 40. zauważył na przykładzie angielskiego miasta Wolverhampton, że zlewiska funkcjonalne centrów lokalnych nie mają formy zbliżonej do okręgów, a bardziej do półokręgów, w których handlowe centra lokalne powiązane z węzłami komunikacyjnymi przesunięte są w kierunku głównego centrum całego miasta (Brennan, 1948). Wynika to z determinującej codzienne migracje ludności siły przyciągania centrum głównego oddziałującego na obszarze całego miasta (Rys. 23).

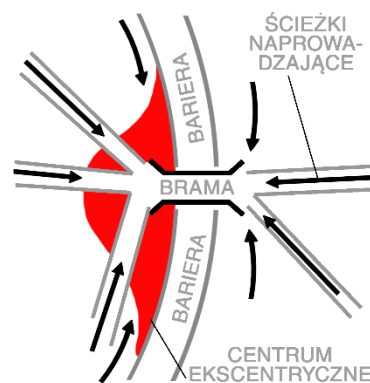
²⁰ Ang. City Country Fingers oznaczające wydłużone pasma gęstej zabudowy miejskiej o szerokości do jednej mili, rozdzielone przez bardziej naturalne pasma zabudowy wiejskiej o szerokości minimum jednej mili (Alexander, i in., 1977, s. 25)



Rys. 23. Zlewiska centrów lokalnych Brennana. Centra lokalne przesunięte są w kierunku głównego centrum miasta [na podstawie (Brennan, 1948)]



Rys. 24. Centra ekscentryczne Alexandra. Znajdujące się na granicach między obszarami społeczności [na podstawie (Alexander, i in., 1977)]



Rys. 25. Zbliżenie na centrum ekscentryczne Alexandra. Naprowadzają na nie ścieżki i bariery wyznaczające obszar społeczności. Opracowanie własne

Alexander rozwija myśl Brennana i proponuje struktury „centrów ekscentrycznych” zlokalizowanych na granicy pomiędzy sąsiadującymi ze sobą społecznościami. Centrum takie obsługuje głównie społeczność znajdującą się w większym oddaleniu od centrum głównego miasta (Rys. 24). Ekscentryczne centra powinny krystalizować się w okolicy bramy łączącej sąsiednie społeczności. Lokalizacja taka generuje naprowadzenia geometryczne – z jednej strony przez ścieżki, ulice i przejścia zbiegające się w rejonie bramy, z drugiej przez bariery wyznaczające granice odrębnych społeczności, Bariery naprowadzające na centrum mogą mieć formę pierzei zabudowy, szpalerów zieleni, barier topograficznych, cieków wodnych czy linii kolejowych (Rys. 25). Centra ekscentryczne charakteryzują się największą w sąsiedztwie gęstość zabudowy (z gradientem intensywności malejącym w miarę oddalania się od centrum głównego miasta). W obrębie centrów ekscentrycznych znajdują się ważne przestrzenie publiczne, takie jak ulice i place handlowe uzupełnione o węzły komunikacyjne. Centra te powstają wzdłuż barier i głównej osi naprowadzającej na bramę w granicy otaczającej obszar zamieszkania danej społeczności. Przez to centra takie zyskują formę podkowy z lekkim wybrzuszeniem na jej osi symetrii (Alexander, i in., 1977, s. 90, 151).

Znaczenie symboliczne

Podkreślona formalnie lokalizacja punktów głównych przekłada się na ich dużą rangę symboliczną. Punkty główne zapisują się w mapach mentalnych użytkowników przestrzeni. W przypadku dominant wysokościowych i obiektów wykorzystujących topografię terenu fizycznie górują one nad resztą krajobrazu i są widoczne z wielu lokalizacji. Średniowieczne zamki zakładano na wzgórzach nie tylko ze względów obronnych. Miały one przypominać o swojej obecności, symbolizować władzę, autorytet i wzbudzać w mieszkańcach podgrodzia poczucie bezpieczeństwa. Symbole z definicji obejmują jednocześnie wiele znaczeń. Widoczne na dziesiątki kilometrów zamki obronne z jednej strony obiecywały zysk i wyznaczały kierunek dla zmierzających do miasta kupców, a z drugiej budziły strach i odstraszały potencjalnych najeźdźców. Przez wieki podporządkowywania sobie okolicznej kompozycji przestrzennej punkty główne stają się elementami dziedzictwa historycznego i kulturowego. Znajdujące się w podkreślonych kompozycyjnie punktach ważne funkcje administracyjne, religijne, edukacyjne czy społeczne oddziałują na całe pokolenia stopniowo wpisując się coraz mocniej w tożsamość mieszkańców dzielnic czy miast. Obiekty takie jak Zamek na Wawelu, Wieża Eiffla, Akropol czy pomnik Waszyngtona oddziałują w kulturze tak szeroko, że funkcjonują jako ikony dla całych narodów.

Nastawieni przyszłościowo projektanci, świadomi tych relacji, także przy współczesnych realizacjach mogą kształtować symbolikę projektowanych miejsc. Krystyna Guranowska-Gruszecka zwraca uwagę na przykłady względnie nowych warszawskich dominant wysokościowych funkcjonujących jako punktowe podkreślenia rangi węzłów miejskości. Warto wspomnieć między innymi o Błękitnym Wieżowcu podkreślającym wagę placu Bankowego i sprzęgającym jego całe założenie kompozycyjne w całość. Innymi wysokościowcami pełniącym podobną rolę są Intraco, wyznaczający granicę Śródmieścia Północnego czy Plac Unii wyznaczający granicę Śródmieścia Południowego (Guranowska-Gruszecka, 2018).

Znaczenie ekonomiczne

Nasycone symbolicznie, dobrze skomunikowane punkty główne często powiązane są z aktywnymi i atrakcyjnymi ekonomicznie przestrzeniami publicznymi. Większa intensywność zabudowy wiąże się z dużym zagęszczeniem różnego rodzaju funkcji stymulujących interakcje między nimi. Mieszanka wzajemnie wspierających się obiektów usługowych, handlowych, kulturalnych, rekreacyjnych, edukacyjnych i administracyjnych

tworzy stymulującą rozwój miejsca sieć lokalnych powiązań. Christopher Alexander ściśle łączy ideę ekscentrycznych centrów sąsiedzkich z ich funkcjami komercyjnymi proponując lokalizację w ich obrębie ulic i promenad handlowych oraz węzłów aktywności lokalnej (Alexander, i in., 1977). Jan Gehl także zwraca uwagę na fenomen zagęszczenia sieci aktywności miejskich oraz interakcji międzyludzkich w obrębie intensywnych, dostępnych i dobrze zintegrowanych funkcjonalnie przestrzeni publicznych (Gehl, 1971). Kompozycja urbanistyczna jest w dużej mierze procesem komercyjnym. Sławomir Gzell pisze:

„Powiada się, że urbanistyka preferuje sytuację w której strumień pieniędzy z niesprecyzowanego źródła pomógł będzie w tworzeniu pięknych krajobrazów dla samej miłości do urody otaczającego nas świata. Tak nie jest. Każdy urbanista wie (choćby z wykładanej na uczelniach) historii urbanistyki, że teoria „wartości rzeczy pięknej” (...) obowiązuje od dawna – trzeba tylko to zrozumieć. Dziś też urbanistyka powiada, że nie ma kontradystoryjności między tym, co komercyjne a tym, co zakomponowane, albo inaczej: kontradystoryjność ta może nie występować i nie powinna.” (Gzell, 2013, s. 236)

Ceny nieruchomości charakteryzują się bardzo dużym zróżnicowaniem i są pochodną wielu czynników, jednak ze względu na prestiżową lokalizację można spodziewać się, że wartość działek znajdujących się w obrębie lub pobliżu kompozycyjnego punktu głównego może być wyższa niż działek okolicznych. Harmonia kompozycji bezpośrednio przekłada się na estetykę przestrzeni, a walory estetyczne są jednym z kryteriów decydujących o cenie nieruchomości (Śleszyński, i in., 2018). W przypadku punktów głównych wyznaczonych przez przecięcie osi, w których jedna z nich stanowi krawędź urbanistyczną (jak w proponowanych przez Alexandra centrach ekscentrycznych) wartość ekonomiczna terenu jest dodatkowo zwiększona przez pozytywny wpływ krawędzi kompozycyjnych na ceny nieruchomości. Lokalizacje znajdujące się w przestrzeniach stykowych charakteryzują się większymi walorami krajobrazowymi w stosunku do przestrzeni wewnętrznych obszarów (Paszkowski, 2007). Mocnym przykładem tego efektu jest krajobrazowy wpływ krawędzi tworzonej przez skarpe warszawską na wyższe ceny nieruchomości znajdujących się w jej obrębie (Śleszyński, i in., 2018; Achmatowicz-Otok i Jarosz, 1996). Kompozycyjne punkty główne o ponadlokalnej skali oddziaływania, stanowiące jednocześnie ikony dziedzictwa kulturowego mogą zwiększać potencjał turystyczny danego miejsca, a nawet całego miasta. Ze względu na znaczenie

ekonomiczne punktów głównych, ich rozpoznawanie, wzmacnianie i tworzenie powinno leżeć w sferze zainteresowań deweloperów budujących nasze miasta.

2.4.6. Rola punktów głównych w planowaniu przestrzennym

Analizy kompozycyjne oraz rozpoznawanie i wyznaczanie punktów głównych zajmują ważną rolę w planowaniu przestrzennym. W publikacji *Przestrzenie publiczne miast europejskich: Projektowanie urbanistyczne* prof. Katarzyna Pluta pisze:

„(...)Kompozycja urbanistyczna stanowi główne kryterium urody każdego miasta, jest to czynnik, który decyduje o jakości przestrzeni (...). Początkiem wszelkich prac związanych z poprawą krajobrazu miasta, wykorzystujących teorie kompozycji urbanistycznej, powinna być analiza morfologii stanu istniejącego. Dotyczy to zwłaszcza stanu istniejącego.” (Pluta, 2014, s. 79-81)

Ustawa o planowaniu i zagospodarowaniu przestrzennym jest podstawowym aktem prawnym regulującym planowanie przestrzenne w Polsce. W artykule 2. zaznacza, że uwarunkowania kompozycyjno-estetyczne są jednymi z kluczowych wyznaczników osiągnięcia ładu przestrzennego (Sejm RP, 2003). Istniejące i projektowane osie naprowadzeń widokowych wskazujących na lokalne punkty główne są ważnymi aspektami świadczącymi o jakości przestrzeni miejskiej. Ich uwzględnianie jest jednym z ważniejszych obowiązków spoczywających na urbanistach próbujących osiągnąć ład przestrzenny. Cytując Jana Gehla:

„Patrzenie na życie miasta – oto jedna z najważniejszych i najpopularniejszych jego atrakcji. (...) Urbanista powinien jednak poświęcać uwagę także widokowi na inne atrakcje: wodę, drzewa, kwiaty, fontanny czy architekturę, bo tylko wtedy możemy mówić o pełniejszej dbałości o dobrą jakość miasta. (...) Ponieważ niezakłócony widok na miejskie atrakcje jest tak istotny, to pole widzenia i wyznaczające je linie muszą być traktowane równie poważnie jak same widoki.” (Gehl, 2014, s. 148)

W *Życiu między budynkami* Gehl definiuje przestrzeń projektową planowania przestrzennego rozpinając ją na pięciu wymiarach reprezentujących możliwe do podjęcia działania planistyczne. Każdy z tych wymiarów opisywany jest przez dwa przeciwne sobie kierunki:

- Rozpraszenie i gromadzenie.
- Segregowanie i integrowanie.
- Odpychanie i zapraszanie.
- Zamykanie i otwieranie.

- Zmniejszanie i zwiększanie (Gehl, 1971).

Punkty główne są elementami kompozycji maksymalizującymi te procesy. Mają potencjał do gromadzenia ludzi i integracji funkcji ze względu na naprowadzające na punkt osie i otwarcia widokowe. Mogą charakteryzować się zwiększoną w stosunku do otoczenia intensywnością – z jednej strony aktywności mieszkańców, a z drugiej – zlokalizowanej w obrębie punktu zabudowy.

Wprowadzone do polskiej teorii planistycznej przez Zbigniewa Zuziaka „węzły miasta” oznaczające intensywne, wielowarstwowe, metropolitalne struktury przestrzenno-funkcjonalne często tworzą się w pobliżu lub bezpośrednio w obrębie urbanistycznych punktów głównych (Guranowska-Gruszecka i Łaskarzewska, 2018). Krystyna Guranowska-Gruszecka i Małgorzata Łaskarzewska piszą o węzłach miasta następująco:

„Węzły miasta mogą dotyczyć różnych zagadnień: głównie przestrzennych i funkcjonalnych tworzy je zabudowa kształtowana jako identyfikacja miasta wyróżniająca się dominacją przestrzenną np. poprzez wysokość lub intensywność, o wyróżniającą się funkcji – np. nowe centra usługowe, administracyjne, koncentracje miejsc pracy, centra kultury a także koncentracje funkcji rekreacyjnych. Istotą tych miejsc powinna być dbałość o tożsamość kulturową.”
(Guranowska-Gruszecka i Łaskarzewska, 2018, s. 10)

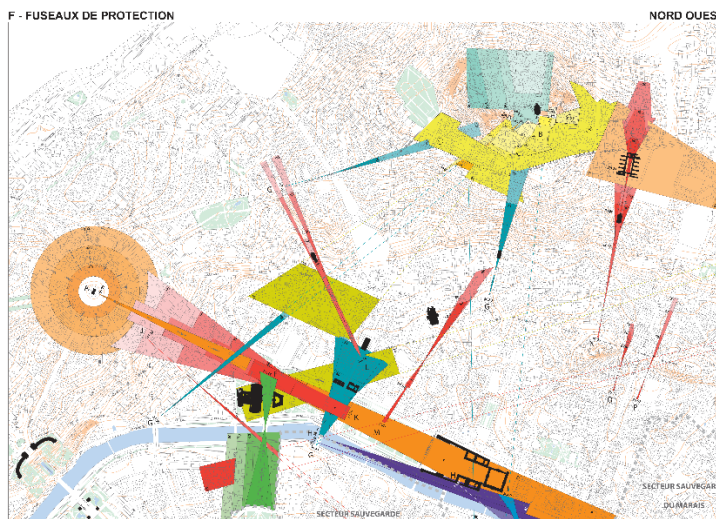
Trochę innym terminem są definiowane osobno „węzły miejskości”. O ile węzły miasta można wyróżnić głównie w skali całej metropolii to węzły miejskości mogą przyjmować różne skale, a głównym ich wyznacznikiem jest zwiększony poziom aktywności miejskiej w ich obrębie:

„Węzły miejskości (WM) – to wielowarstwowe formy zagospodarowania przestrzennego odpowiadające koncentracjom funkcji, czyli aktywnościom, związanym z miejskością. Przykładami węzłów miejskości są: centra miast, centra dzielnicowe i osiedlowe, a także inne koncentracje funkcjonalno-przestrzenne, np. miejsca czy obszary wokół przystanków transportu zbiorowego, gdzie intensywność zabudowy rośnie odpowiednio do dostępności komunikacyjnej centralnego miejsca danego skupiska i gdzie wykształcił się lub pojawia się pewien typ przestrzeni urbanistycznej, który można uznać jako przestrzeń publiczną. W przypadku układów miejskich mających silny związek z gospodarką opartą na wiedzy (...), istotną rolę odgrywają węzły

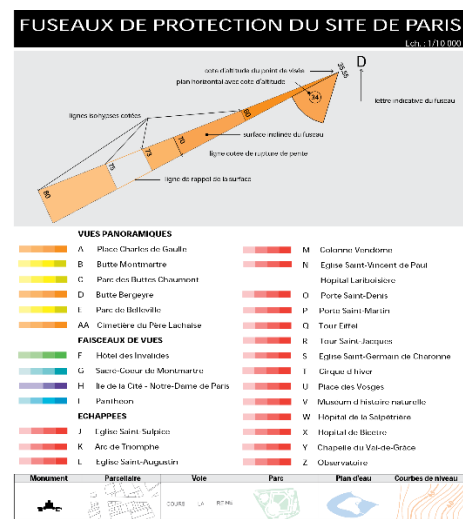
miejskości odpowiadające kampusom uniwersyteckim, parkom nauki, parkom technologicznym itp.” (Guranowska-Gruszecka i Łaskarzewska, 2018, s. 11)

Punkty formalnie podkreślone często podawane są jako przykłady miejsc sprzyjających formowaniu się węzłów miejskości. Anna Jeziorska podaje jako przykład wspomniany już wcześniej Plac Zbawiciela w Warszawie będący kompozycyjnym centrum założenia gwieździstego (Jeziorska, 2018). Do podobnych wniosków doszli urbaniści z Oddziału Warszawskiego SARP w *Studium Koncepcyjnym dotyczącym centrów lokalnych w Warszawie* z 2015 roku. W badaniu ankietowym mającym pomóc w identyfikacji centrów lokalnych zauważono, że warto, by centrum takie wyróżniało się jakimś punktem charakterystycznym nadającym mu niepowtarzalny charakter. Jako przykład Krzysztof Herman w jednym z aneksów podaje park im. Zasława Malickiego w Warszawie, w którym punktem ogniskowym całego zespołu parkowo-mieszkaniowego jest centralny staw (Herman, 2015). Wiele proponowanych w studium centrów lokalnych powiązanych jest ze znaczącymi węzłami komunikacyjnymi, przecięciami osi naprowadzających, dominantami przestrzennymi i miejscami intensyfikacji aktywności miejskiej (SARP, Oddział Warszawski, 2015).

Punkty główne i punkty kompozycyjnie podkreślone uwzględniane są w procesie planowania przestrzennego. Dobrym przykładem była trzydziesta piąta wersja, paryskiego planu urbanistycznego PLU (fr. *Plan Local d’Urbanisme de Paris*). Plan ten chronił istniejące założenia kompozycyjne i ekspozycje widokowe (Rys. 26, Rys. 27). Nowsze wersje PLU podporządkowują wszelkie interwencje projektowe obowiązującej logice kompozycyjnej (Le Conseil de Paris, 2023). W PLU znajdują się zapisy identyfikujące i chroniące najważniejsze dla kompozycji urbanistycznej punkty orientacyjne oraz obiekty współtworzące elementy ciągle naprowadzające na te punkty (Le Conseil de Paris, 2023).



Rys. 26. Strefy ochronne z wyraźnie oznaczonymi punktami głównymi pochodzące z trzydziestej piątej wersji paryskiego planu urbanistycznego PLU dla obszaru Pól Elizejskich (Le Conseil de Paris, 2015).



Rys. 27. Legenda do rysunku stref ochronnych z trzydziestej piątej wersji PLU (Le Conseil de Paris, 2015).

Rozważania na temat punktów głównych znaleźć można także w opracowaniach planistycznych polskich miast. Za przykłady można podać dokumenty planistyczne Warszawy i Łodzi. W obecnie obowiązującym *Studium Uwarunkowań i Kierunków Zagospodarowania Przestrzennego Warszawy* z 2006 roku (ze zmianami z 2021 roku) wskazane centra dzielnicowe i lokalne także są ściśle powiązane z kompozycją przestrzenną:

„Dla tworzenia elementów identyfikacji przestrzennej w strukturze przestrzennej miasta (...) istotne jest zaakcentowanie centrów dzielnicowych i lokalnych jako dominant (wyróżników) urbanistycznych o wysokich walorach architektonicznych. (...) [Wymagane jest] kształtowanie zabudowy centrów w sposób tworzący spójną kompozycję przestrzenną, podkreślającą rangę miejsca, między innymi poprzez wprowadzanie wysokiego standardu rozwiązań architektonicznych, technicznych i materiałowych.” (Prezydent m.st. Warszawy, 2006 ze zmianami, s. 106)

Projekt nowego Studium Warszawy z 2023 roku rozwija tę myśl i wyznacza trzy hierarchie centrów urbanistycznych. W największej skali zdefiniowanych jest pięć „bram wielkomiejskich” pełniących funkcję centrów ponaddzielnicowych. Powiązane są one z multimodalnymi węzłami komunikacyjnymi (także komunikacji dalekobieżnej), koncentrują one funkcje usługowe, miejsca pracy i charakteryzują się wysoką intensywnością zabudowy. Ranga bram wielkomiejskich powinna być podkreślona przez dominanty wysokościowe

sprzęgające kompozycję przestrzenną. Kolejną kategorię stanowią centra dzielnicowe funkcjonujące w skali pojedynczych dzielnic²¹. One także mają potencjał do wiązania się z kompozycją przestrzenną w celu podkreślenia ich tożsamości, wzmocnienia indywidualnych cech i budowania ich rozpoznawalności. W najmniejszej skali studium definiuje centra lokalne obsługujące obszary wyznaczone przez zasięg pieszego spaceru. Studium dopuszcza lokalizację dominant wysokościowych i miejscowych podwyższeń zabudowy względem otoczenia w celu podkreślenia kompozycji urbanistycznej centrów lokalnych, dzielnicowych i bram wielkomiejskich. Dominanty takie powinny być powiązane z osiami kompozycyjnymi i widokowymi oraz z panoramami i sylwetami miasta. Studium rekomenduje, by kompozycja urbanistyczna była budowana także przy wykorzystaniu niskiej, średniej i wysokiej zieleni miejskiej (Prezydent m.st. Warszawy, 2023).

Studium Łodzi z 2018 roku wskazuje obiekty wyróżniające się w krajobrazie miasta ze względu na znaczenie kompozycyjne, kubaturę, formę, wysokość lub walory architektoniczne. Punkty te stanowią ważny element analiz krajobrazowych, polegających między innymi na sprawdzaniu ich widoczności z osi widokowych i z punktów widokowych. W części kierunkowej studium zalecono zagospodarowywanie terenów sąsiednich wyróżniających się obiektów, zapewniające ich odpowiednią ekspozycję i relacje widokowe. Nowe obiekty projektowane jako wyróżniające się w krajobrazie powinny być lokalizowane z uwzględnieniem powiązanych z nimi ciągów widokowych (Prezydent miasta Łodzi, 2018). Archiwalne studium Łodzi z 2010 roku uwzględnia dwa punkty główne. Pierwszy z nich, czyli tzw. „Łódzki Manhattan” z centralnym węzłem komunikacyjnym, wyznaczony jest przez przecięcie dwóch głównych osi krystalizujących plan Łodzi (osi historycznej północ-południe i osi współczesnej wschód-zachód). Drugi punkt to Atlas Arena, będąca dominantą w zamknięciu kompozycyjnym wewnętrznej osi krystalizującej wschód-zachód (Prezydent miasta Łodzi, 2010).

2.4.7. Automatyczne rozpoznawanie punktów głównych

O ile projektowanie punktów głównych w kompozycjach przestrzennych leży głównie w sferze zainteresowań architektów, urbanistów, planistów, artystów i projektantów interfejsów

²¹ Nazwa „centrum dzielnicowe” jest trochę niefortunna, ponieważ sugeruje pojedynczy punkt administracyjno-usługowy obsługujący daną dzielnicę. W rzeczywistości autorzy studium wyznaczyli w większości dzielnic po kilka centrów dzielnicowych.

graficznych, to, ze względu na ich rozległe znaczenia opisane powyżej, rozpoznawanie punktów głównych jest przydatne także w innych branżach. Ekstrakcja punktów formalnie podkreślonych może być przydatna nie tylko w praktyce projektowej, lecz także w pracy jednostek samorządu terytorialnego, administracji, deweloperów, inwestorów, rzeczoznawców majątkowych, aktywistów oraz może być wykorzystywana przez samych mieszkańców zainteresowanych jakością urbanistyczną ich miast.

Wraz ze stopniowym wchodzeniem otaczających nas miast do sfery wirtualnej i pojawianiem się ich cyfrowych reprezentacji, możliwości automatyzacji analiz przestrzennych stają się coraz większe. Pojawiają się coraz to nowsze narzędzia GIS-owskie, operujące na różnorodnych danych źródłowych, a istniejące bazy danych stają się coraz obszerniejsze i dokładniejsze. Kompozycja przestrzenna jest jednak dziedziną mocno subiektywną, w literaturze brak jest wspólnego języka i konstrukcji teoretycznej pozwalającej na techniczną analizę elementów kompozycyjnych (Mironowicz, 2013). Co więcej, zautomatyzowane rozpoznawanie punktów głównych w kompozycjach przestrzennych jest dosyć niszowym problemem i nie jest szeroko dyskutowane w literaturze. Za inspirację i podstawę metodyczną mogą jednak posłużyć narzędzia stosowane do rozwiązywania problemów pokrewnych.

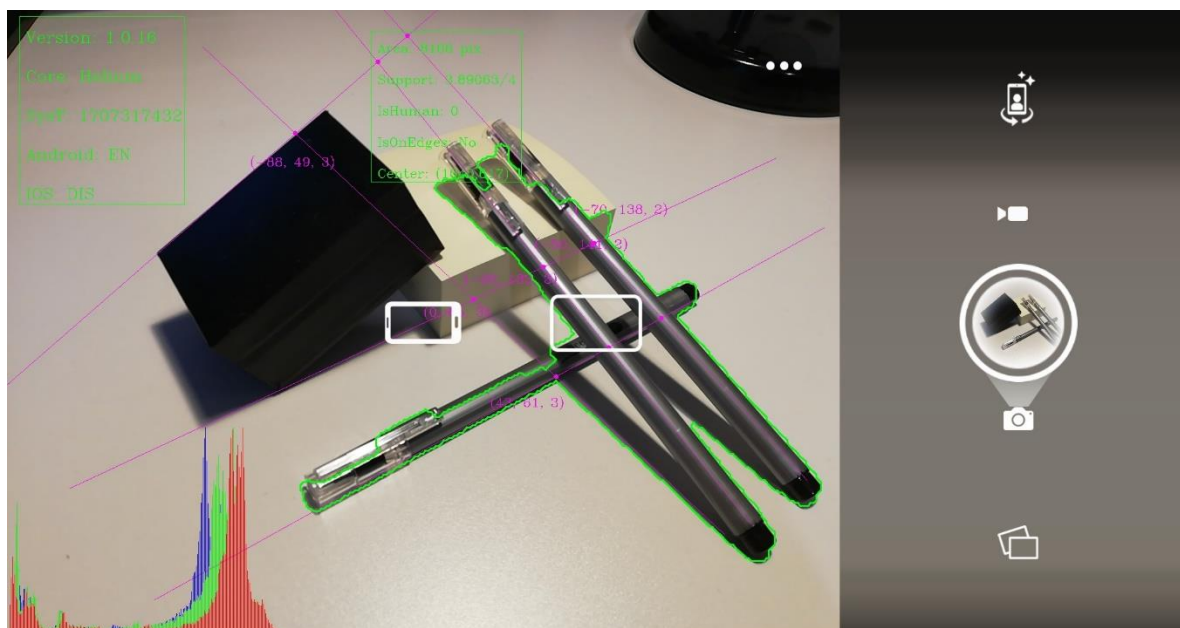
Wskazywanie pewnych istotnych punktów jest dosyć szeroko stosowane jako narzędzie wspomagające i uczytelniające cyfrowe interfejsy graficzne. Przykładem może być program AutoCad, który już w wersji 2.0 z 1984 roku wprowadził opcję „OSNAP” (ang. Object Snap) pozwalającą na automatyczne przyciąganie kursora do pewnych punktów referencyjnych takich jak centrum okręgu, środek lub zakończenie odcinka albo miejsce przecięcia linii (Hurley, 2008). Funkcja „OSNAP” jest na tyle intuicyjna, że 30 lat później w najnowszych wersjach programu działa ona niemal identycznie jak w wersji 2.0. Rozpoznawanie tych punktów oparte jest na bardzo prostych zasadach matematycznych i nie uwzględnia bardziej złożonych zasad kompozycyjnych, chociaż biorąc pod uwagę możliwości obliczeniowe współczesnego sprzętu komputerowego, zdecydowanie mogłaby uwzględniać także punkty istotne na wyższych poziomach abstrakcji kompozycyjnej.

Innym przykładem automatycznego rozpoznawania pewnych istotnych punktów są współczesne sterowniki aparatów fotograficznych w telefonach komórkowych, które potrafią automatycznie rozpoznawać ważne elementy znajdujące się w kadrze (jak np. twarze) i ustawiać na nich ostrość. Niektóre z nich dostarczają także użytkownikom informacji o tym, jak ustawić aparat w przestrzeni, aby kompozycja na zdjęciu była jak najlepiej wykadrowana.

Jednym z takich systemów jest program Camera51 z 2014 roku (Rehm, 2014). Algorytm w programie Camera51 opracowany został przez Ofriego Masada i Ariela Shamira z firmy ArtInCam Ltd (Rys. 28). Program wykorzystuje nieneuronowe metody rozpoznawania obrazu do ekstrakcji ze zdjęcia podstawowych, dwuwymiarowych elementów kompozycyjnych w postaci jednolitych obszarów i ich geometrycznych centrów, klastrów zgrupowanych ze sobą obszarów, głównych elementów liniowych i osiowych, punktów świetlnych oraz ludzkich sylwetek i twarzy. Następnie dla rozpoznanych elementów obliczane są funkcje reprezentujące najbardziej zbliżone możliwe zasady kompozycyjne, obejmujące:

- Zasadę trójpodziału kadru (podział kadru na 3 równe części horyzontalne i wertykalne. Elementy liniowe powinny pokrywać się z liniami definiującymi podział, a obiekty punktowe i obszarowe powinny pokrywać się z punktami przecięcia linii podziału).
- Zasadę przekątnych (linie skośne powinny pokrywać się z przekątnymi kadru).
- Zasadę przestrzeni (wykorzystanie tła kompozycji do podkreślenia elementu pozytywowego).
- Zasadę kompozycji centralnej (elementy punktowe i obszarowe powinny znajdować się w centrum kadru).
- Zasadę kompozycji symetrycznej (z osią symetrii umieszczoną w centrum kadru).
- Zasadę złotego podziału (podział kadru na części odpowiadające złotemu podziałowi).

Funkcje te są następnie sumowane dla rozpoznanych elementów kompozycyjnych i na tej podstawie generowane jest maksimum funkcji pochodnej złożonej z wybranej sumy zasad kompozycyjnych. Punkt ten reprezentuje położenie centrum nowego kadru. Użytkownik otrzymuje na ekranie aparatu intuicyjną instrukcję jak przesunąć aparat, by poprawić kadrowanie kompozycji (Masad i Shamir, 2017).

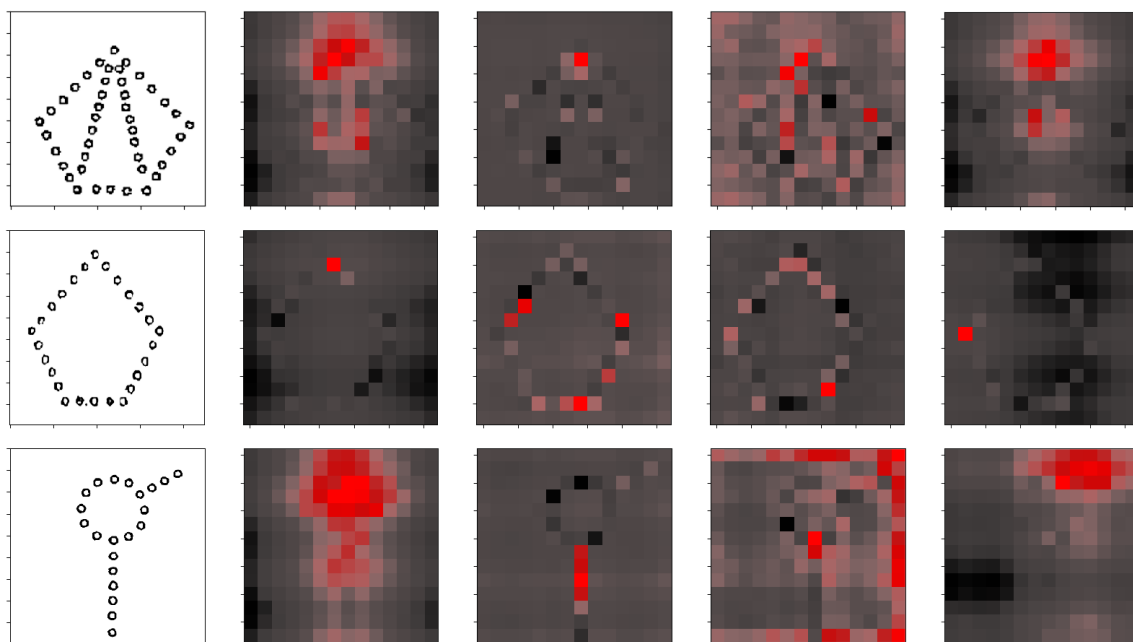


Rys. 28. Zrzut ekranu z trybu deweloperskiego programu Camera51, pokazujący rozpoznane przez aplikację elementy kompozycyjne. Ikona w centrum ekranu pokazuje, w jaki sposób przesunąć aparat, by polepszyć kompozycję kadru (Masad i Shamir, 2017).

Punkty główne ogólnie można traktować jako swego rodzaju maksima w wielowymiarowej przestrzeni cech kompozycyjnych. W ich obrębie dochodzi do przecięcia osi, nałożenia na siebie pól ekspozycji widokowych, intensyfikacji aktywności miejskich, zagęszczenia znaczeń symbolicznych, a często też punkty główne znajdują się na szczytach wzniesień lub stanowią dominantę przestrzenną. Problem w tym, że sam gradient „kompozycyjnej ważności punktu” zależy od wielu nakładających się na siebie czynników, które ciężko jednoznacznie zidentyfikować i opisać numerycznie. Można próbować w takich przypadkach poszukiwać łatwo mierzalnych wskaźników skorelowanych z występowaniem punktów głównych. W kontekście urbanistycznym w analizach przestrzennych często stosowana jest tak zwana metoda „transectowa”, nazywana także metodą profili urbanistycznych. Metoda ta opiera się na podziale morfologicznym miasta na strefy przy pomocy wybranych wskaźników urbanistycznych (Duany i Talen, 2002). Głównym wskaźnikiem składowym zazwyczaj jest gęstość zabudowy. Za przykład takiego systemu można podać „SmartCode”, opracowany przez firmę DPZ CoDesign, który stanowi modelowy szablon do projektowania urbanistycznego. W SmartCode wyszczególniono zbiór stref uszeregowanych od terenów naturalnych przez strefę wiejską, podmiejską, miejską, śródmiejską aż po rdzeń urbanistyczny (Duany Plater-Zyberk & Company, 2003). Zakładając za Christopherem Alexandrem,

że lokalne i globalne punkty centralne w miastach skorelowane są z maksimami intensywności tkanki miejskiej, gradienty profili „transectowych” można wykorzystać do wskazywania potencjalnych punktów głównych w dużych skalach urbanistycznych. Sam proces rozpoznawania stref „transectowych” może być przeprowadzany automatycznie na podstawie wskaźników GIS-owskich (Jachimowicz, 2023).

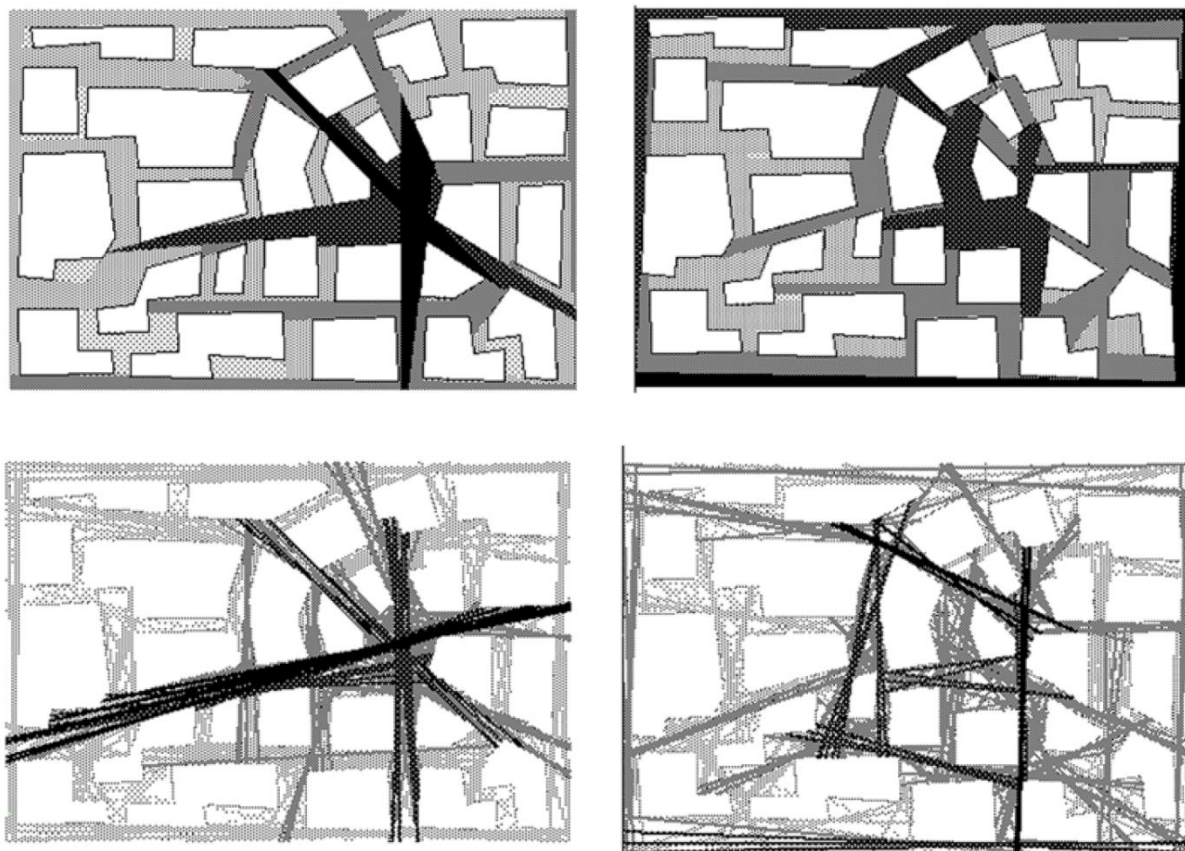
W projektowaniu architektonicznym wspomagany komputerowo często do odczytywania podobnych, ukrytych gradientów i odnajdywania ich maksimów wykorzystuje się uczenie maszynowe i sieci neuronowe. Oprócz zastosowań omówionych szczegółowo w kwerendzie zastosowania głębokiego uczenia maszynowego w CAAD za przykład można podać opracowany przez badaczy z IAAC system „Context Decoder”, służący do określania jakości miejsc w przestrzeni publicznej na podstawie ośmiu wskaźników, obejmujących między innymi gęstość zaludnienia i zabudowy, mapę komfortu termicznego, indeks powiązań z różnymi trybami transportu czy popularność danego miejsca wśród mieszkańców. System ten został przetestowany na przykładzie Neapolu i potrafił trafnie oceniać atrakcyjność poszczególnych stref (Marsillo, i in., 2022). Sieci neuronowe dobrze nadają się do odczytywania tego typu kontekstów, można więc oczekiwać, że dobrze poradzą sobie także w przetwarzaniu problemów kompozycyjnych. Jeśli pobierzemy z Internetu niewielką sieć neuronową typu ViT (Dosovitskiy, i in., 2021), wytrenowaną do rozpoznawania obiektów znajdujących się na zadanych zdjęciach i pokażemy jej przykłady prostych kompozycji przestrzennych, zauważymy, że sieć taka będzie przykładała wagę do punktów formalnie podkreślonych (Rys. 29). Dzieje się tak pomimo faktu, że sieć ta nie została wytrenowana *stricte* do czytania kompozycji przestrzennych. Próbuje ona tylko rozpoznać, czy dany obraz przedstawia kota, drzewo czy łańcuch.



Rys. 29. Wizualizacja wybranych głowic uwagi sieci neuronowej przetwarzającej proste kompozycje przestrzenne naszkicowane przez Juliusza Żórawskiego (Żórawski, 1962). Kolorem czerwonym oznaczone są miejsca w obrazie, które dana głowica uwagi uznała za istotne. Wykorzystana sieć to Vision Transformer (Dosovitskiy, i in., 2021) wytrenowany przez Rossa Wightmana (Wightman, 2019) do rozpoznawania klas ze zbioru obrazów ImageNet (Russakovsky, i in., 2015). Sieć nie próbuje odczytywać kontekstów kompozycyjnych, tylko stara się rozpoznać, co znajduje się na obrazie (w tym przypadku wszystkie szkice Żórawskiego zostały rozpoznane jako zdjęcia łańcuchów). Sieć zwraca uwagę na różne fragmenty obrazu, jednak dużą wagę przykłada także do punktów formalnie podkreślonych. Opracowanie własne.

Najbardziej zbliżoną do rozpoznawania punktów głównych metodę proponuje Bill Hillier, w książce *Space is the machine*. Twórca teorii składni przestrzeni opisuje rdzenie integracji (ang. Integration cores). Hillier definiuje integrację jako miarę przestrzeni będącą odwrotnością sumy liczby osobnych przestrzeni, jakie musi przejść obserwator poruszający się między wszystkimi tymi przestrzeniami po kolei. Punkty, w których dochodzi do maksymalizacji tej miary, to rdzenie integracji (Hillier, 2007). W publikacji zaprezentowane są dwie metody ich wyznaczania. Pierwsza to dwuwymiarowa analiza nakładających się elementów wypukłych przeprowadzana przy pomocy komputerowej symulacji ruchu użytkownika przestrzeni w skali lokalnej. Druga to analiza jednowymiarowych elementów liniowych, polegająca na integracji

w skali globalnej wszystkich elementów liniowych równoległych do krawędzi kwartałów zabudowy. Obie metody przynoszą podobne rezultaty (Rys. 30).



Rys. 30. Dwie metody wyznaczania rdzeni integracji zaproponowane przez Billa Hilliera. U góry – metoda analizy nakładających się elementów wypukłych, u dołu – metoda analizy jednowymiarowych elementów liniowych. Metody zostały zaprezentowane na dwóch modelach tkanki miejskiej, jednej z czytelnym punktem głównym, drugiej bez takiego punktu. Obie metody dają zbliżone rezultaty (Hillier, 2007).

Podsumowując, automatyczne rozpoznawanie punktów głównych w kompozycjach przestrzennych, z uwagi na ich funkcjonalne, społeczne, symboliczne i ekonomiczne znaczenie, może znaleźć szerokie zastosowanie na wielu polach. Punkty te są istotnie nie tylko dla urbanistów i architektów, lecz także dla jednostek administracyjnych, deweloperów, rzeczoznawców oraz mieszkańców miast. Choć istnieją narzędzia do automatycznego wskazywania pewnych kluczowych punktów, problematyka automatyzacji rozpoznawania punktów głównych w kompozycjach przestrzennych nie była do tej pory eksplorowana. Biorąc pod uwagę złożoność układów przestrzennych i wielowymiarowość opisujących je cech, sieci neuronowe wydają się obiecującym narzędziem, które może sprostać temu zadaniu.

2.5. Uwarunkowania do wykorzystania sieci neuronowych w CAAD

2.5.1. Sieci neuronowe – narzędzie wielu dyscyplin

W celu pełniejszego zrozumienia potencjału sieci neuronowych mogących posłużyć do rozwiązania problemu rozpoznawania kompozycyjnych punktów głównych, należy przyrzeć się ich strukturze, historii i zasadom działania. Na samym początku warto dokonać rozróżnienia między uczeniem maszynowym, głębokim uczeniem maszynowym i sieciami neuronowymi a sztuczną inteligencją. Wszystkie te pojęcia stosowane są w literaturze dosyć nieprecyzyjnie, autorzy często decydują się na korzystanie z działających na wyobraźnię pojęć w celu przykucia uwagi, ale niekoniecznie zgodnie z rzeczywistą tematyką publikacji. Głębokie sieci neuronowe (nazywane w tej pracy synonimicznie z głębokim uczeniem maszynowym) są podkategorią uczenia maszynowego. Wielu autorów każde zastosowanie uczenia maszynowego nazywa „sztuczną inteligencją” (AI od ang. Artificial Intelligence). AI jest terminem nierozzerwalnie budzącym skojarzenia z jej przedstawieniami w fantastyce naukowej. Mówiąc „sztuczna inteligencja” widzimy oczyma wyobraźni komputer HAL9000 z *Odysei Kosmicznej* Kubricka (2001: A Space Odyssey, 1968), Skynet z *Terminatora* (The Terminator, 1984) czy zabójcze maszyny z *Matrixa* (The Matrix, 1999). Określenie, czym rzeczywiście jest sztuczna inteligencja, nie jest celem niniejszej dysertacji, jednak z utylitarne punktu widzenia uważam, że nadużywanie tego terminu może prowadzić do niekorzystnych skutków. Już w latach 50. twórca jednej z pierwszych sieci neuronowych Frank Rosenblatt próbował podniecać opinię publiczną zestawiając swój wynalazek z popularnymi wyobrażeniami science fiction. Jak pisał w 1958 roku New York Times:

„Marynarka Wojenna pokazała dzisiaj embrion elektronicznego komputera, który zgodnie z przewidywaniami będzie w stanie chodzić, mówić, widzieć, pisać, samodzielnie się reprodukować i będzie świadomy swojej egzystencji. (...) Dr Frank Rosenblatt, projektant Perceptronu przeprowadził demonstrację. Powiedział, że maszyna będzie pierwszym urządzeniem myślącym jak ludzki mózg. (...) [P]owiedział, że Perceptrony mogą być wysyłane ku planetom jako mechaniczni odkrywcy kosmosu. (...) Późniejsze Perceptrony będą potrafiły rozpoznawać ludzi i wołać ich po imieniu, oraz będą natychmiastowo tłumaczyć mowę z jednego języka na mowę lub tekst w innym języku.” [tłumaczenie własne] (New York Times, The, 1958, s. 9)

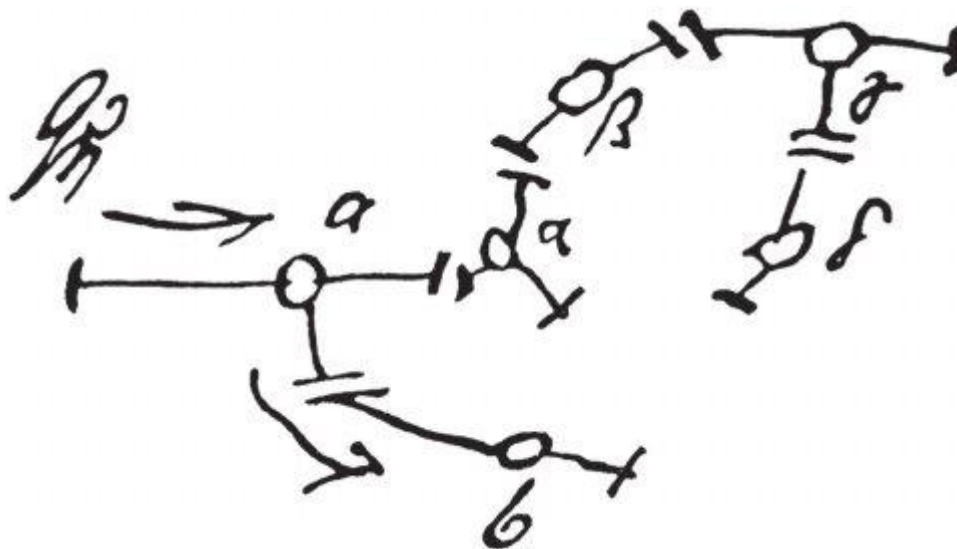
Sieci neuronowe są w gruncie rzeczy prostymi w założeniach, łatwymi w implementacji algorytmami uczącymi się statystycznie przetwarzać dane wejściowe na dane wyjściowe. Takie pragmatyczne podejście do sieci neuronowych zdecydowanie ujmuje im trochę uroku, ale myślenie o nich w kategorii „sztucznej inteligencji” wydaje się zbyt romantyczne, nacechowane emocjonalnie i może onieśmielać ich potencjalnych użytkowników.

Przed przejściem do dalszej części wywodu warto pokrótce, na intuicyjnym poziomie, przedstawić podstawy działania sieci neuronowych i ich historię. Mimo, że w ramach naszej dziedziny głębokie uczenie maszynowe wydaje się techniką nowatorską, w swoich założeniach sieci neuronowe nie są niczym nowym. Ich podstawy teoretyczne były opracowane w latach 50. dwudziestego wieku, a w praktyce były stosowane w ramach CAAD już od lat 90. dwudziestego wieku (Openshaw, 1993; Stevenson, 1994a; Stevenson, 1994b; Black, 1995; Kindangen, 1996).

Pierwszą siecią neuronową był opracowany w latach 50. dwudziestego wieku SNARC Marvin'a Minsky'ego (ang. Stochastic Neural Analog Reinforcement Calculator). Główne inspiracje teoretyczne dla Minsky'ego (Minsky i Sykes, 2017) stanowiły: teoria matematycznego opisu biologicznych sieci neuronowych Warrena McCullocha i Waltera Pittsa z lat 40. dwudziestego wieku (McCulloch i Pitts, 1943), książka o matematycznej biofizyce Nicolasa Rashevsky'ego z lat 30. (Rashevsky, 1938) oraz jeszcze starsza próba opisu psychologii jako ścisłej dziedziny naukowej opracowana w 1895 roku przez Sigmunda Freuda (Freud, 1950/1895)²². Sieć ta była urządzeniem mechaniczno-elektrycznym potrafiącym uczyć się na zasadzie uczenia ze wzmacnianiem. SNARC został przetestowany na problemie nawigacji w labiryncie. Za każdym razem gdy podejmował prawidłową decyzję, zostawał nagradzany przez operatora. SNARC był modelem stochastycznym, w którym połączenia między neuronami były

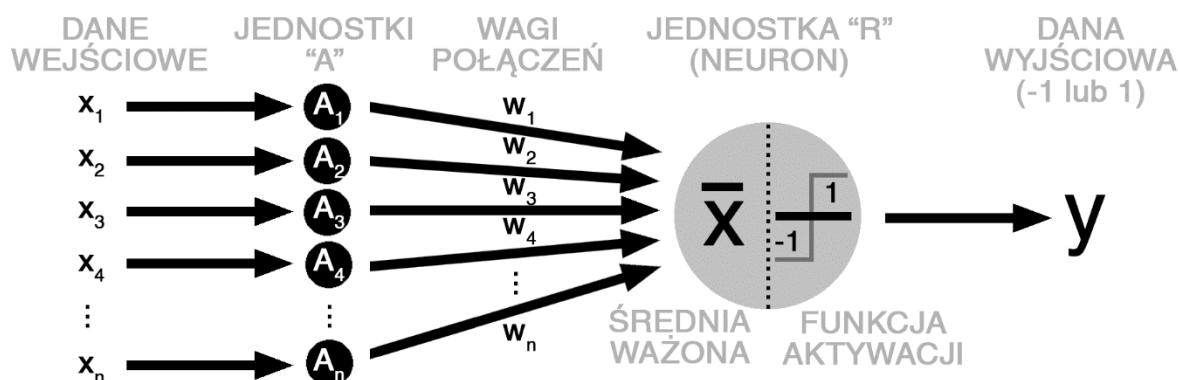
²² W *Projekcie Naukowej Psychologii* Freud nie tylko zaproponował swoją dosyć ezoteryczną i niepodpartą eksperymentalnie teorię sieci neuronowych (McCarley, 1998), ale przedstawił między innymi także pomysł zasad „przyjemności” i „nieprzyjemności” sterujących systemami biologicznymi. Według Freuda systemy takie wyposażone są w sprzężenie zwrotne, pozwalające na wzmacnianie zachowań skutkujących otrzymaniem bodźców przyjemnych i osłabianie zachowań skutkujących otrzymaniem bodźców nieprzyjemnych. Idea ta stanowi podstawę pod współcześnie rozumiany trening sieci neuronowych, w którym połączenia między neuronami przynoszące sieci korzyść ulegają wzmocnieniu, podczas gdy połączenia niekorzystne ulegają osłabieniu (Freud, 1950/1895). Freudowi nie udało się za życia opublikować „Projektu” w formie publikacji naukowej. Została ona wydana dopiero w latach 50. dwudziestego wieku.

probabilistyczne. Szansa, że sygnał z poprzedniego neuronu aktywuje neuron kolejny, zależała od elektronicznie ustawionego prawdopodobieństwa. Im częściej dany neuron aktywował się przy jednoczesnym otrzymaniu przez sieć nagrody, tym bardziej zwiększało się prawdopodobieństwo aktywacji neuronu (Minsky i Sykes, 2011).



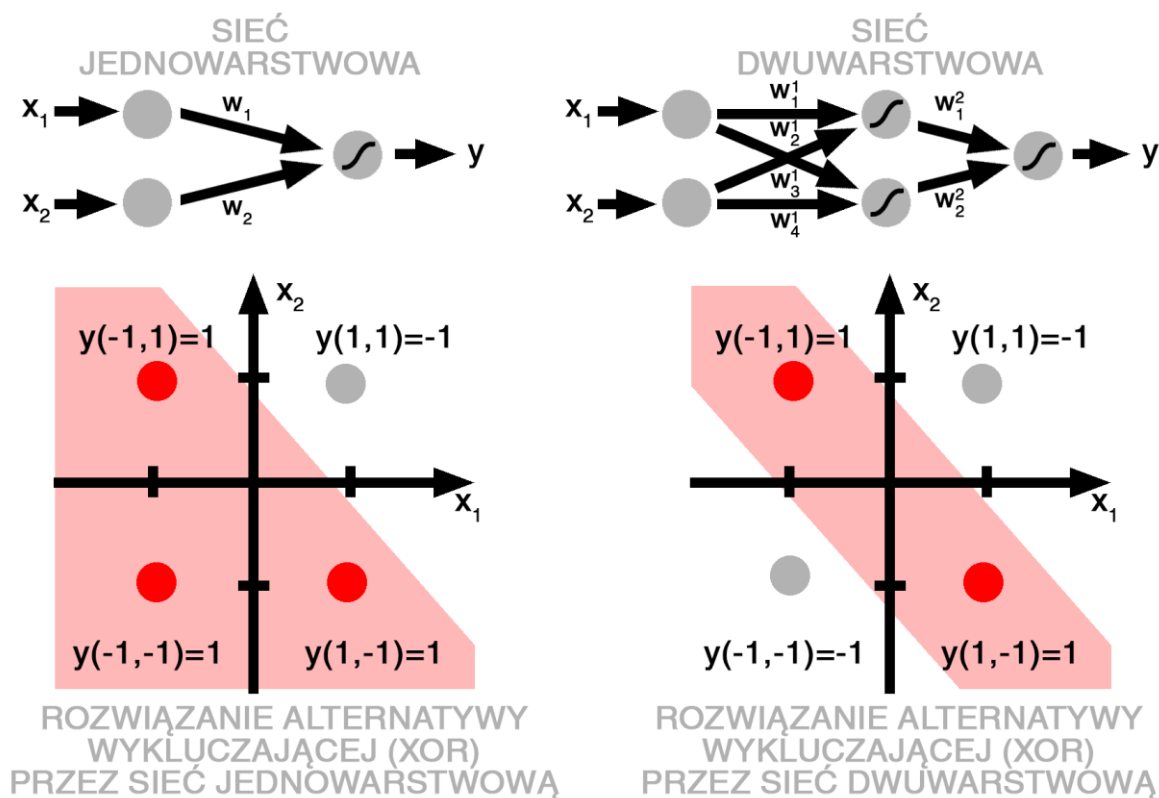
Rys. 31. Szkic Sigmunda Freuda z 1885 roku, będący prawdopodobnie pierwszym koncepcyjnym modelem sieci neuronowej. Rysunek przedstawia przepływ energii od bodźca Q' przez neuron a do neuronu b (Freud, 1950/1895). Teoria Freuda była dosyć niejasna, niepoparta empirycznie i ostatecznie okazała się błędna (McCarley, 1998), jednak stała się jedną z wielu inspiracji dla twórców pierwszych funkcjonalnych sieci neuronowych z lat 50. dwudziestego wieku (Minsky i Sykes, 2017).

We współczesnych sieciach neuronowych połączenia między neuronami działają trochę inaczej. Poszczególne sygnały wejściowe do neuronu są najpierw skalowane przez wagę siły połączenia, a potem są ze sobą sumowane. Jeśli suma osiągnie pewną graniczną wartość, neuron ulega aktywacji. Pierwszą siecią wykorzystującą tę współczesną zasadę był perceptron Rosenblatta opracowany już kilka lat po SNARCu. Początkowo perceptron został opisany teoretycznie jako model biologicznego mózgu (Rosenblatt, 1958), potem został zbudowany w formie fizycznego urządzenia (Rosenblatt, 1961). Perceptron był w praktyce jednowarstwową siecią neuronową, zdolną dokonywać prostej klasyfikacji na podstawie bodźców optycznych. W sieciach neuronowych informacje potrzebne do rozwiązania danego problemu są zapisywane w sile połączeń między danymi wejściowymi, neuronami mogącymi ulec aktywacji i danymi wyjściowymi (Rys. 32).



Rys. 32. Diagram perceptronu elementarnego z jednym neuronem dokonującym klasyfikacji binarnej. Dane wejściowe zespolone są stałym połączeniem z jednostkami asocjacyjnymi „A”. Połączenia między jednostkami „A” a jednostką responsywną „R” są skalowane względem wag, w których zapisana jest „zdolność” perceptronu do rozwiązania danego problemu. Jednostka „R”, którą dzisiaj nazwalibyśmy neuronem, liczy średnią ważoną sygnałów wejściowych i przepuszcza ją przez schodkową funkcję aktywacji. Efektem jest dana wyjściowa: klasa przyjmująca wartość -1 lub 1. Ilustracja przygotowana na podstawie (Rosenblatt, 1961).

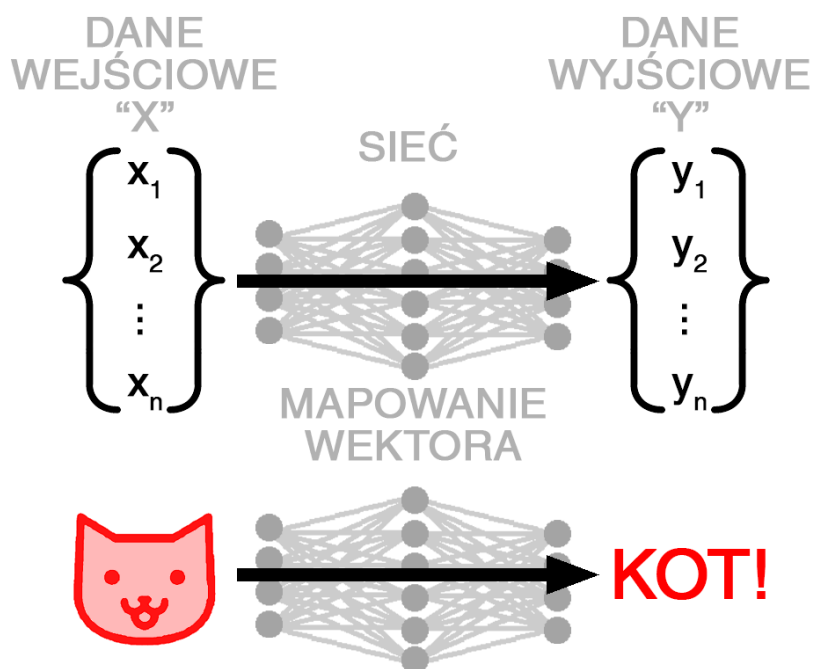
W przypadku perceptronów wagi połączeń mogły być ustalane ręcznie przez operatora na podstawie eksperymentów lub automatycznie przez jeden z możliwych algorytmów zmieniających wagi połączeń w przypadku błędnej klasyfikacji danych treningowych (Hay, i in., 1960; Widrow i Hoff, 1960). W praktyce, algorytmy treningowe były bardzo powolne (czas potrzebny na wytrenowanie perceptronu wzrastał wykładniczo wraz ze wzrostem poziomu trudności problemu) (Olazaran, 1996) i pozwalały na trenowanie jedynie perceptronów jednowarstwowych (Hawkins, 1961). Stanowiło to poważny problem. Główną wadą jednowarstwowego perceptronu Rosenblatta była zdolność do przeprowadzania jedynie liniowej separacji uczonych klas. Innymi słowy, jednowarstwowa sieć neuronowa nie jest w stanie rozwiązać problemu alternatywy wykluczającej (XOR), co bardzo ogranicza jej możliwości (Minsky i Papert, 1969). Konieczne są minimum dwie warstwy, by móc reprezentować alternatywę wykluczającą, a co za tym idzie – mieć aparat do operowania pełną logiką matematyczną (Rys. 33).



Rys. 33. Porównanie rozwiązania problemu alternatywy wykluczającej przez jednowarstwową i dwuwarstwową sieć neuronową. Klas XOR nie da się oddzielić przy pomocy jednej prostej. Jednowarstwową sieć neuronową nieprawidłowo klasyfikuje wyjście y dla $x_1 = -1$ i $x_2 = -1$. Dwuwarstwową sieć neuronową rozwiązuje problem bez trudu. Zamiast aktywacji schodkowej zastosowano jej różniczkowalne przybliżenie: funkcję sigmoidalną. Ilustracja oparta na wynikach symulacji przeprowadzonej w przeglądarkowej implementacji jednokierunkowej sieci neuronowej ConvNetJS (Karpathy, 2014).

Problemy perceptronów jednowarstwowych, trudności z trenowaniem perceptronów wielowarstwowych, ograniczona moc obliczeniowa i zbyt optymistyczne obietnice głoszone przez Rosenblatta doprowadziły w latach 70. do głębokiego zawodu środowiska wczesnego uczenia maszynowego sieciami neuronowymi i zwrócenia się w kierunku poszukiwania symbolicznej sztucznej inteligencji opartej na ściśle zdefiniowanej logice i zasadach heurystycznych (Olazaran, 1996). Tak zwana „pierwsza zima sztucznej inteligencji”, podczas której mało kto zajmował się sieciami neuronowymi, trwała aż do połowy lat 80., kiedy to Geoffrey Hinton spopularyzował metodę, umożliwiającą efektywne trenowanie wielowarstwowych sieci neuronowych – propagację wsteczną. W swoim artykule, *Learning representations by back-propagating errors* opublikowanym w *Nature*, Hinton zdefiniował

podstawy teoretyczne współczesnych, głębokich sieci neuronowych potrafiących samodzielnie uczyć się na podstawie danych treningowych (Hinton, i in., 1986).



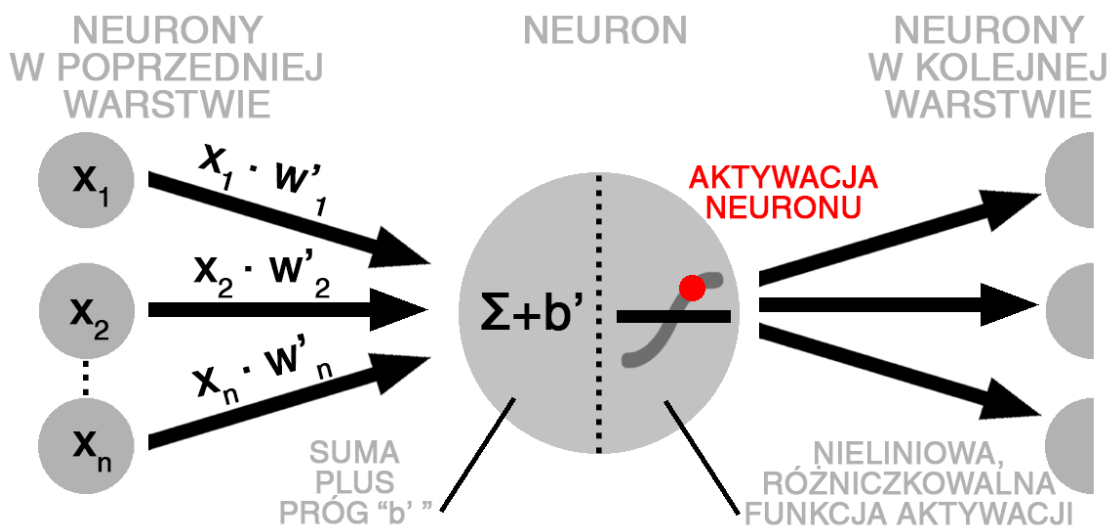
Rys. 34. Sieć neuronowa przekształcająca wektor wejściowy X na wektor wyjściowy Y .

Przykładowo: możemy wytrenowanej sieci pokazać zdjęcie kota i na wyjściu otrzymać prawidłowo zidentyfikowaną etykietę „KOT!”. Opracowanie własne.

Podstawą działania sieci neuronowej jest przetworzenie wielowymiarowego wektora wejściowego X na wielowymiarowy wektor wyjściowy Y . O przetwarzanych wektorach możemy myśleć jako o uporządkowanych ciągach danych liczbowych, albo o strzałkach lub punktach rozpiętych w wielowymiarowej, abstrakcyjnej przestrzeni. W praktyce wektor wejściowy może reprezentować prawie dowolny typ danych – zdjęcie kota (zapisane w formie ciągu wartości jasności pikseli), model GIS-owski miasta czy zwokselizowaną bryłę architektoniczną. Wektor wyjściowy może być interesującą nas klasą obiektu, przybliżeniem funkcji matematycznej czy innym, prawie dowolnym efektem przekształcenia obiektu wejściowego (Rys. 34).

Sieć neuronowa podejmuje decyzje na etapie „propagacji w przód”. W fazie tej informacje przekazane sieci na wejściu przepływają przez kolejne neurony zorganizowane w warstwach. Im więcej warstw ma sieć, tym jest głębsza. W klasycznej, jednokierunkowej sieci neuronowej pojedynczy neuron w danej warstwie połączony jest ze wszystkimi neuronami w warstwie poprzedniej i wszystkimi neuronami w warstwie kolejnej. Neurony w ramach jednej warstwy

nie są ze sobą wzajemnie połączone. Każde połączenie ma przypisaną konkretną wagę (ang. weight). W ramach neuronu najpierw liczona jest suma ważona sygnałów płynących do neuronu z poprzedniej warstwy, potem do sumy dodawany jest próg (ang. bias) i wynik przepuszczany jest przez nieliniową, różniczkowalną funkcję aktywacji (Rys. 35).

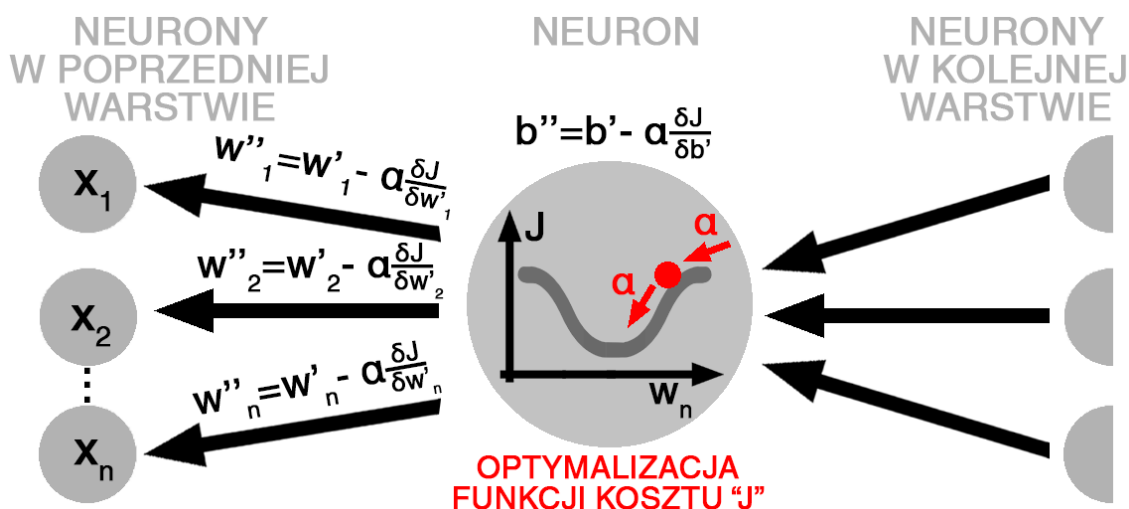


Rys. 35. Diagram propagacji w przód na poziomie pojedynczego neuronu. Sygnały wejściowe x_n sumowane są zgodnie z ich wagami w'_n , do sumy dodawana jest wartość progu b' . Wynik przepuszczany jest przez nieliniową, różniczkowalną funkcję aktywacji. Wartość funkcji stanowi sygnał dla neuronów w kolejnej warstwie. Opracowanie własne na podstawie (Hinton, i in., 1986).

Sieć neuronowa uczy się na etapie „propagacji wstecznej”, w którym progi neuronów i połączenia między konkretnymi neuronami są odpowiednio wzmacniane lub osłabiane w celu optymalizacji rozwiązania zadania. Parametry (wagi i progi) aktualizowane są proporcjonalnie do pochodnej cząstkowej funkcji kosztu²³ (metryka tego, jak źle sieć radzi sobie w rozwiązywaniu danego problemu) względem danego parametru. Dzięki temu, że funkcja aktywacji każdego neuronu jest różniczkowalna, można obliczyć pochodną względem każdego parametru. Z każdym krokiem propagacji wstecznej w czasie treningu funkcja kosztu maleje,

²³ Warto w tym miejscu dokonać rozróżnienia między funkcją kosztu (ang. cost function) i funkcją straty (ang. loss function). Funkcja straty oznacza błąd sieci przy przewidywaniu konkretnego przykładu treningowego, funkcja kosztu oznacza uśredniony błąd sieci na całym zestawie przykładów treningowych. Potocznie oba te pojęcia są używane zamiennie.

a sieć radzi sobie coraz lepiej aż do osiągnięcia minimum lokalnego. Mówimy wtedy, że parametry się zbiegają (ang. convergence) (Rys. 36).



Rys. 36. Diagram propagacji wstecznej na poziomie pojedynczego neuronu. Parametry w''_n oraz b'' są aktualizowane proporcjonalnie do szybkości uczenia α i pochodnej cząstkowej funkcji kosztu J względem danego parametru. Globalnie, z każdym cyklem propagacji wstecznej, funkcja kosztu schodzi w kierunku lokalnego minimum. Im więcej parametrów trenowalnych ma sieć tym więcej wymiarów przestrzennych jest dostępnych na „obejście” minimum lokalnego (minimum lokalne może okazać się w nowym wymiarze hiperprzestrzeni siodłową). Zwiększa się tym samym szansa osiągnięcia minimum globalnego. Opracowanie własne na podstawie (Hinton, i in., 1986).

Dzięki temu, że w głębokiej sieci neuronowej znajdują się tysiące, miliony lub miliardy neuronów ułożonych w wielu warstwach, złożona sieć jest w stanie nauczyć się nawet bardzo skomplikowanych konceptów. Na podstawie danych treningowych sieć buduje wewnętrzną reprezentację mapowania danych wejściowych na dane wyjściowe. Wyuczony przez sieć koncept zapisany jest w wartościach parametrów wag i progów sieci. Zestawienie wielu nieliniowych aktywacji poszczególnych neuronów w kolejnych warstwach pozwala sieci na rozwiązywanie funkcji nieliniowych. Gdyby neurony korzystały z aktywacji liniowych, niezależnie od ilości neuronów i warstw ukrytych cała sieć neuronowa uproszczyłaby się do prostej funkcji liniowej (Ng, 2017). Wprowadzenie nieliniowych aktywacji do sieci neuronowych było oryginalnie inspirowane działaniem biologicznych neuronów w układach nerwowych ludzi i zwierząt. Opracowując swój perceptron, Rosenblatt myślał o nim przede wszystkim jako o modelu ludzkiego mózgu:

„Perceptron jest po pierwsze i przede wszystkim modelem mózgu, a nie wynalazkiem służącym do rozpoznawania wzorców (...). Nie jest to w żadnej mierze model «kompletny» i jesteśmy w pełni świadomi uproszczeń, jakie zastosowaliśmy względem systemów biologicznych, ale jest to przynajmniej model, który można poddać analizie.”
[tłumaczenie własne] (Rosenblatt, 1961, s. VIII)

Według bardzo uproszczonych modeli, biologiczny neuron w stanie niepobudzonym może zostać pobudzony przez odpowiednio silny bodziec, przekazując impuls nerwowy dalej poprzez synapsy do kolejnych neuronów. Aktywacja taka może być w przybliżeniu wymodelowana matematycznie jako funkcja schodkowa (jak w perceptronie) lub, po wygładzeniu, jako funkcja sigmoidalna, która była najpopularniejszą funkcją aktywacji neuronów w sieciach neuronowych do mniej więcej 2010 roku²⁴ (Nair i Hinton, 2010). Obecnie najczęściej korzysta się z funkcji aktywacji ReLU, która w praktyce przyspiesza proces uczenia sieci neuronowej, pozwala na przesyłanie w głąb sieci informacji o intensywności sygnału i sprzyja przyjmowaniu przez parametry wartości zerowej²⁵. Mimo biologicznych inspiracji, syntetyczne sieci neuronowe są w rzeczywistości dosyć dalekie od ich biologicznych odpowiedników. Jak wskazują współczesne badania, do wymodelowania pełnej funkcjonalności pojedynczego neuronu konieczne jest wytrenowanie całej sieci neuronowej, składającej się z tysiąca neuronów syntetycznych uszeregowanych w minimum pięć warstw (Beniaguev, i in., 2021).

Po publikacji Hinton w *Nature* współczesne sieci neuronowe powoli były rozwijane, jednak przez kolejne ćwierć wieku było o nich dosyć cicho ze względu na słabe możliwości obliczeniowe komputerów i nieefektywność stosowanych algorytmów. W połowie

²⁴ Oprócz podobieństwa do funkcji aktywacji neuronu biologicznego, funkcja sigmoidalna ma bardzo dużą zaletę z obliczeniowego punktu widzenia. Pochodna funkcji sigmoidalnej $\sigma(z)$ wynosi $\sigma(z)(1 - \sigma(z))$. Przy obliczaniu pochodnej podczas propagacji wstecznej możemy skorzystać z funkcji $\sigma(z)$, która była już obliczona na etapie propagacji w przód. Znacząco przyspiesza to proces uczenia sieci neuronowej względem funkcji o bardziej skomplikowanych do obliczenia pochodnych.

²⁵ Ang. sparse model (czy „model skąpy”) to taka sieć neuronowa, w której wiele parametrów ma wartość równą zero. Model taki jest lżejszy obliczeniowo, czytelniejszy w interpretacji i łatwiejszy do zapisania na dysku ze względu na możliwość kompresji powtarzających się wielokrotnie zer. W przypadku aktywacji sigmoidalnej parametry mogą zbliżać się asymptotycznie do zera jednak nigdy go nie osiągają, przez co zawsze mają przypisaną jakąś wartość ułamkową.

lat 90. zaczęły pojawiać się pierwsze próby implementacji sieci neuronowych w projektowaniu architektoniczno-urbanistycznym. Stosowane algorytmy ograniczały się do prostych, kilkuwarstwowych sieci złożonych z kilkunastu do kilkudziesięciu neuronów. Szczegółowo zastosowania te zostały omówione w podrozdziale 2.2., ale skrótowo, podejmowane próby obejmowały między innymi modelowanie naturalnej wentylacji wewnątrz budynków w zależności od kierunku wiatru (Kindangen, 1996), modelowanie interakcji przestrzennej na przykładzie wzorców migracji (Black, 1995) lub przemieszczanie się mieszkańców miast do pracy z uwzględnieniem kosztów podróży (Openshaw, 1993). W tej ograniczonej grupie problemów proste sieci neuronowe generowały satysfakcjonujące i precyzyjne rezultaty, szczególnie w porównaniu z konkurencyjnymi metodami opartymi na regresji liniowej czy metodach intuicyjnych. Głównym problemem, na który zwracali uwagę autorzy, były limitacje obliczeniowe sprzętu komputerowego, ograniczające możliwość stosowania większych sieci.

Kolejny przełom w popularności sieci neuronowych nastąpił w 2012 roku, kiedy splotowa sieć neuronowa AlexNet drużyny SuperVision pokonała wszystkie alternatywne metody uczenia maszynowego i metody heurystyczne w konkursie *ImageNet Large Scale Visual Recognition Challenge*, polegającym na rozpoznawaniu i klasyfikacji zdjęć przedstawiających różnego rodzaju obiekty i zwierzęta (Russakovsky, i in., 2015). Moment ten rozpoczął trwającą do dziś „rewolucję głębokiego uczenia maszynowego”. Kluczowymi cechami sieci AlexNet były jej stosunkowo duże rozmiary (650 tysięcy neuronów i 60 milionów parametrów zorganizowanych w 8 warstwach) i nowoczesna metoda implementacji, wykorzystująca procesor graficzny GPU zamiast tradycyjnego procesora CPU, co umożliwiło trening tak dużego modelu w ciągu jedynie pięciu-sześciu dni. W 2012 roku było to sporym osiągnięciem (Krizhevsky, i in., 2012). Demonstracja możliwości głębokiego uczenia maszynowego w syntetycznym przetwarzaniu obrazu przez AlexNet zainspirowała znaczący wzrost popularności sieci neuronowych w architekturze. Dla architektów obraz jest podstawowym medium prezentacji projektu, więc w naturalny sposób sieci neuronowe pobudziły wyobraźnię dużej części środowiska CAAD.

2.5.2. Rodzaje sieci neuronowych wykorzystywanych w CAAD

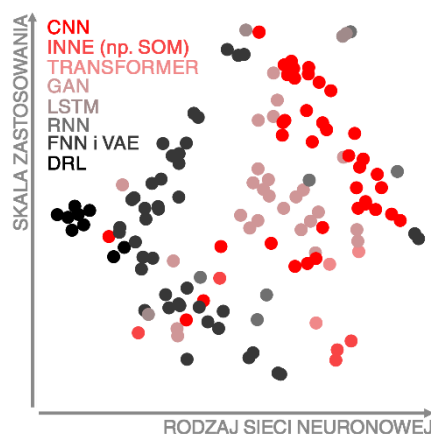
W ostatnich latach pojawiło się wiele rodzajów sieci neuronowych służących rozwiązywaniu problemów różnego typu. Maithani, Arora i Jain widzą możliwość wykorzystania głębokiego uczenia maszynowego tam, gdzie mamy do czynienia ze złożonymi problemami nieliniowymi, opisanymi przez dane multimodalne, tam, gdzie musimy przetwarzać jednocześnie duże ilości

danych wejściowych i tam, gdzie zależy nam na obiektywnym przetwarzaniu nieobarczonym przyjętymi *a priori* założeniami (Maithani, i in., 2010). Oprócz tych czterech podstawowych czynników, na podstawie rozpoznanych w podrozdziale 2.2. zastosowań, można także zwrócić uwagę na dodatkowe zalety sieci neuronowych:

- Bardzo dobrze skalują się do zróżnicowanej liczby danych wejściowych i poziomu trudności przetwarzanych problemów.
- Na współczesnym sprzęcie komputerowym działają względnie szybko, mogą być trenowane i uruchamiane na biurowych i domowych komputerach, na serwerach w chmurze, a nawet na nowoczesnych smartfonach.
- Są proste w implementacji dzięki szerokiej ofercie istniejących środowisk uczenia maszynowego i dużej dostępności kompatybilnego sprzętu.
- Mogą czerpać korzyści z obszernych baz danych dostępnych przez Internet czy zebranych przez systemy Smart-City.



Rys. 37. Rodzaje sieci neuronowych w publikacjach dotyczących komputerowego wspomaganie projektowania architektonicznego. Opracowanie własne.



Rys. 38. Wizualizacja rodzajów sieci neuronowych w publikacjach CAAD wykonana przy pomocy algorytmu t-SNE na podstawie 37 wyodrębnionych cech badanych artykułów. Każdy punkt reprezentuje odrębną publikację. Opracowanie własne.

Warto przyrzeć się najpopularniejszym rodzajom sieci neuronowych wykorzystywanych w projektowaniu architektonicznym wspomaganym komputerowo w celu poznania podstaw ich działania i rozpoznania możliwości ich zastosowania z punktu widzenia projektanta chcącego rozszerzyć swój warsztat o narzędzia oparte na głębokim uczeniu maszynowym. W tym celu ponownie przeanalizowałem wyniki kwerendy zastosowań głębokiego uczenia maszynowego w CAAD omówione w sekcji 2.2. Tym razem jednak podzieliłem zebrane publikacje względem typu zastosowanej sieci neuronowej (Rys. 37, Rys. 38).

Jednokierunkowe sieci neuronowe (FNN)

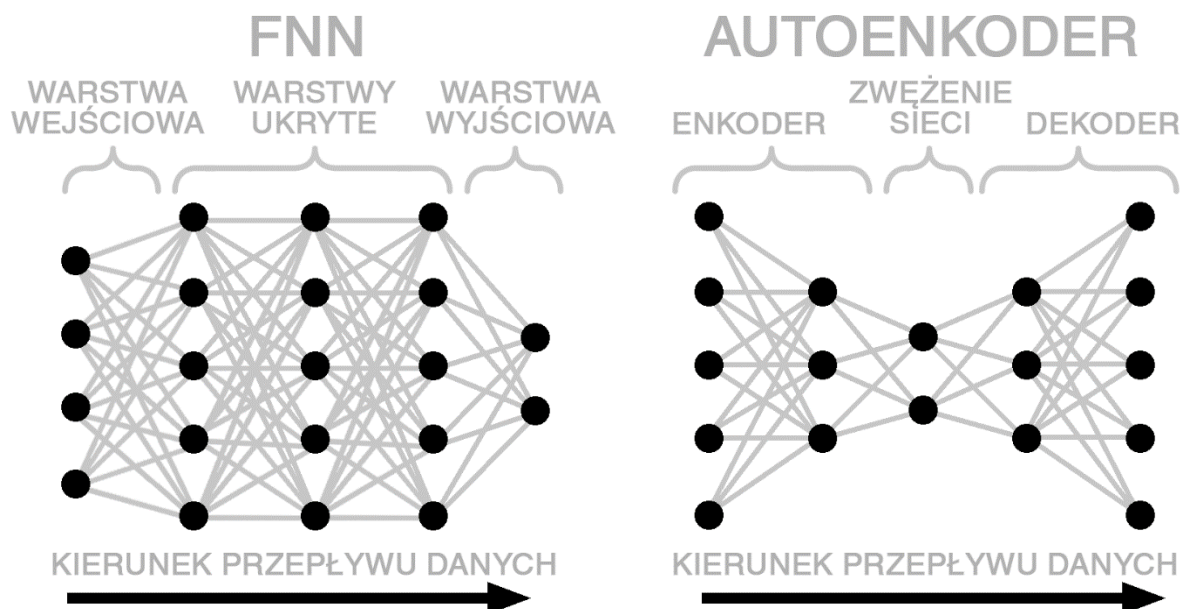
Najprostszym i najstarszym typem sieci neuronowych są sieci FNN, czyli jednokierunkowe sieci neuronowe o warstwach w pełni połączonych (ang. Fully-Connected Feedforward Neural Network), co znaczy, że każdy neuron jednej warstwy jest połączony z każdym neuronem warstwy kolejnej i każdym neuronem warstwy poprzedniej. W FNN sygnał przepływa w sieci jednokierunkowo od warstwy wejściowej przez warstwy ukryte po warstwę wyjściową. Sieci jednokierunkowe różnią się od siebie liczbą warstw oraz liczbą znajdujących się w nich neuronów.

Sieci jednokierunkowe mają szeroki zakres zastosowań tam, gdzie bezpośrednio można dokonać mapowania pewnego wektora wejściowego na odpowiadający mu wektor wyjściowy. W przeanalizowanych w sekcji 2.2. publikacjach FNN często wykorzystywane są do przybliżania wyników złożonych symulacji komputerowych w celu oszczędności czasu i zmniejszenia zapotrzebowania na moc obliczeniową. Przykładowo, przy pomocy FNN można przewidywać wyniki symulacji wydajności środowiskowej i energetycznej elewacji (Sebestyen i Tyc, 2020) lub całych budynków (Stevenson, 1994b; Asl, i in., 2017; Singh, i in., 2020). Przybliżać można także metryki normalnie dostępne jedynie metodami pomiarowymi w już wybudowanym obiekcie, w fizycznym prototypie lub modelu. FNN wykorzystywane są w przybliżaniu wartości metryk komfortu wizualnego stanowisk pracy w przestrzeniach biurowych (Chatzikonstantinou i Sariyildiz, 2016) czy w przewidywaniu kosztów i konsekwencji środowiskowych projektowanej modernizacji obiektu (Nicholas, i in., 2021). Na podstawie parametrów opisujących projekt lub budynek istniejący oraz jego lokalizację można także przewidywać cenę nieruchomości (Lam, i in., 2008; Abidoeye i Chan, 2017), koszty cyklu życia budynku (Gao i Pishdad-Bozorgi, 2019) czy żywotność jego elewacji (Silva, i in., 2011; Dias, i in., 2013). FNN dobrze radzą sobie też w prostych problemach klasyfikacyjnych danych wejściowych przedstawionych w postaci numerycznej. Można przy ich pomocy

klasyfikować typologicznie przestrzenie we wnętrzach budynków (Sedlmeier i Feld, 2018) lub rozpoznawać pomieszczenia i pozycję we wnętrzu na podstawie zbioru danych nawigacyjnych (Mehmood i Tripathi, 2013; Tamas i Toth, 2018). Jednokierunkowe sieci neuronowe mogą służyć optymalizacji parametrów projektu względem interesującej projektanta metryki, jak wydajności energetycznej obiektu (Li i Chen, 2021) lub wydajności wentylacji naturalnej wewnątrz budynku (Kindangen, 1996). FNN mogą także przetwarzać sensoryczne dane pomiarowe pomagając w ich interpretacji. Przykładowo, możliwe jest rozpoznawanie liczby użytkowników przestrzeni publicznej na podstawie sensorów znajdujących się w chodnikach (Kirova i Markopoulou, 2020), przetwarzanie surowych trajektorii GPS na wzorce aktywności w przestrzeni miejskiej (Siła-Nowicka, i in., 2016), identyfikowanie terenów nowozurbanizowanych poprzez pomiary satelitarne (Liu i Lathrop Jr, 2002) czy rozpoznawanie ugięcia elastycznych materiałów budowlanych na podstawie mierzonej oporności prądu w odkształconej membranie (Koshelyuk, i in., 2019). Jednokierunkowe sieci neuronowe sprawdzają się także jako samodzielne, proste algorytmy generatywne, przetwarzające parametry wejściowe na geometrię (Cudzik i Radziszewski, 2018; Yetkin i Sorguç, 2019; Zheng, i in., 2020).

Szczególnym typem jednokierunkowej sieci w pełni połączonej są autoenkodery (ang. Autoencoders), w których kolejne warstwy sieci mają stopniowo coraz mniej neuronów, aż do zwężenia w połowie głębokości sieci (ang. Bottleneck). Po zwężeniu warstwy stopniowo się rozszerzają aż do osiągnięcia rozmiarów warstwy wejściowej. Sieć taką można wyobrazić sobie jako klepsydę, w której pierwszą połowę nazywamy enkoderem, drugą – dekoderem (Rys. 39). Autoenkoder uczy się kompresować dane wejściowe do jej mniej wymiarowej reprezentacji w przestrzeni ukrytej (ang. Latent Space) definiowanej przez parametry najwęższej warstwy w centrum sieci. Dekoder uczy się odtwarzać dane wejściowe na podstawie ich skompresowanej reprezentacji w przestrzeni ukrytej. Struktura taka może być szczególnie przydatna w eksploracji przestrzeni projektowej modeli parametrycznych. Na reprezentacjach projektów w przestrzeni ukrytej można przeprowadzać operacje semantyczne [np. odejmować reprezentację wektorową jednej geometrii od reprezentacji innej geometrii (Sebestyen, i in., 2021)], a eksploracja ciągłej przestrzeni ukrytej pozwala na odnajdywanie nowych form przestrzennych stworzonych w oparciu o interpolację między formami zadanymi (de Miguel, i in., 2019). Możliwe wariacje parametrów projektu mapowane są na przestrzeń o stosunkowo niewielkiej liczbie wymiarów. Dzięki znacznej redukcji liczby wymiarów względem warstwy wejściowej, analiza wzorców aktywacji neuronów w przestrzeni ukrytej

może być także pomocna w identyfikacji i interpretacji najważniejszych parametrów przetwarzanego problemu. Wykorzystanie autoenkoderów w ten sposób może być szczególnie korzystne w sytuacjach, w których alternatywne metody redukcji wymiarów, takie jak PCA (Pearson, 1901) czy t-SNE (Hinton i van der Maaten, 2008) nie są wystarczające.



Rys. 39. Diagram klasycznej, jednokierunkowej sieci w pełni połączonej (po lewej) oraz sieć typu autoenkoder (po prawej). Wykorzystywane w praktyce sieci zazwyczaj mają więcej neuronów oraz warstw. Opracowanie własne.

Splotowe sieci neuronowe (CNN)

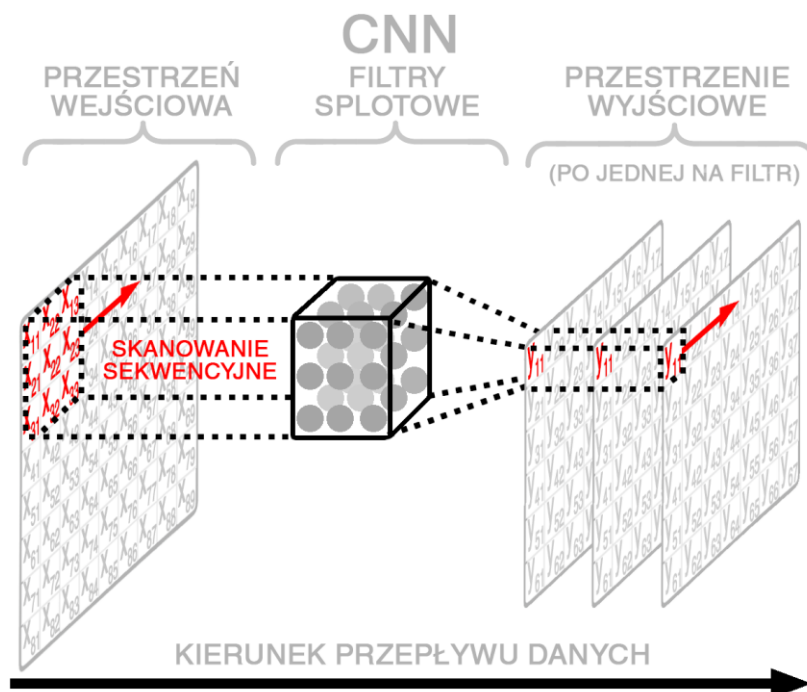
Kolejnym, często wykorzystywanym w kontekstach architektoniczno-urbanistycznych typem sieci neuronowych są CNN, czyli splotowe sieci neuronowe (ang. Convolutional Neural Networks). W sieciach splotowych zamiast warstw w pełni połączonych stosowanych w FNN wykorzystuje się warstwy splotowe złożone z filtrów splotowych połączonych jednocześnie tylko do części aktywacji warstwy poprzedzającej. Filtry splotowe pokrywają całość przestrzeni wejściowej przesuwając się po niej sekwencyjnie. Dzięki temu rozwiązaniu jeden wytrenowany filtr może rozpoznawać wiele podobnych cech w różnych rejonach przestrzeni wejściowej (Rys. 40). Splotowe sieci neuronowe zostały stworzone z myślą o syntetycznym przetwarzaniu obrazów, rozpoznawaniu w nich wzorów i ich klasyfikacji (Fukushima, 1980; LeCun, i in., 1989; Krizhevsky, i in., 2012). Dzięki sekwencyjnemu skanowaniu danych wejściowych przez filtry sieci typu CNN wykazują przestrzenne uprzedzenie dotyczące

struktury danych [ang. Spatial Inductive Bias (Mitchell, 2017)]. Oznacza to, że sieci splotowe szczególnie dobrze radzą sobie z danymi powiązanymi przestrzennie (takimi jak obrazy, chmury punktów czy wielowymiarowe diagramy). Wydaje się, że dzięki temu bardzo dobrze nadają się do przetwarzania architektoniczno-urbanistycznych kontekstów przestrzennych (Dzieduszyński, 2022b). W splotowych sieciach neuronowych oprócz warstw splotowych wykorzystywane są także warstwy innego typu, np. warstwy w pełni połączone. Sieci, w których występują jedynie warstwy splotowe, nazywamy sieciami w pełni splotowymi (FCN, ang. Fully-Convolutional Network). Jedną z zalet takich modeli jest niezależność struktury sieci i liczby trenowalnych parametrów od wymiarów danych wejściowych. Przykładowo, FCN mogą być wykorzystywane do przetwarzania obrazów wejściowych o różnych rozmiarach i proporcjach.

Ze względu na duży wybór gotowych rozwiązań do przetwarzania obrazów, w CAAD sieci splotowe najczęściej stosowane są do przetwarzania danych przedstawionych w formie graficznej, takich jak: zdjęcia panoramiczne ulic i wnętrz (Zhang, i in., 2020; Yazıcıoğlu i Dino, 2021; Qi, i in., 2020; Kinugawa i Takizawa, 2019; Law, i in., 2018; Fukuda, i in., 2017; Sun, i in., 2019), zdjęcia satelitarne i lotnicze (Meeran i Joyce, 2020; Iino, i in., 2018; Ikeno, i in., 2020; Newton, i in., 2020; Newton, 2021; Li i Zhu, 2020; Qayyum, i in., 2019; Pan, i in., 2021), obrazy z kamer monitoringu (Sukel, i in., 2019; Sukel, i in., 2020a; Nakabayashi, i in., 2021; Jørgensen, i in., 2020), rzuty, przekroje i inne rysunki architektoniczne (Liu, i in., 2020; Ng, i in., 2019; Çolakoğlu i Uzun, 2019; Xiao, i in., 2020; Brown, i in., 2020; Sharma, i in., 2017) czy stworzone cyfrowo wizualizacje modeli trójwymiarowych (Takizawa i Furuta, 2017). Niektórzy wykraczają jednak poza łatwo dostępne modele służące analizie obrazów i wykorzystują CNN do przetwarzania danych architektonicznych reprezentowanych w inny sposób. Przykładowo sieci splotowe dobrze radzą sobie z projektami przedstawionymi w formie grafów reprezentujących układ funkcjonalny lub przestrzenny obiektu (Eisenstadt, i in., 2019; As, i in., 2018; Zhang, 2020), sparametryzowanymi modelami przedstawionymi w postaci grafów (del Campo, i in., 2020), chmurami punktów (Stojanovic, i in., 2019; Sinke, i in., 2021), czy reprezentacjami isovist (Peng, i in., 2017).

Mimo rosnącej obecnie popularności nowszych typów sieci neuronowych, takich jak transformery wizualne czy modele dyfuzyjne, proste sieci splotowe nadal stosowane są bardzo często (Rhee, i in., 2023) i po odpowiednim skalowaniu pozwalają osiągnąć wyniki porównywalne z tymi oferowanymi przez nowsze, bardziej skomplikowane algorytmy

(Liu, i in., 2022). Wśród najpopularniejszych modeli splotowych sieci neuronowych stosowanych w CAAD warto wyróżnić: AlexNet (Krizhevsky, i in., 2012), VGG (Zisserman i Simonyan, 2015), ResNet (He, i in., 2016), U-Net czy ConvNeXt (Liu, i in., 2022; David i Leita, 2022).



Rys. 40. Diagram warstwy splotowej w splotowej sieci neuronowej. Dane wejściowe skanowane są sekwencyjnie przez filtry splotowe obliczające wartości funkcji splotu na podstawie odczytanych sygnałów. Dane na wyjściu mają formę wielowymiarowych tablic, po jednej tablicy na jeden filtr splotowy. Trenowalne parametry znajdują się w filtrach splotowych. Opracowanie własne.

Generatywne sieci przeciwstawne (GAN)

Podobnie jak architekci, badacze i inżynierowie działający w dziedzinie uczenia maszynowego często wykorzystują dostępne im prostsze moduły w celu składania z nich większych całości, oferujących szersze możliwości niż same elementy składowe. Jednym z przykładów takich kompleksowych rozwiązań są GANy, czyli generatywne sieci przeciwstawne (ang. Generative Adversarial Networks) (Goodfellow, i in., 2014). GANy są algorytmami złożonymi z dwóch konkurujących ze sobą sieci neuronowych. Pierwsza z nich to generator odpowiedzialny za tworzenie danych jak najlepiej wpisujących się w dystrybucję zbioru treningowego, druga z nich to dyskryminator próbujący odróżnić oryginalne dane treningowe od tych stworzonych

przez generator. W trakcie treningu obie sieci uczą się wspólnie. W miarę jak dyskryminator coraz trafniej odróżnia dane syntetyczne od prawdziwych, generator tworzy dane coraz bardziej podobne do danych treningowych. Gdy GAN jest już wytrenowany, do przetwarzania danych zazwyczaj używa się tylko generatora. Jako generatory w GANach najczęściej wykorzystywane są sieci CNN (np. U-Net) (Isola, i in., 2017).

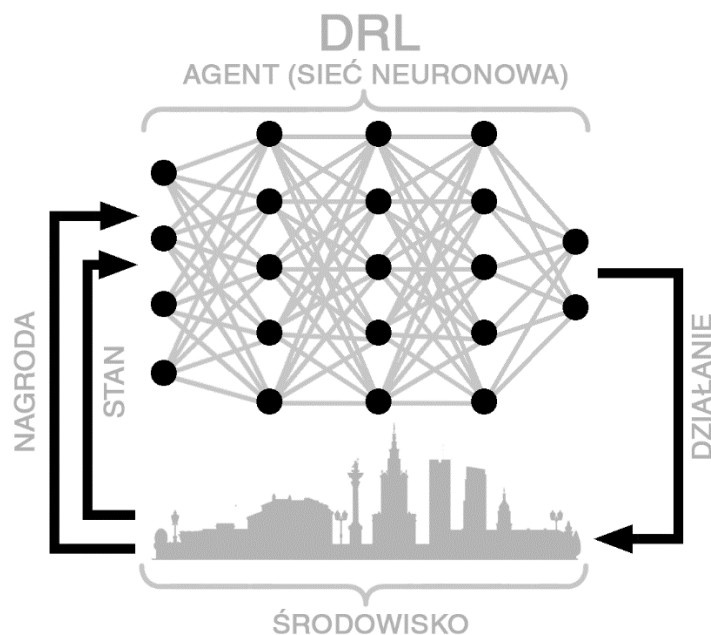
W CAAD GANy używane są zazwyczaj tam, gdzie na podstawie pewnych warunków wejściowych chcemy wygenerować realistycznie wyglądające dane (np. obrazy), trafnie wpisujące się w dystrybucję treningową. Przykładowo, GANy można wykorzystać do wizualizacji projektów poprzez naniesienie na zdjęcia elewacji projektowanych elementów (Nicholas, i in., 2021) czy też usuwanie i maskowanie na zdjęciach elementów, które w planach mają być rozebrane lub wyburzone (Kikuchi, i in., 2021). GANy stosowane są do odczytywania ukrytych informacji z kontekstów przestrzennych (przy założeniu, że interesujące nas informacje są skorelowane z prezentowanymi danymi wejściowymi). Badacze podejmowali próby przewidywania aktywności fizycznej na projektowanych terenach na podstawie map GIS (Sun, i in., 2020), odczytywania funkcji pomieszczeń z rzutów architektonicznych (Zheng i Huang, 2018), czy nawet odczytywania *genius loci* określonej okolicy ze zbioru panoramicznych zdjęć Street View (Steinfeld, 2019). GANy można też wykorzystać do proponowania docelowych rozwiązań projektowych, takich jak: systemy instalacyjne wpisujące się w zadany rzut architektoniczny (Sato, i in., 2020), koncepcyjne rozkłady funkcjonalne (Eisenstadt, i in., 2019; As, i in., 2018), podziały funkcjonalne projektowanych kwartałów (Tian, 2020) czy rozwiązania detali (Alani i Al-Kaseem, 2021; Zandavali i Garcia, 2019; Thomsen, i in., 2019).

Najpopularniejszymi modelami GANów stosowanych w CAAD są Pix2Pix (Isola, i in., 2017), StyleGAN (Karras, i in., 2021) i StyleGAN2 (Karras, i in., 2020).

Głębokie uczenie ze wzmocnieniem (DRL)

Głębokie uczenie ze wzmocnieniem DRL (ang. Deep Reinforcement Learning) stanowi wariant uczenia ze wzmocnieniem, w którym agent sterowany przez sieć neuronową, lub grupa takich agentów, podejmują akcje w środowisku symulacyjnym. Sterującą agentem sieć neuronowa uczy się podejmować decyzje zapewniające jak najwyższą nagrodę (Rys. 41). Ostatnimi czasy największe sukcesy w dziedzinie DRL osiągał zespół Google Deepmind. Opracowany przez nich system AlphaGo osiągnął mistrzowski poziom w grze planszowej Go (Silver, i in., 2016), AlphaStar potrafił biegle grać w grę komputerową *Starcraft 2* (Vinyals, i in., 2019),

a AlphaFold perfekcyjnie przewidywał trójwymiarowe struktury skomplikowanych białek na podstawie sekwencji ich aminokwasów (Jumper, i in., 2021).



Rys. 41. Diagram ilustrujący pętlę działania agenta DRL w środowisku symulacyjnym.

Działanie agenta zostaje nagrodzone przez funkcję straty i skutkuje zmianą jego stanu.

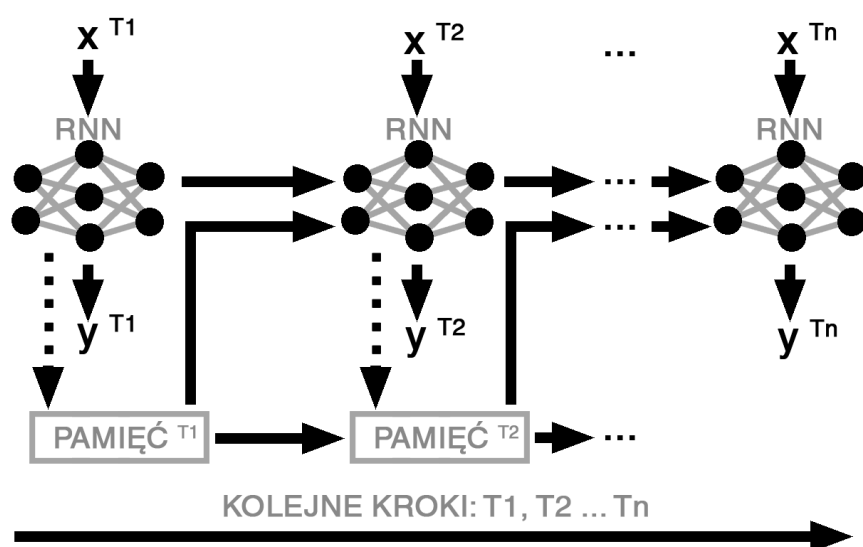
Opracowanie własne.

W problemach architektonicznych DRL dobrze radzi sobie tam, gdzie w środowisku symulacyjnym można wymodelować zachowanie rzeczywistych fenomenów projektowych, wykonawczych czy eksploatacyjnych. W aplikacjach tego typu wytrenowani symulacyjnie agenci sterowani przez sieci neuronowe reprezentować mogą dynamicznie sterowane ustroje konstrukcyjne (Hosmer i Tigas, 2019), charakterystykę środowiskową obiektu budowlanego (Song i Yuan, 2021), poszczególne pomieszczenia w budynkach mieszkalnych (Velošo i Krishnamurti, 2020) czy autonomiczne drony odpowiedzialne za cyfrową fabrykację obiektów budowlanych (Fang, i in., 2020). Środowiska wykorzystywane w DRL mogą być także bardziej abstrakcyjne i niekoniecznie muszą reprezentować środowisko fizyczne. Głębokie uczenie ze wzmacnianiem może być wykorzystywane jako podstawa algorytmów generatywnych poszukujących najlepszych rozwiązań w przestrzeni projektowej (Han, i in., 2020). W takich symulacjach agenci starają się optymalizować podejmowanie decyzji projektowych (Mintrone i Erioli, 2021), wybierając przykładowo najlepsze do sytuacji moduły architektoniczne zaprojektowane innymi metodami (Hosmer, i in., 2020).

Algorytmy DRL wykorzystywane w CAAD zazwyczaj nie korzystają z gotowych, ogólnodostępnych rozwiązań. Większość z nich projektowana jest bezpośrednio do rozwiązywanego problemu. Do tworzenia środowisk symulacyjnych najczęściej stosowane są platformy Rhino Grasshopper, Unity 3D lub Python TensorFlow. Agenci zazwyczaj opierają się na technikach Q-Learning (Watkins i Dayan, 1992), DDPG [ang. Deep Deterministic Policy Gradient (Lillicrap, i in., 2015)] lub PPO [ang. Proximal Policy Optimization (Schulman, i in., 2017)].

Rekurencyjne sieci neuronowe (RNN) i sieci długiej pamięci krótkotrwałej (LSTM)

Współcześnie jedne z największych i najbardziej imponujących sukcesów w dziedzinie głębokiego uczenia maszynowego osiągane są na polu przetwarzania języka naturalnego NLP (ang. Natural Language Processing). Jednymi z pierwszych modeli sieci neuronowych potrafiących skutecznie przetwarzać ciągi długiego tekstu były rekurencyjne sieci neuronowe RNN (ang. Recurrent Neural Networks) i ich wariant, czyli sieci długiej pamięci krótkotrwałej LSTM (ang. Long Short-Term Memory Networks). Sieci te operują na danych sekwencyjnych takich jak tekst, dźwięk, uporządkowane ciągi czy szeregi czasowe. W klasycznych RNN sieć otrzymuje sekwencję danych wejściowych podzielonych na pojedyncze pakiety. Oprócz nich sieć dostaje także wartości stanów ukrytych obliczonych przez tę samą sieć w poprzednich krokach. Dzięki temu połączeniu rekurencyjnemu sieci łatwiej jest nauczyć się sekwencyjnych zależności w analizowanych danych (np. zasad gramatycznych występujących w ciągu tekstowym. Proste RNN bywają jednak dosyć niestabilne ze względu na problem z eksplodowaniem lub zanikaniem sygnałów wraz z kolejnymi krokami w dłuższych sekwencjach danych wejściowych (Tadeusiewicz i Szaleniec, 2015). Jednym z rozwiązań tego problemu jest wyposażenie sieci w mechanizm pamięci krótkotrwałej. Pamięć taka pozwala sieci LSTM na zapisywanie najważniejszych informacji do wykorzystania w przyszłych krokach (Rys. 42). Dzięki pamięci sieci typu LSTM o wiele lepiej radzą sobie z przetwarzaniem dłuższych ciągów danych (Hochreiter i Schmidhuber, 1997).



Rys. 42. Diagram ilustrujący sposób działania sieci RNN typu LSTM. W następujących po sobie krokach sieć otrzymuje wejściowe dane z sekwencji wejściowej X i przetwarza je na sekwencję wyjściową Y . Dodatkowo sieć otrzymuje na wejściu stany ukryte sieci z poprzedniego kroku. W modelu LSTM sieć otrzymuje także mechanizm pamięci, pozwalający jej nauczyć się możliwości zapisywania pewnych istotnych danych w komórkach pamięci, których stan sieć może odczytać w kolejnych krokach iteracji.

Opracowanie własne.

Obecnie istnieją nowsze i bardziej skuteczne modele do przetwarzania danych sekwencyjnych (omówione w kolejnym punkcie), jednak w projektowaniu architektonicznym wspomagany komputerowo RNN i LSTM nadal są czasem wykorzystywane. LSTM dobrze uczą się szeregów czasowych, więc potrafią względnie trafnie przewidywać przyszłe stany procesów ciągłych. Wykorzystuje się je między innymi do przewidywania zatorów komunikacyjnych na ulicach miast (Guo, i in., 2019), rozpoznawania przyszłych trajektorii użytkowników przestrzeni miejskiej (Wu, i in., 2019) czy identyfikacji potencjalnych obszarów objętych rozrostem tkanki urbanistycznej (Boulila, i in., 2021). Sieci rekurencyjne nadają się do sterowania ciągłymi procesami fabrykacji robotycznej reprezentowanej przez szereg czasowy instrukcji narzędzi CAM (Luo, i in., 2018; Chen, i in., 2019; del Castillo y López, 2019). Mogą także robić to, do czego były pierwotnie zaprojektowane, czyli przetwarzać tekst naturalny. Przykładowo, mogą być składowymi systemów klasyfikacji zgłoszeń problemów urbanistycznych wysyłanych przez mieszkańców w formularzach kontaktowych administracji miast (Sukel, i in., 2020a). W niektórych przypadkach dane wejściowe trudno jest przetwarzać w ich pierwotnej formie. Niekiedy warto przetworzyć je wtedy do formy sekwencyjnej.

Przykładowo, w celu odczytania z dwuwymiarowych zdjęć panoramicznych wnętrza mieszkań informacji o położeniu krawędzi ścian obraz wejściowy można przedstawić jako ciąg kolumn jego pikseli składowych. Możliwe jest wtedy wykorzystanie zdolności LSTM do rozpoznawania zależności sekwencyjnych do rozpoznawania narożników w pomieszczeniach. Przy takim przedstawieniu danych LSTM łatwo uczy się przykładowo, że pomieszczenia są zazwyczaj prostopadłościanami, więc odległość między narożnikami pierwszym a drugim często bywa zbliżona do odległości między narożnikiem trzecim a czwartym. System taki potrafi znajdować narożniki nawet wtedy, gdy są one częściowo przesłonięte przez wyposażenie wnętrza (Sun, i in., 2019). Innym przykładem takiego podejścia jest potraktowanie projektu parametrycznego obiektu jako pewnego rodzaju „zdania” złożonego z sekwencyjnie ułożonych bloków instrukcji możliwego do analizy przy pomocy LSTM. Sieć pozwala wtedy na eksplorację przestrzeni projektowej poprzez sugerowanie alternatywnych dróg przepływu danych w modelu parametrycznym (Toulkeridou, 2019).

Podobnie jak w przypadku DRL, proste sieci RNN i LSTM zazwyczaj przygotowywane są od zera na potrzeby konkretnego problemu. Dostrajanie wcześniej wytrenowanych modeli nie jest często stosowane. Z najpopularniejszych typów prostych sieci rekurencyjnych warto wymienić dwukierunkowe RNN i LSTM (ang. Bidirectional RNNs/LSTMs), skanujące sekwencję wejściową w obu kierunkach, oraz sieci GRU (ang. Gated Recurrent Unit Networks), będące uproszczonym wariantem sieci LSTM (Cho, i in., 2014).

Inne sieci neuronowe, transformery, modele dyfuzyjne

Chcąc ustalić, w którym momencie rozpoczęła się trwająca obecnie rewolucja głębokiego uczenia maszynowego można wskazać przynajmniej kilka dat:

- 1951: stworzenie przez Marviną Minsky’ego SNARCa, czyli pierwszej funkcjonalnej sieci neuronowej (Minsky i Sykes, 2011).
- 1986: publikacja Rumelharta, Hintona i Williamsa w *Nature* popularyzująca propagację wsteczną umożliwiającą projektowanie samouczących się sieci neuronowych (Hinton, i in., 1986).
- 2012: sieć AlexNet Alexa Krizhevskiego wprowadzająca splotowe sieci neuronowe do głównego nurtu badań (Krizhevsky, i in., 2012).

Najnowszym pretendentem jest rok 2017, czyli moment publikacji artykułu *Attention Is All You Need*, w którym zaprezentowano nowy typ sieci neuronowej – transformer (Vaswani, i in.,

2017). To właśnie na transformerach oparta jest większość dużych modeli językowych LLM (ang. Large Language Model) zdobywających szturmem kolejne miliony użytkowników na całym świecie (Hu, 2023). Transformery są sieciami neuronowymi wyposażonymi w mechanizm uwagi, dzięki któremu na każdym etapie generowania danych wyjściowych sieć może zwracać uwagę na kontekst znajdujący się w różnych miejscach sekwencji wejściowej oraz we fragmentach sekwencji wygenerowanych przez model w poprzednich krokach (Vaswani, i in., 2017). Transformery najczęściej wykorzystywane są w NLP, ale są one także stosowane do przetwarzania obrazów przedstawionych w postaci sekwencyjnej. Sieci takie nazywamy transformerami wizualnymi ViT (ang. Vision Transformer) (Dosovitskiy, i in., 2021; Radford, i in., 2021). W CAAD transformery często wykorzystuje się do tzw. analizy sentymentu (ang. sentiment analysis) polegającej na ocenie, czy dany tekst jest nacechowany pozytywnie czy negatywnie. Wyniki analizy sentymentu tekstów pochodzących przykładowo z mediów społecznościowych powiązanych z ich danymi geolokacyjnymi pozwalają na przeprowadzanie analiz urbanistycznych (Aman, i in., 2022). Dodatkowo, ze względu na ich zdolności generalizacyjne, duże modele językowe takie jak GPT4 (OpenAI, 2023), Llama 2 (Meta AI, 2023) czy Gemini (Google, Gemini Team, 2023) zaczynają być wykorzystywane w biurach projektowych jako syntetyczni asystenci wspierający architektów w oprogramowywaniu rozwiązań, poszukiwaniu inspiracji i przetwarzaniu danych. Dostępność LLM w postaci tanich lub darmowych API i interfejsów czatowych zwiększa liczbę ich potencjalnych użytkowników o projektantów bez odpowiedniego przygotowania informatycznego i o biura projektowe niedysponujące odpowiednio dużą mocą obliczeniową, pozwalającą na korzystanie z własnych modeli głębokiego uczenia maszynowego offline.

Oprócz transformerów dużą popularnością cieszą się obecnie generatywne modele dyfuzyjne służące do tworzenia z szumu obrazów i innych typów danych (np. wideo lub audio). Algorytmy dyfuzyjne stopniowo usuwają szum progresywnie, generując coraz lepiej zdefiniowane dane wyjściowe. Modele dyfuzyjne często oparte są na transformerach wizualnych, ale mogą wykorzystywać także prostsze sieci, jak np. CNN (Ho., i in., 2020). Podobnie jak w przypadku LLM, duże modele dyfuzyjne takie jak Dall-E 2 (Ramesh, i in., 2022), Stable Diffusion (Rombach, i in., 2022) czy Disco Diffusion (Alembics, 2021) dostępne są w postaci prostych w użytkowaniu interfejsów online, dzięki czemu weszły już do wielu biur projektowych jako narzędzia służące poszukiwaniu rozwiązań i wizualizacji projektów (RIBA, 2024).

3. Część eksperymentalna

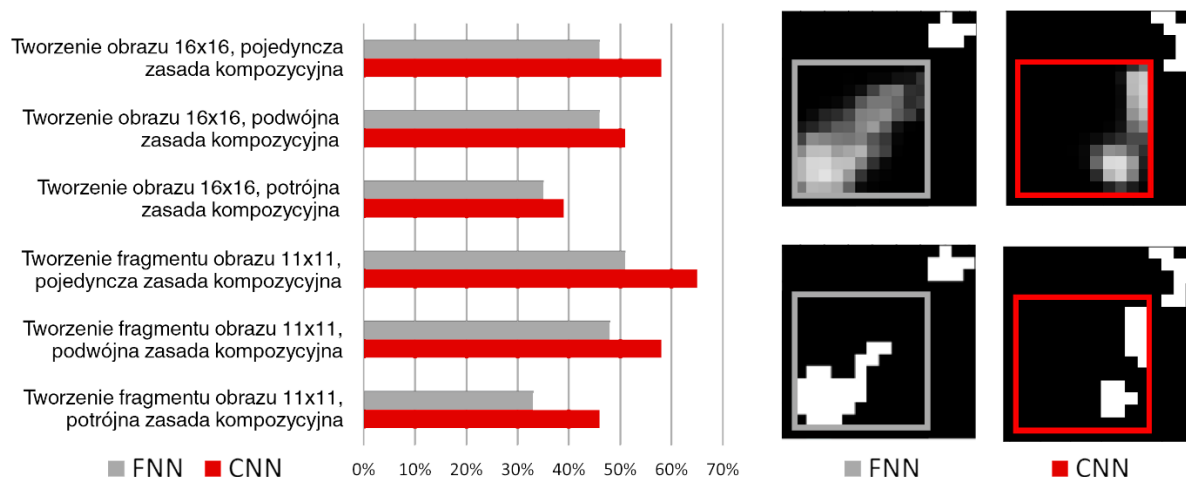
3.1. Eksperyment wstępny

3.1.1. Cele szczegółowe eksperymentu wstępnego

Wnioski płynące z kwerendy stanu wiedzy sugerują, że prawidłowo zaprojektowane, odpowiednio duże sieci neuronowe powinny dobrze radzić sobie w przetwarzaniu przestrzennych kontekstów kompozycyjnych. Co więcej, wydaje się, że przynajmniej w części zaprezentowanych zastosowań głębokiego uczenia maszynowego w CAAD i dziedzinach pokrewnych konteksty kompozycyjne są w jakimś stopniu uwzględniane przez stosowane algorytmy. Względy kompozycyjne nie zostały jednak do tej pory przebadane w oderwaniu od innych aspektów neuronowego przetwarzania danych. Do momentu przeprowadzenia eksperymentu wstępnego nie było także jasne, do jakiego stopnia złożone, rozmyte logicznie zasady kompozycyjne, stosowane zazwyczaj przez ludzi, mogą być odczytywane, przetwarzane i aplikowane w praktyce przez maszyny. Eksperyment wstępny miał stanowić weryfikację koncepcji, operującą na abstrakcyjnych i mierzalnych zasadach kompozycyjnych, przed opracowaniem docelowego narzędzia wykorzystującego głębokie uczenie maszynowe do ekstrakcji rzeczywistych elementów kompozycyjnych w przestrzeni architektoniczno-urbanistycznej. W celu odizolowania problemu kompozycji przestrzennej od innych zmiennych, przygotowałem eksperyment w sposób, który umożliwił jakościową i ilościową analizę skuteczności zastosowania prostych sieci neuronowych w przetwarzaniu kontekstów kompozycyjnych. W poniższym podrozdziale znajduje się sumaryczne podsumowanie przebiegu eksperymentu. W szczegółach został on opisany w *International Journal of Architectural Computing* (Dzieduszyński, 2022a).

3.1.2. Metodyka eksperymentu wstępnego

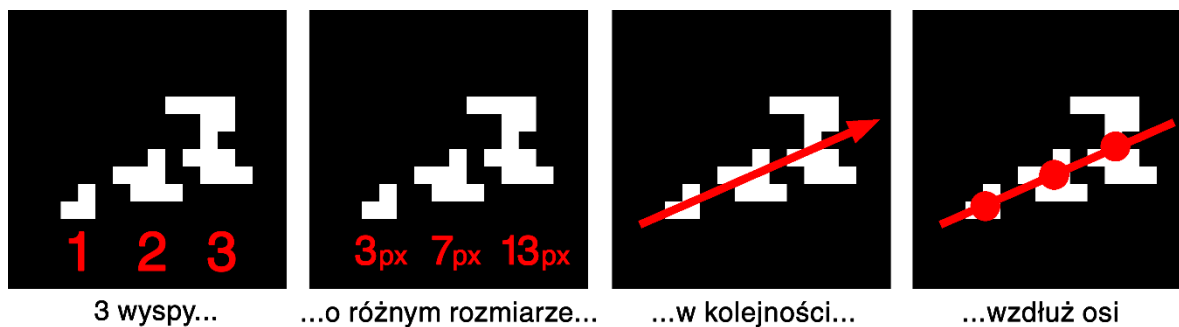
Ze względu na uprzedzenie dotyczące struktury danych (ang. spatial inductive bias) dające spłotowym sieciom neuronowym przewagę nad sieciami jednokierunkowymi w przetwarzaniu danych przestrzennych (Mitchell, 2017) oraz w związku z wynikami testów wstępnych porównujących ze sobą sieci typu FNN i CNN (Rys. 43), zdecydowałem się na wykorzystanie w prototypie sieci typu CNN.



Rys. 43. Wstępne testy skuteczności (%) FNN oraz CNN w przetwarzaniu wielowymiarowych zadań przestrzennych (po lewej) oraz przykłady rozwiązań wygenerowanych przez oba typy sieci (po prawej). Oba typy sieci miały za zadanie wygenerowanie trzech wysp spełniających serię zasad kompozycyjnych reprezentowanych w zbiorze treningowym. CNN osiąga skuteczność o 4 - 14% lepszą niż FNN o porównywalnej liczbie parametrów (Dzieduszyński, 2022b).

Wybrana sieć neuronowa miała za zadanie nauczenie się złożonej zasady kompozycyjnej na podstawie zadanych przykładów treningowych. Następnie sieć była testowana pod względem jej umiejętności do odczytywania nowych, niespotkanych w trakcie treningu kontekstów kompozycyjnych i do projektowania nowych kompozycji wpisujących się w konteksty i respektujących wytrenowaną, złożoną zasadę kompozycyjną. Wykorzystana w badaniu złożona zasada kompozycyjna składała się z następujących, trzech prostych zasad składowych:

- Na monochromatycznym obrazie znajdują się trzy białe wyspy na czarnym tle.
- Wyspy różnią się między sobą rozmiarem.
- Wyspy ułożone są osiowo w kolejności od najmniejszej do największej (Rys. 44) (Dzieduszyński, 2022a).

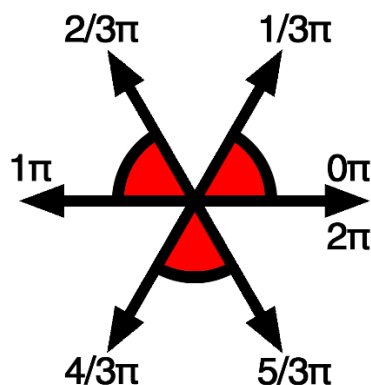


Rys. 44. Proste zasady kompozycyjne składające się na złożoną zasadę kompozycyjną wykorzystaną w eksperymencie wstępnym (Dzieduszyński, 2022a).

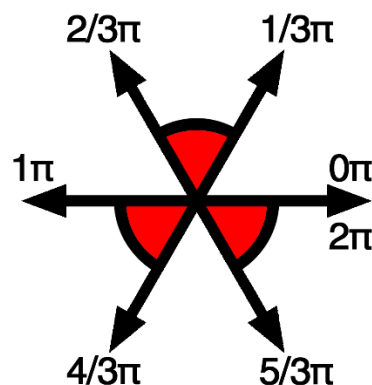
Spełnienie każdego z tych warunków można jednoznacznie ocenić przy pomocy parametrycznych algorytmów ilościowych. Algorytmy ilościowe testujące poszczególne zasady kompozycyjne zostały przeze mnie opracowane w środowisku Python. Przy wystarczającej liczbie przykładów testowych skuteczność sieci neuronowej można ocenić statystycznie względem każdego ze zdefiniowanych warunków kompozycyjnych. W celu wytrenowania i przetestowania sieci neuronowej stworzyłem bazę danych podzielonych na cztery podzbiory:

- Zbiór treningowy – złożony z obrazów reprezentujących badaną, złożoną zasadę kompozycyjną. Zbiór ten służył sieci do ustawienia parametrów minimalizujących funkcję straty.
- Zbiór walidacyjny – służący do zaprojektowania hiperparametrów sieci osiągającej najlepsze wyniki, przy jednoczesnym ograniczeniu ryzyka przestrojenia hiperparametrów do zbioru testowego (ang. over-tuning), czyli sytuacji, w której dobrane hiperparametry sieci zapewniają szczególnie dobry wynik dla konkretnego zbioru danych.
- Zbiór testowy – wykorzystywany do finalnej oceny skuteczności sieci neuronowej osiągającej najlepsze wyniki na zbiorze walidacyjnym. Zbiór testowy składał się z innych przykładów niż zbiór treningowy i walidacyjny. Przykłady były także próbkowane z innej dystrybucji niż przykłady treningowe (Rys. 45, Rys. 46). Dzięki temu rozwiązaniu nie było możliwości, by obrazy ze zbiorów testowych pokrywały się z obrazami ze zbiorów treningowych.
- W celach diagnostycznych stworzyłem także zbiór testowy próbkowany z dystrybucji treningowej, służący do rozpoznania, czy wprowadzone różnice w dystrybucji

nie wpływają zbyt poważnie na osiągnięte przez sieć wyniki (ang. covariate shift) (Dzieduszyński, 2022a).



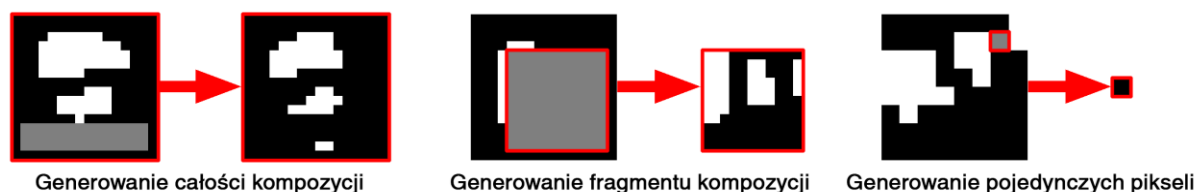
Rys. 45. Dozwolone osie kompozycyjne dla zbioru treningowego i zbioru testowego o dystrybucji treningowej (Dzieduszyński, 2022a).



Rys. 46. Dozwolone osie kompozycyjne dla zbioru walidacyjnego i zbioru testowego (Dzieduszyński, 2022a).

Sieci typu CNN zostały poddane próbom w trzech zadaniach o zróżnicowanym poziomie trudności (Rys. 47). Każdy scenariusz operował na osobnej bazie danych (ze względu na różnice w rozmiarach obrazów i trybów zakrywania kompozycji):

- I. Generowanie całości kompozycji (zakrytych fragmentów o różnych rozmiarach wraz z odtworzeniem zadanych kontekstów) na obrazach wielkości 16 x 16 pikseli.
- II. Generowanie zakrytych fragmentów o wielkości 11 x 11 pikseli uzupełniających kompozycję wielkości 16 x 16 pikseli.
- III. Generowanie pojedynczych, zakrytych pikseli kompozycji wielkości 8 x 8 pikseli (Dzieduszyński, 2022a).

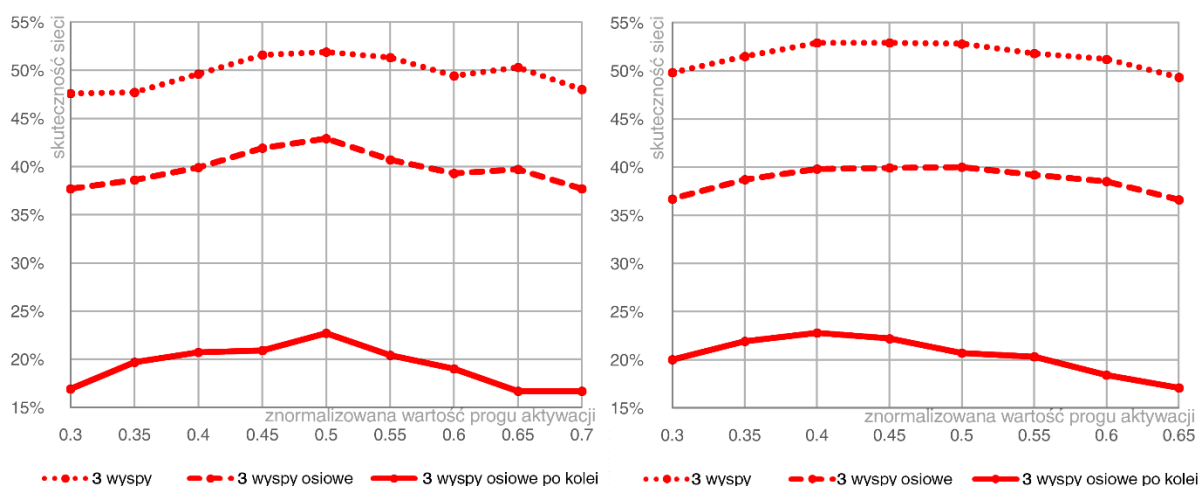


Rys. 47. Trzy scenariusze testowane w prototypie wstępnym polegające na generowaniu całości kompozycji, jej fragmentu oraz pojedynczego piksela (Dzieduszyński, 2022a).

Zastosowane sieci neuronowe zostały zaprojektowane hiperparametrycznie w środowisku Tensorflow 2. Ich struktura była luźno inspirowana siecią AlexNet (Krizhevsky, i in., 2012). Liczba neuronów, warstw i pozostałe hiperparametry zostały dopasowane do scenariusza, formatu danych wejściowych i wyjściowych oraz zoptymalizowane względem maksymalnej skuteczności na zbiorze walidacyjnym metodą poszukiwania losowego (ang. random search). Wartość progu aktywacji pikseli wyjściowych została ustalona metodą poszukiwania sieciowego (ang. grid search) dla zakresu 0,3 - 0,65 (Rys. 48). Definiowane hiperparametry obejmowały:

- Liczbę neuronów w sieci i w poszczególnych warstwach.
- Liczba i rodzaj warstw (warstwy splotowe i warstwy w pełni połączone).
- Techniki regularyzacyjne (dropout w zakresie 0 - 10% neuronów w warstwach w pełni połączonych oraz early stopping).
- Funkcje aktywacji neuronów w poszczególnych warstwach (ReLU i sigmoid).
- Funkcję straty (strata Hubera w scenariuszu I i II, binarna entropia krzyżowa (ang. binary cross-entropy) w scenariuszu III).
- Algorytm optymalizujący (Adam).
- Szybkość uczenia się (0.001).
- Wielkość partii treningowych (ang. batch size, 256 - 4096).
- Czas treningu.
- Wartość progu aktywacji pikseli wyjściowych [0,4 oraz 0,5 (Dzieduszyński, 2022a)].

Na wejściu sieci neuronowe otrzymywały monochromatyczny obraz z zakrytym fragmentem kompozycji i musiały przewidzieć uzupełnienie zakrytego fragmentu pasujące do zasad kompozycyjnych reprezentowanych przez przykłady w zbiorze treningowym. Kompozycje wykorzystane do stworzenia zbiorów treningowych, walidacyjnych i testowych zostały zsyntetyzowane przy pomocy osobnego, autorskiego narzędzia parametrycznego zaprogramowanego w języku Python i niewykorzystującego uczenia maszynowego. Zbiory treningowe składały się z 7000 zróżnicowanych obrazów, a zbiory testowe, walidacyjne i walidacyjne o dystrybucji treningowej – z 700 każdy. Czas treningu liczony w epokach różnił się w zależności od przyjętych wielkości partii treningowych w poszczególnych epokach, ale każdą z sieci trenowano przez porównywalny czas, wynoszący około dwa dni na karcie graficznej Nvidia GTX960M.



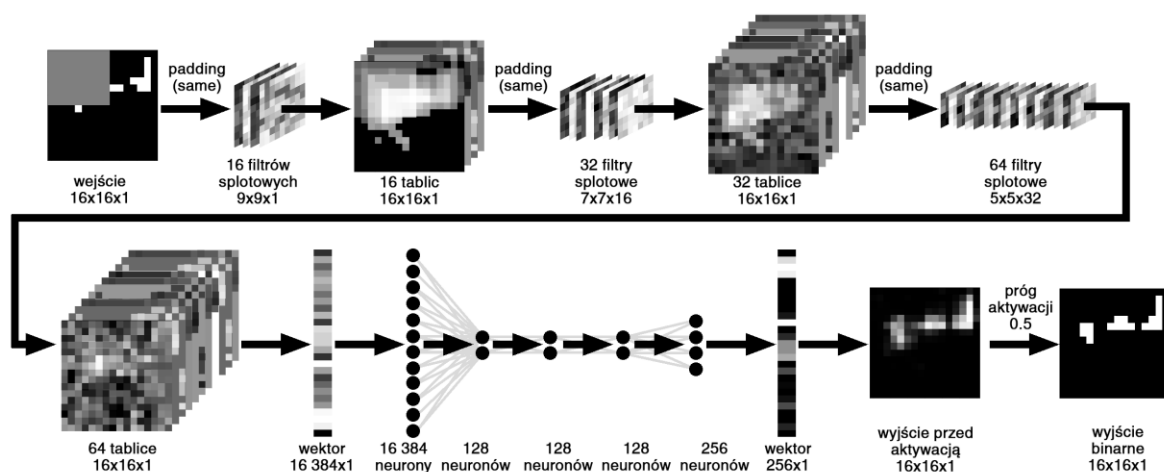
Rys. 48. Przykład przeprowadzonej optymalizacji hiperparametru znormalizowanej wartości progu aktywacji pikseli wyjściowych. Optymalizacja przeprowadzona została metodą grid search w zakresie od 0,35 do 0,65 na zbiorach walidacyjnych dla scenariusza I i II względem skuteczności sieci na badanych zasadach kompozycyjnych. Optymalna wartość wyniosła 0,5 dla scenariusza I (po lewej) i 0,4 dla scenariusza II (po prawej) (Dzieduszyński, 2022a).

3.1.3. Wyniki eksperymentu wstępnego

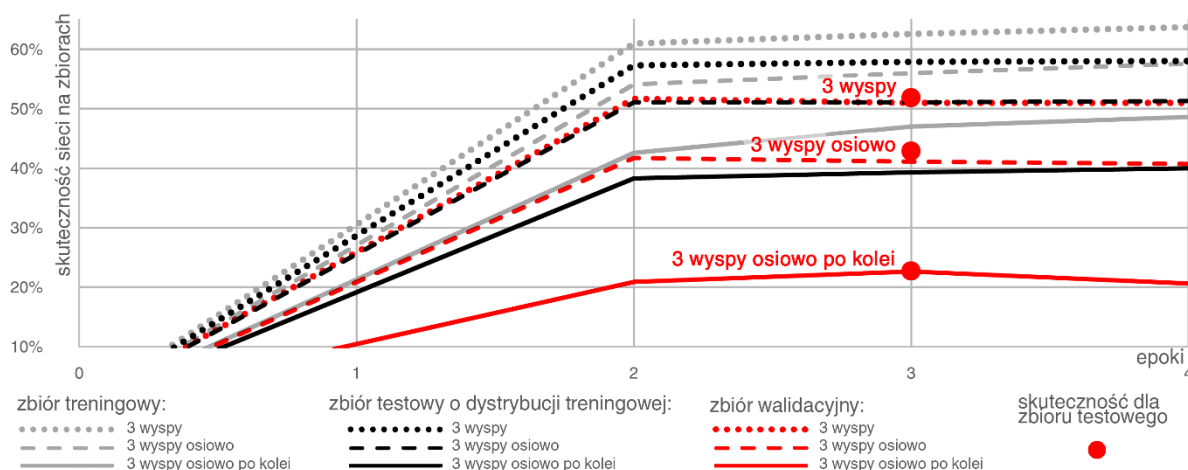
Generowanie całości kompozycji, scenariusz I

W najtrudniejszym scenariuszu pierwszym, w którym sieć neuronowa musiała radzić sobie ze zmiennym rozmiarem i proporcjami okna zakrywającego kompozycję i musiała jednocześnie odtworzyć konteksty kompozycyjne, sieć poradziła sobie najslabiej [Rys. 49 (Dzieduszyński, 2022a)]. W tym przypadku na zbiorze testowym sieć osiągnęła wynik 51,9% dla jednej zasady kompozycyjnej, 42,9% dla dwóch zasad kompozycyjnych jednocześnie i 22,7% dla wszystkich trzech zasad [Rys. 50 (Dzieduszyński, 2022a)]. Sieć w miarę dobrze radziła sobie z przypadkami, w których w zadanych kontekstach było widać załączek wszystkich trzech wysp i w których zakryty obszar nie był duży. Najczęstsze błędy wynikały ze zbyt małej pewności przewidywań sieci skutkującej zbyt słabymi aktywacjami niegenerującymi pikseli, mimo widocznej w aktywacjach struktury trzech wysp. W innych przypadkach sieć nie radziła sobie z samym zadaniem odtworzenia niezaskłoniętych fragmentów kompozycji (co nie było zmienną badaną w eksperymencie), przez co obraz wyjściowy nie pasował do obrazu

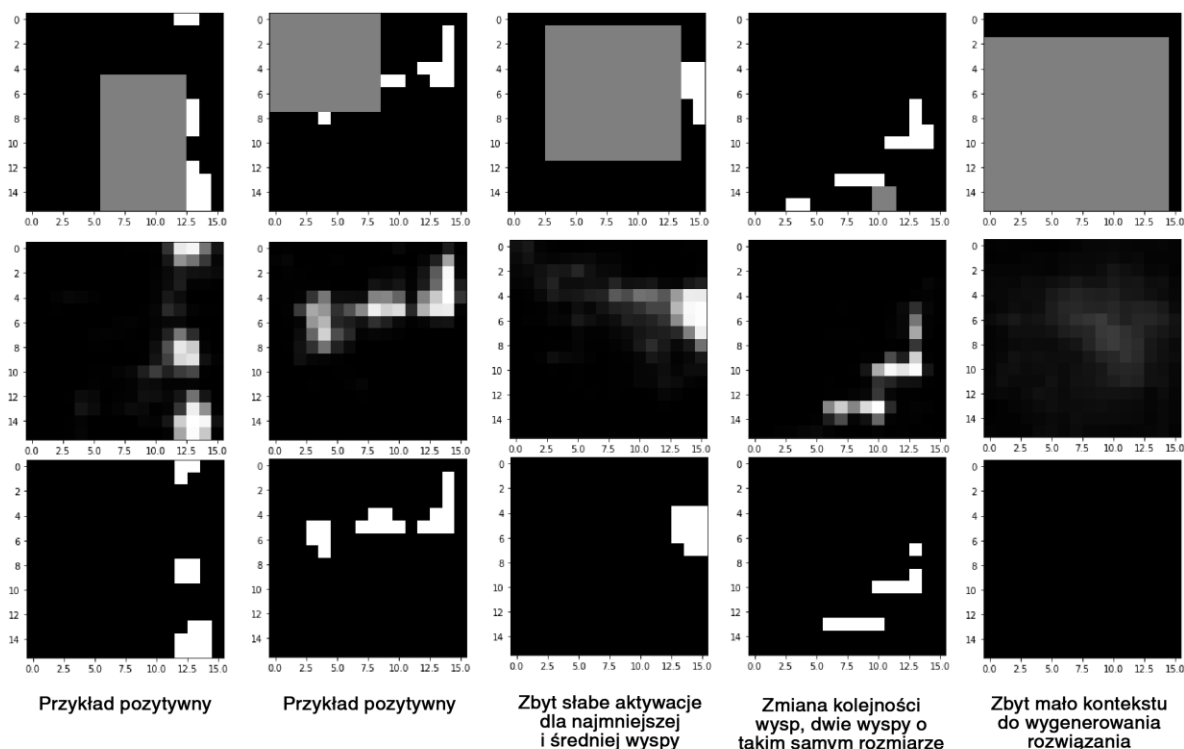
wejściowego. Sieć nie potrafiła także generować prawidłowych kompozycji, gdy zadane konteksty były zbyt ubogie (Rys. 51).



Rys. 49. Diagram sieci neuronowej wykorzystanej w scenariuszu I (Dzieduszyński, 2022a).



Rys. 50. Wykres przedstawiający ewolucję skuteczności sieci w scenariuszu I w miarę treningu. By zapobiec przetrenowaniu sieci, trening został zatrzymany w epoce trzeciej (technika regularyzacyjna early stopping) (Dzieduszyński, 2022a).

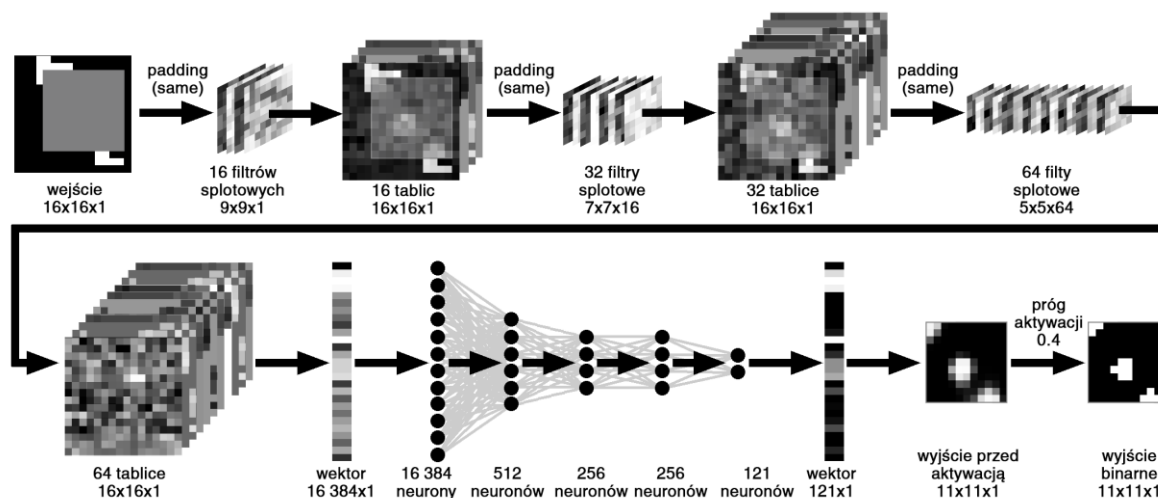


Rys. 51. Wybór przykładów pozytywnych rozwiązanych prawidłowo i negatywnych, w których sieć miała problemy (Dzieduszyński, 2022a).

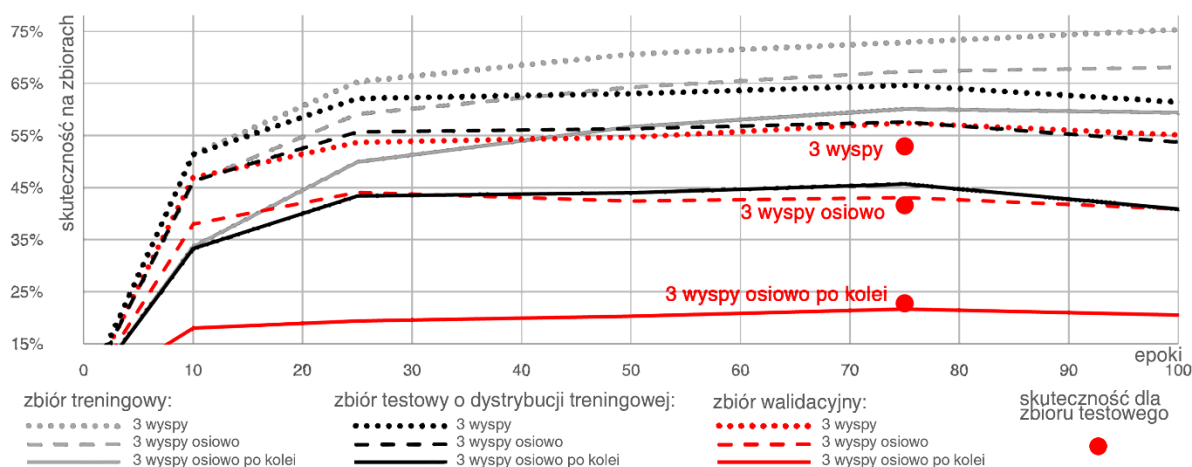
Generowanie fragmentu kompozycji, scenariusz II

W drugim scenariuszu sieć nie musiała już uczyć się odtwarzania całości kompozycji włącznie z jej odkrytymi fragmentami. Sieć musiała jedynie generować zakryty fragment kompozycji o wymiarach 11 x 11 pikseli (Rys. 52). Stały rozmiar generowanego fragmentu kompozycji wynikał z faktu wykorzystania nie w pełni splotowej sieci neuronowej (Long, i in., 2015) i konieczności dopasowania wymiarowości wyjścia sieci neuronowej do stałej struktury sieci (ang. dimensionality mismatch). Uniknięcie błędów wynikających z nieprawidłowego odtworzenia kompozycji uczyniło otrzymane wyniki. Co więcej, nawet pomimo statystycznego zwiększenia obszaru generowanej kompozycji względem scenariusza pierwszego (121 pikseli w scenariuszu II i średnio 81 pikseli w scenariuszu I), nieznacznie poprawiła się skuteczność sieci. Sieć na zbiorze testowym osiągnęła wynik 52,9% dla pojedynczej zasady kompozycyjnej, 41,6% dla podwójnej zasady i 22,8% dla pełnej zasady potrójnej [Rys. 53 (Dzieduszyński, 2022a)]. W większości przypadków zasada liczby składowych elementów kompozycji była zachowana. W przypadkach, w których zadany

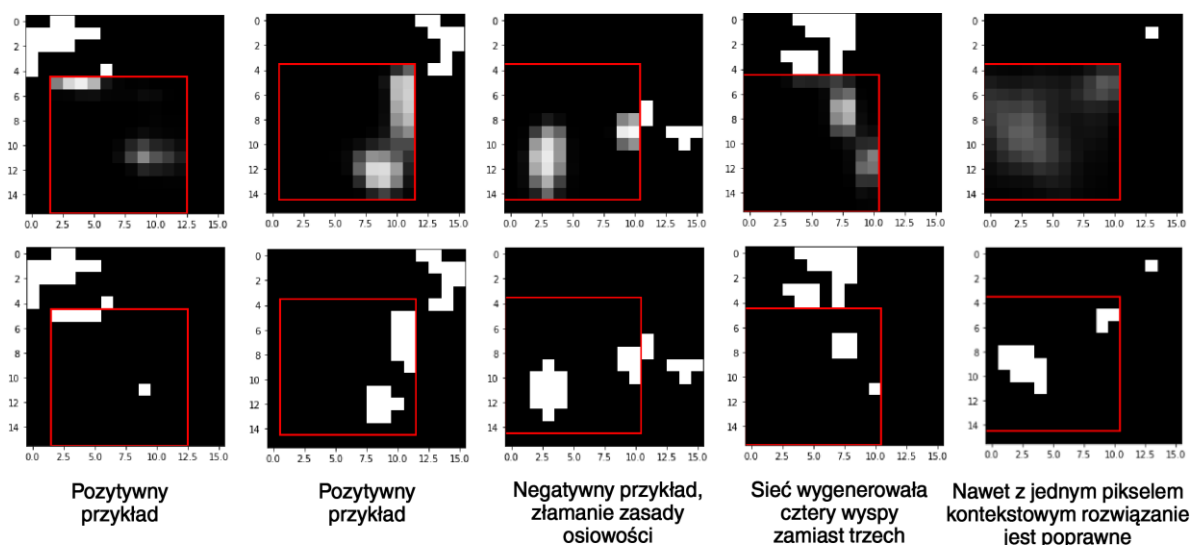
kontekst okazał się w scenariuszu I zbyt ubogi, sieć scenariusza II radziła sobie o wiele lepiej (Rys. 54).



Rys. 52. Diagram sieci neuronowej wykorzystanej w scenariuszu II (Dzieduszyński, 2022a).



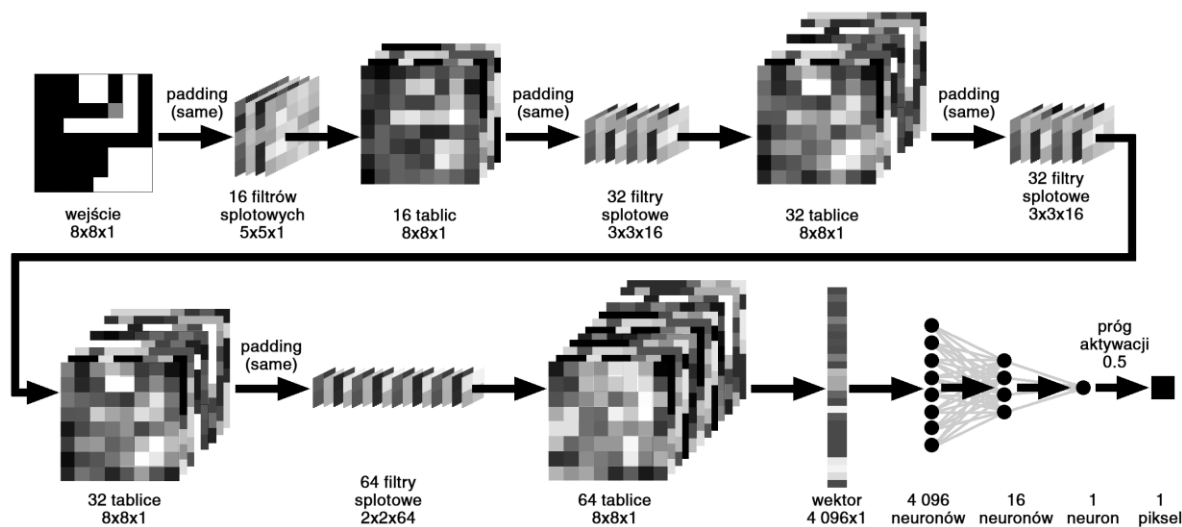
Rys. 53. Wykres przedstawiający ewolucję skuteczności sieci w scenariuszu II w miarę treningu. By zapobiec przetrenowaniu sieci, trening został zatrzymany w epoce 75 (Dzieduszyński, 2022a).



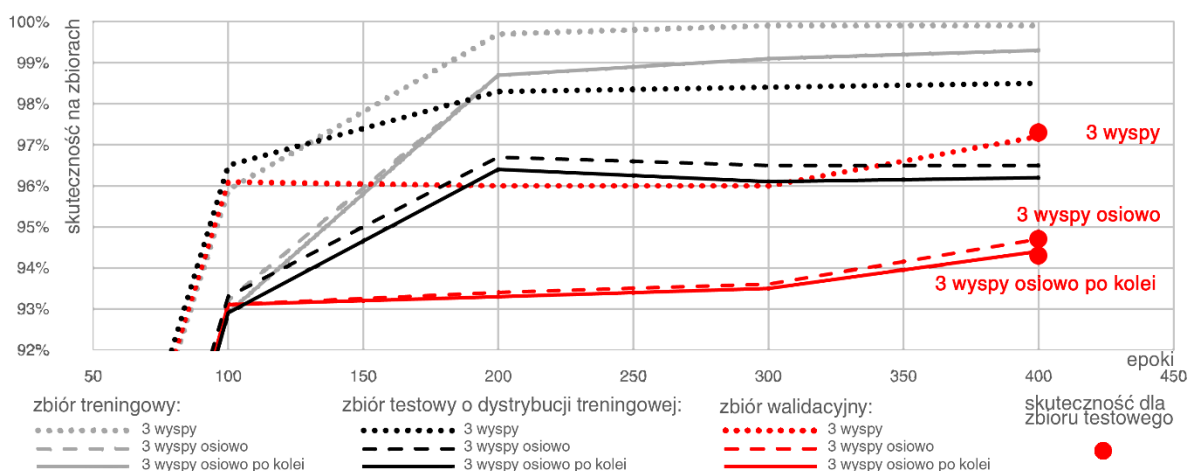
Rys. 54. Wybór przykładów pozytywnych, z którymi sieć scenariusza II radziła sobie bardzo dobrze i negatywnych, w których sieć miała problemy (Dzieduszyński, 2022a).

Generowanie pojedynczych pikseli, scenariusz III

W trzecim scenariuszu przetestowałem skuteczność sieci w podejmowaniu najbardziej podstawowych decyzji kompozycyjnych, polegających na generowaniu pojedynczych brakujących pikseli w kompozycji przestrzennej (Rys. 55). Dzięki zmniejszeniu rozmiaru przetwarzanych obrazów do kwadratów wielkości 8 x 8 pikseli, już pojedynczy, źle wygenerowany piksel, mógł złamać zasadę osiowości. Jeden piksel mógł w wielu przypadkach łamać także zasadę liczby elementów (poprzez wygenerowanie nowej, jednopikselowej wyspy lub poprzez połączenie istniejących wysp ze sobą) oraz zasadę kolejności wysp (jeden piksel mógł doprowadzić do zrównania wielkości wysp).



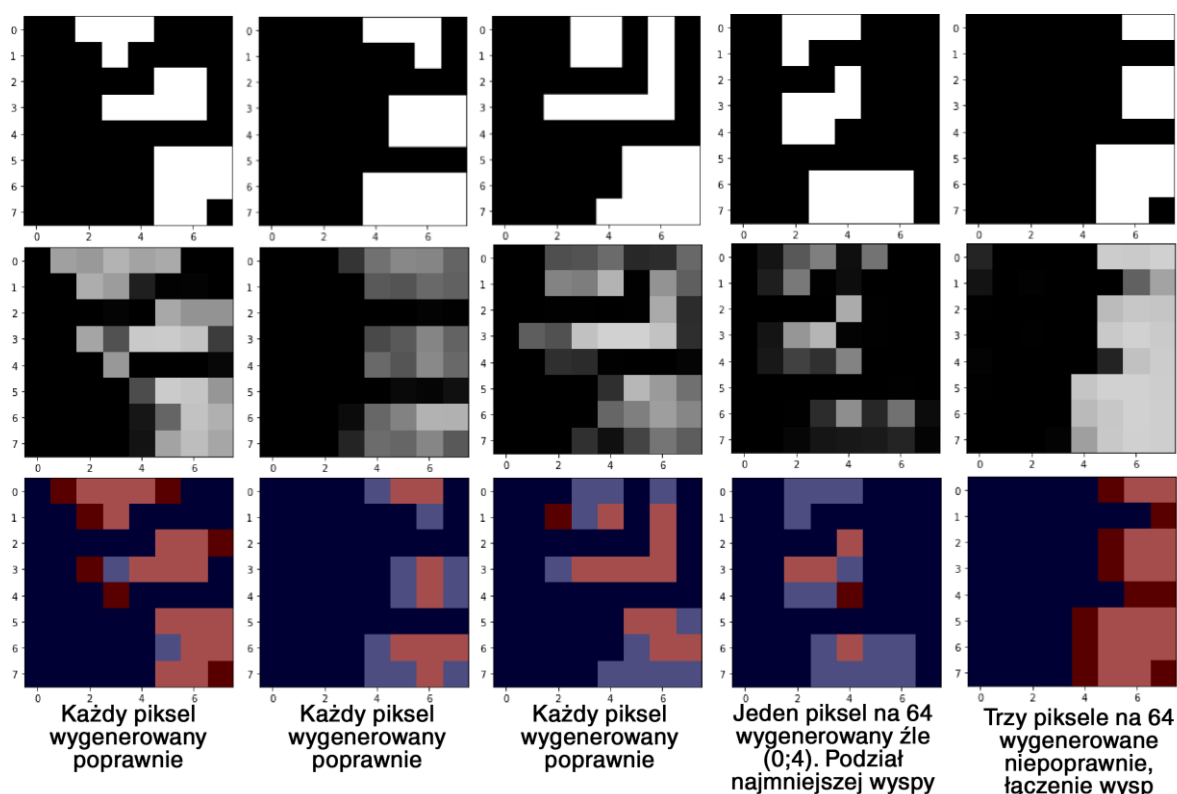
Rys. 55. Diagram sieci neuronowej wykorzystanej w scenariuszu III (Dzieduszyński, 2022a).



Rys. 56. Wykres przedstawiający ewolucję skuteczności sieci w scenariuszu III w miarę treningu. W tym przypadku doprowadzono trening do końca, early stopping nie był konieczny (Dzieduszyński, 2022a).

W scenariuszu trzecim sieć radziła sobie prawie perfekcyjnie w przetwarzaniu całości złożonej zasady kompozycyjnej (Rys. 56). W zbiorze testowym kompozycja prawidłowo składała się z trzech wysp w 97,9%, z trzech wysp ułożonych osiowo w 96,7%, a z trzech wysp ułożonych osiowo i uszeregowanych od największej do najmniejszej w 95,1% (Dzieduszyński, 2022a). W prawie każdym przypadku sieć prawidłowo uzupełniała brakujący piksel kompozycji. Analiza wzorców aktywacji pozwala zauważyć, że sieć bardzo konsekwentnie unika generowania pikseli w miejscach, w których wygenerowanie piksela skutkowałoby

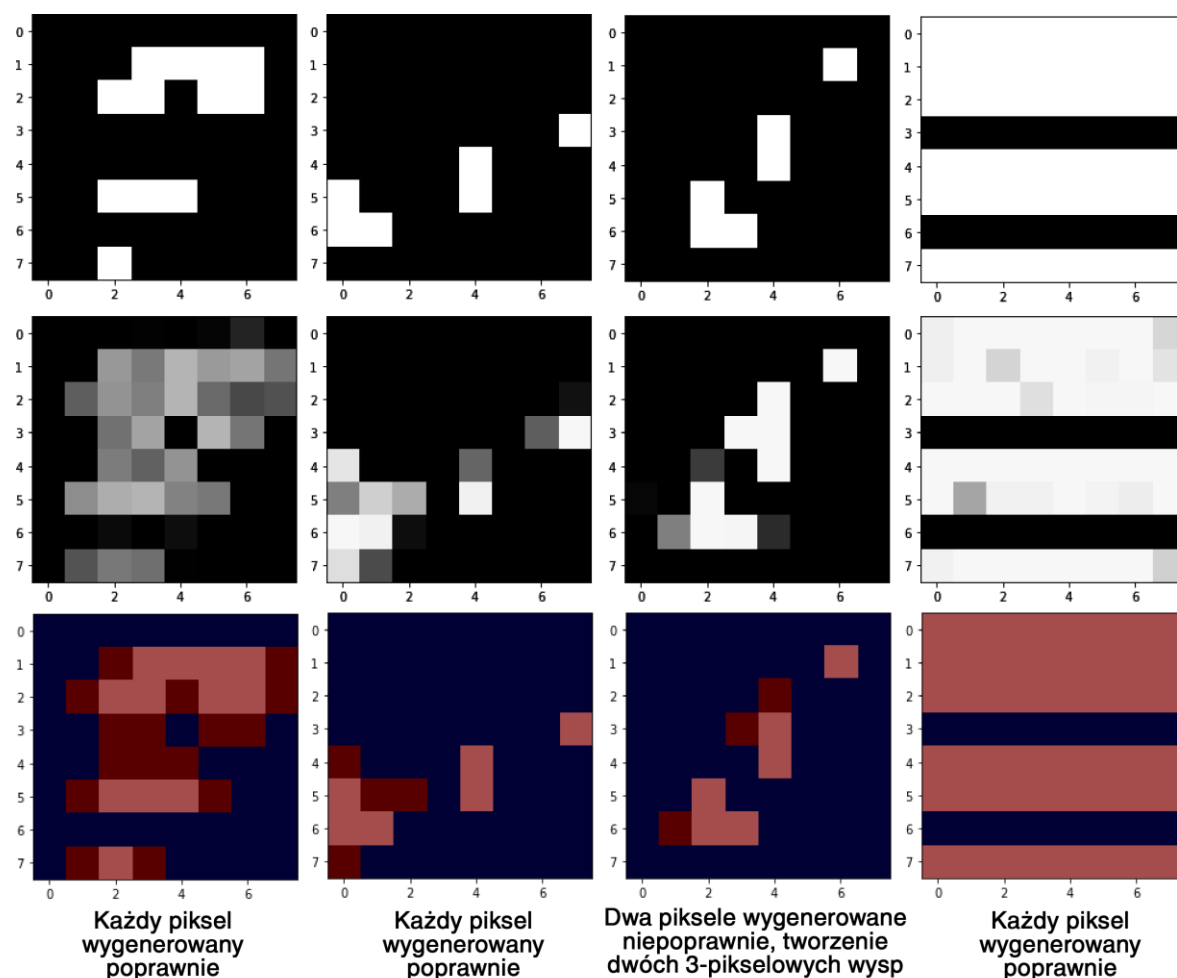
zespoleńiem dwóch wysp ze sobą. Sieć także bardzo silnie unikała generowania nie-osioowych pikseli w pustce otaczającej wyspy. Ciągłość cienkich wysp o szerokości jednego piksela też była zachowywana w większości przypadków. Kilka błędów popełnionych przez sieć dotyczyło okazjonalnego łączenia wysp ze sobą i przełamywania ciągłości cienkich wysp (Rys. 57).



Rys. 57. Diagram przedstawiający wyniki działania sieci w scenariuszu. Każdy z pikseli w kompozycji był sekwencyjnie zakrywany i generowany przez sieć osobno. W górnym rzędzie – kompozycje wejściowe, w środkowym – mapa aktywacji dla każdego z 64 pikseli, w dolnym – decyzje podjęte przez sieć dla każdego piksela (czerwony kolor oznacza decyzję o narysowaniu białego piksela, niebieski kolor oznacza decyzję o wygenerowaniu czarnego piksela). Wybór przykładów pozytywnych i negatywnych (Dzieduszyński, 2022a).

W celu dodatkowego przetestowania sieci, ręcznie narysowałem kilka nietypowych kompozycji przestrzennych niewpisujących się w dystrybucję zbioru treningowego, ale spełniających wszystkie warunki kompozycyjne (Rys. 58). Przypadki te testowały występowanie wysp o wielkości pojedynczego piksela oraz występowanie wysp różniących się rozmiarem jedynie o jeden piksel. Ostatnim przypadkiem była kompozycja złożona z wysp

o maksymalnej możliwej wielkości (podzielonych jedynie cienkimi pasmami o szerokości jednego piksela). Również na tych przykładach brzegowych sieć osiągnęła bardzo wysoką skuteczność wynoszącą 98,8%.



Rys. 58. Wyniki sieci dla ręcznie rysowanych przykładów brzegowych. Każdy z pikseli w kompozycji był sekwencyjnie zakrywany i generowany przez sieć osobno. W górnym rzędzie – kompozycje wejściowe, w środkowym – mapa aktywacji dla każdego z 64 pikseli, w dolnym – decyzje podjęte przez sieć dla każdego piksela (czerwony kolor oznacza decyzję o narysowaniu białego piksela, niebieski kolor oznacza decyzję o wygenerowaniu czarnego piksela) (Dzieduszyński, 2022a).

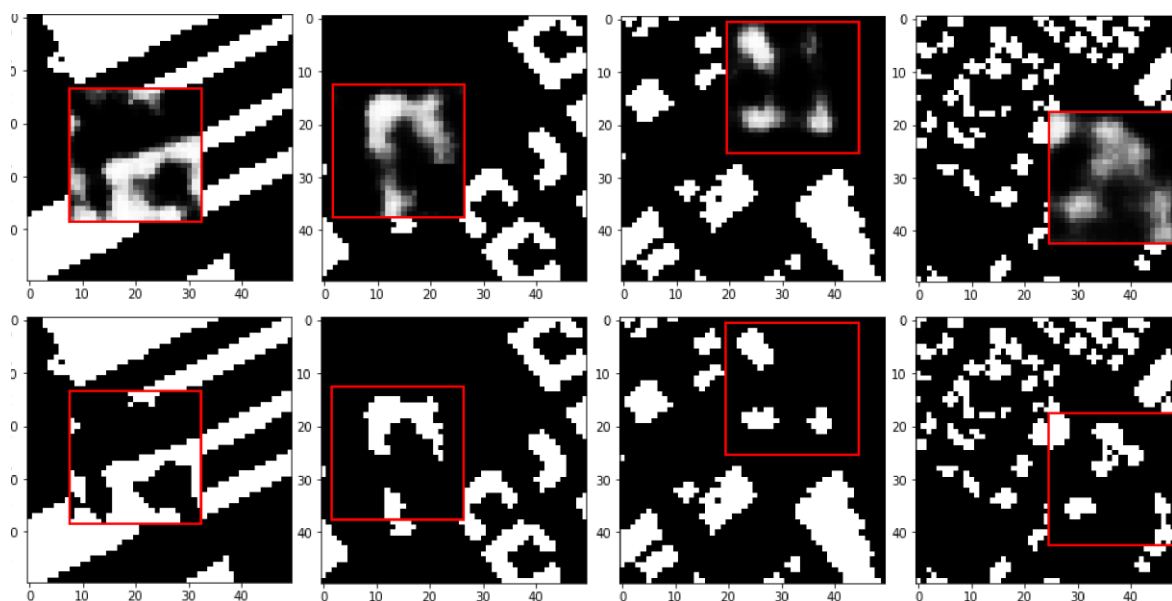
3.1.4. Wnioski z eksperymentu wstępnego

Prototyp wstępny został oceniony zarówno jakościowo, jak i ilościowo. Opracowany system wykazał bardzo wysoką skuteczność sięgającą 95,1 - 97,9% w przetwarzaniu złożonych zasad

kompozycyjnych przy podejmowaniu prostych decyzji przestrzennych w scenariuszu III. Im trudniejsze stawało się zadanie, tym słabsze wyniki były osiągane przez sieć neuronową, jednak nadal zawierały się we względnie wysokim zakresie 22,7 - 52,9% w zależności od złożoności mierzonych zasad. Bardzo wysoka wydajność w scenariuszu trzecim sugeruje, że sieć na podstawie zbioru treningowego potrafi prawidłowo nauczyć się zadanych zasad kompozycyjnych, jednak w scenariuszach I i II otrzymany wynik obniżyło dodanie zadania na etapie przetwarzania i aplikacji zasad kompozycyjnych, polegające na konieczności odtwarzania niezakrytych fragmentów kompozycji lub przetwarzania większej liczby sygnałów. Być może dodatkowym czynnikiem ograniczającym skuteczność w scenariuszach I i II był stosunkowo niewielki rozmiar wykorzystanej sieci neuronowej i zbioru treningowego. 7000 przykładów treningowych to w kontekście głębokiego uczenia maszynowego bardzo mało i taka objętość zbioru treningowego prowadzi do bardzo szybkiego przetrenowania sieci. Innym problemem zidentyfikowanym w badaniu były zbyt agresywne różnice wprowadzone do dystrybucji między zbiorami treningowym i walidacyjnym, a zbiorem testowym. Różnice te miały docelowo zademonstrować zdolność sieci do generalizowania na niewidziane w treningu rodzaje kompozycji, jednak doprowadziły do znacznego obniżenia wyników na zbiorze testowym. Możliwe, że sieć „uznała” różnice w dystrybucji za pewne dodatkowe zasady kompozycyjne. Problem ten w dziedzinie głębokiego uczenia maszynowego nazywa się problemem przesunięcia dystrybucji (ang. covariate shift). Wydajność sieci mierzona na zestawie walidacyjnym (składającym się również z przykładów niewidzianych przez sieć na etapie treningu, jednak trzymających się tego samego zakresu możliwych osi kompozycyjnych), osiągnęła w scenariuszu I 40 - 58%, w scenariuszu II 46 - 65%, a w scenariuszu III 96 - 98,5%.

Ze względu na złożoność rzeczywistego procesu projektowania architektoniczno-urbanistycznego wydaje się, że bezpośrednie zastosowanie narzędzia na podobnym poziomie zaawansowania co opracowany prototyp wstępny mogłoby przynieść ograniczone rezultaty. By sprawdzić jednak, do jakiego stopnia podobne narzędzie radziłoby sobie w rozpoznawaniu prawdziwych kontekstów przestrzennych, wytrenowałem nieznacznie powiększoną sieć ze scenariusza II na diagramatycznych schematach zabudowy warszawskiego Śródmieścia (tzw. szwarcplany). Następnie aplikowałem tak wytrenowaną sieć neuronową do uzupełniania brakującej zabudowy na terenie bardziej peryferyjnej dzielnicy Białołęka. Sytuacja ta w założeniu miała symulować „przeszczepienie” zasad kompozycyjnych

z warszawskiego śródmieścia na Białąkę przy jednoczesnym uzupełnianiu i respektowaniu zastanych na Białące kontekstów (Rys. 59). Co ciekawe, nawet niewielka sieć neuronowa, wytrenowana na bardzo ograniczonych, dwuwymiarowych kontekstach przestrzennych potrafiła odczytywać i wykorzystywać pewne urbanistyczne zasady kompozycyjne. Sieć wydawała się respektować zasady takie jak gęstość zabudowy, osiowość zabudowy, ogólną skalę obrysów budynków i konieczność kontynuacji pierzei.



Rys. 59. Przykłady białoleckich struktur urbanistycznych zaproponowanych przez sieć neuronową wytrenowaną na szwaroplanach zabudowy Śródmieścia Warszawy. Zabudowa proponowana przez sieć znajduje się w czerwonych ramkach. Nawet przy niewielkiej rozdzielczości wynikającej z ograniczeń strukturalnych sieci i bardzo ograniczonych kontekstach reprezentowanych w przetwarzanych danych, sieć wydaje się respektować gęstość i osiowość zabudowy, skalę obrysów budynków i konieczność kontynuacji pierzei (Dzieduszyński, 2022a).

3.2. Eksperyment główny

3.2.1. Cele szczegółowe eksperymentu głównego

Po zademonstrowaniu zdolności algorytmów głębokiego uczenia maszynowego do przetwarzania abstrakcyjnych kontekstów przestrzennych, kolejnym krokiem była próba aplikacji bardziej zaawansowanych sieci neuronowych do analizowania realnych

i obszerniejszych kontekstów przestrzennych w rzeczywistych miastach. Głównym celem eksperymentu głównego było stworzenie prototypowego ekstraktora punktów głównych i punktów formalnie podkreślonych w kompozycjach urbanistycznych. Projektowane narzędzie miało umożliwiać seryjne i powtarzalne przeprowadzanie czasochłonnych analiz kompozycyjnych. Ekstraktor taki stanowiłby wsparcie w praktyce projektowej architektów i urbanistów poprzez zmniejszenie ryzyka przeoczenia istotnych kontekstów i popełnienia błędu projektowego. W praktyce planistycznej i administracyjnej ekstrakcja punktów głównych potencjalnie mogłaby służyć do uczytelniania pozornie chaotycznych układów przestrzennych poprzez wskazywanie miejsc, którym powinna być poświęcona szczególna uwaga (np. ze względu na konieczność ich ochrony czy ze względu na ich potencjał rozwojowy i krystalizacyjny). Prototypowany algorytm mógłby być także wykorzystywany przez wszelkie instytucje zainteresowane ekonomicznym znaczeniem punktów głównych.

Przetwarzanie kontekstów kompozycyjnych jest jednym z ważniejszych elementów procesu projektowego. Eksperyment główny stanowił szansę na zademonstrowanie w praktyce koncepcji projektowania hiperparametrycznego, będącego ewolucją paradygmatu projektowania parametrycznego. Demonstracja wymagała wykorzystania algorytmów uczenia maszynowego do przetwarzania złożonych, realnych kompozycji przestrzennych w rzeczywistych miastach. Proces przygotowania prototypu można było oprzeć na głównych założeniach architektury hiperparametrycznej:

- Podejście do problemu projektowego na wyższym poziomie abstrakcji, niż w projektowaniu parametrycznym poprzez definiowanie hiperparametrów sieci neuronowej.
- Wspólne odczytywanie kontekstów projektowych, zarówno przez projektanta, jak i algorytm głębokiego uczenia maszynowego.
- Uzupełnienie narzędzi opartych na dedukcji także o narzędzia wykorzystujące rozumowanie indukcyjne.
- Automatyzacja procesów, które do tej pory mogły być przeprowadzane jedynie przez człowieka.

Celem była także jakościowa i ilościowa ocena skuteczności opracowanego prototypu i sprawdzenie, czy zwiększenie złożoności problemu względem eksperymentu wstępnego doprowadzi do osiągnięcia granic możliwości stosowanych sieci neuronowych.

3.2.2. Metodyka eksperymentu głównego

Kodowanie kontekstów przestrzennych – mapa cech

Projektowane narzędzie w założeniu odczytywać ma geometryczne uwarunkowania przestrzenne i na ich podstawie ustalać lokalizację ewentualnych głównych punktów kompozycyjnych. Uwarunkowania przestrzenne musiały być przedstawione w sposób spełniający następujące założenia:

- Dane powinny reprezentować kluczowe, geometryczne cechy przestrzenne opisujące w możliwie pełny sposób kompozycję przestrzenną w skali urbanistycznej.
- Dane powinny być aktualne i na bieżąco aktualizowane, by odzwierciedlać obecny stan przestrzeni.
- Poszczególne cechy powinny być jak najbardziej od siebie niezależne (jedna cecha nie powinna bezpośrednio wynikać z innej).
- Powinno być możliwe pobranie danych opisujących dowolne miejsce na Ziemi (lub możliwie największego zakresu możliwych miejsc) z zachowaniem zgodności danych między poszczególnymi miejscami.
- Dane nie powinny naruszać prywatności osób i innych interesariuszy.
- Dane powinny mieć wystarczająco dużą rozdzielczość umożliwiającą odczytywanie z nich kontekstów kompozycyjnych w skali urbanistycznej.
- Dane powinny mieć wystarczająco niską rozdzielczość umożliwiającą ich efektywne przetwarzanie maszynowe na dostępnym sprzęcie.
- Dane powinny być łatwo dostępne, darmowe i na otwartej licencji.
- Dane powinny być możliwe do pobrania przez Internet przy pomocy API.
- Dane powinny być przedstawione w formacie umożliwiającym ich przetworzenie przy pomocy dostępnych algorytmów uczenia maszynowego.

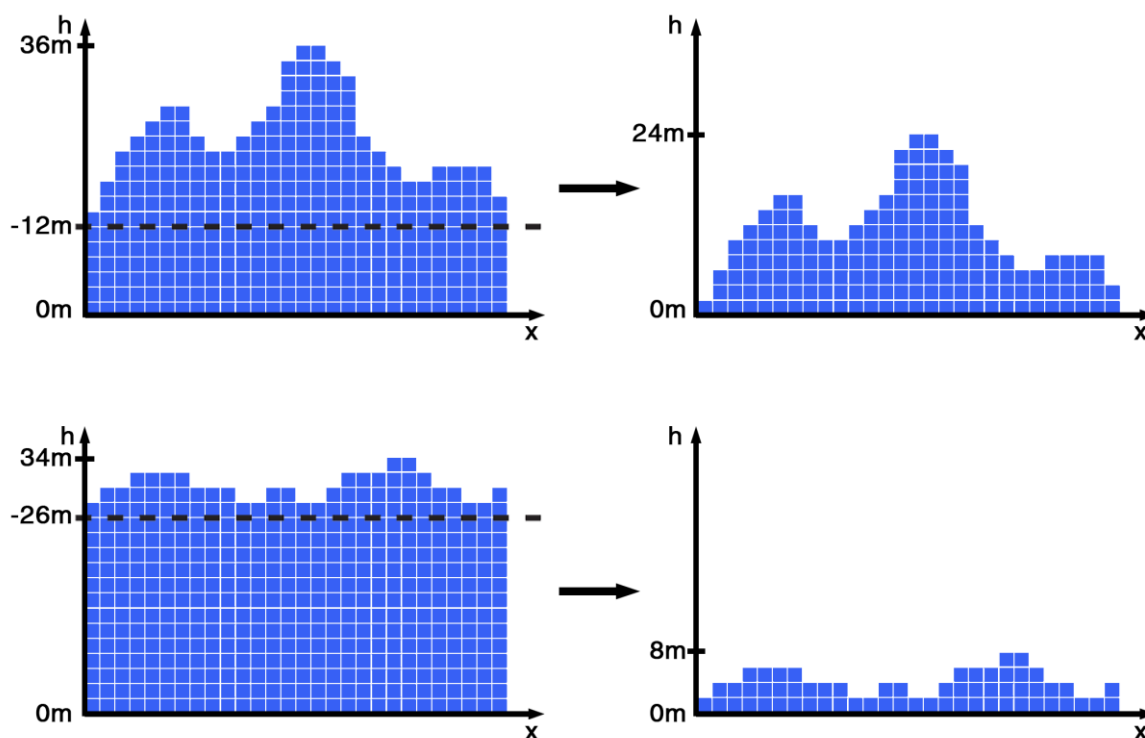
Do projektowania map cech wybrałem narzędzie Mapbox Studio (Mapbox, 2023), pozwalające na tworzenie własnych, rastrowych map opartych na Open Street Map (OpenStreetMap contributors, 2023). Szablony stworzone w Mapbox Studio można opublikować i następnie odczytywać przy pomocy zapytań API dla dowolnych współrzędnych geograficznych w dużym zakresie skal i rozdzielczości. Oś horyzontalna map rastrowych odpowiada długości geograficznej, a oś wertykalna szerokości geograficznej. W serwisie Mapbox Studio przygotowałem 3 osobne szablony reprezentujące trzy osobne przestrzenie cech uznane

za najbardziej kluczowe w kontekście urbanistycznej kompozycji przestrzennej. Wybór cech kluczowych został dokonany na podstawie wniosków płynących z omówionej literatury przedmiotu oraz na podstawie dostępności danych w serwisie OSM. Pierwsza mapa cech reprezentuje topografię wraz z zabudową, druga – zieleń, a trzecia – układ komunikacyjny.

Zabudowa i topografia terenu wspólnie stanowią najważniejszy kontekst budujący kompozycję przestrzenną. Zabudowa otaczająca obserwatora tworzy płaszczyzny naprowadzające i zatrzymujące wzrok, formuje rytmy, definiuje które elementy składają się na tło, a które wybijają się na pierwszy plan jako dominanty (Żórawski, 1962; Wejchert, 1984). Topografia terenu także formuje naprowadzenia widokowe, bariery przestrzenne, uwidacznia niektóre obiekty, a niektóre ukrywa (Ching, 2007; Wejchert, 1984). Elementy topograficzne, takie jak doliny rzek czy skarpy, mogą stanowić jedne z głównych elementów tożsamości miasta (Pluta, 2001). Przestrzeń cech topografii i zabudowy została zakodowana w postaci rastrowego, ośmiobitowego obrazu:

- 0: Tło.
- 0 - 127: Topografia terenu centrowana względem wartości minimalnej (Rys. 60), skok o jedną wartość w przestrzeni cech oznacza skok o 2 m wysokości rzeczywistej.
- 0 - 128: Zabudowa terenu, mosty, i wiadukty. Skok o jedną jednostkę odpowiada 2 m wysokości rzeczywistej.

Mapy topografii i zabudowy są sumowane do jednej mapy, w której wartości pikseli zawierają się w zakresie 0 - 255.



Rys. 60. Diagram przedstawiający centrowanie map topograficznych względem wartości minimalnej. Przy pobieraniu danych z serwisu Mapbox opracowany skrypt API rozpoznaje najmniejszą wartość piksela na obrazie i odejmuje tę wartość od wartości pozostałych pikseli. Dzięki temu mapa topograficzna przedstawia tylko wysokości względne i pozostawia więcej przestrzeni dla mapy zabudowy. Opracowanie własne.

Drugą mapę cech stanowi mapa intensywności zieleni. Jak pisze Jan Gehl:

„Drzewa, zieleń i kwiaty odgrywają kluczową rolę wśród elementów przestrzeni miejskiej. Drzewa (...) definiują przestrzeń miasta i pomagają zaakcentować ważne miejsca. Duże drzewo na placu informuje: «To jest miejsce», a szpalery na bulwarach podkreślają linearny porządek, zaś zwieszające się nad ulicami gałęzie sugerują istnienie zielonej przestrzeni w mieście.” (Gehl, 2014, s. 179-180)

Na rolę zieleni w budowaniu kompozycji przestrzennej miejsca zwraca także Katarzyna Pluta. Szpalery zieleni mogą tworzyć ściany wewnątrz urbanistycznych, a trawniki czy klomby na posadzce urbanistycznej mogą działać jak naprowadzenia (Pluta, 2014). Zieleń traktowana jest jako element służący budowaniu kompozycji urbanistycznej też przez dokumenty planistyczne, takie jak Studium Warszawy (Prezydent m.st. Warszawy, 2023). Zieleń

znajdującą się w polu percepcji zakodowałem względem jej intensywności opierając się na klasach występujących w OSM:

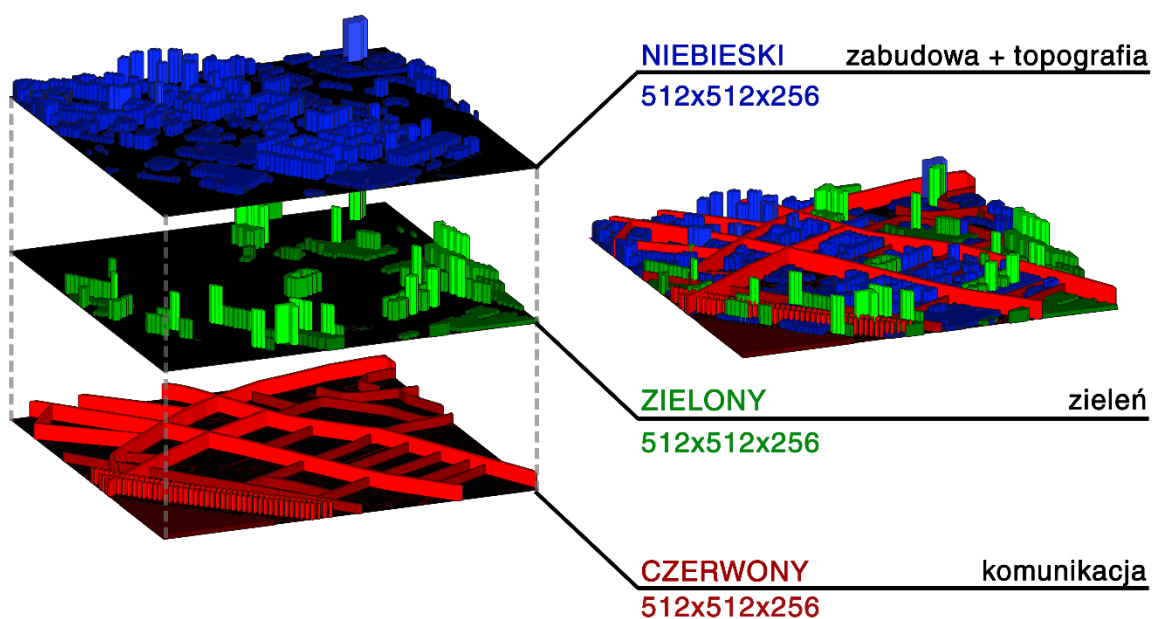
- 0: Tło.
- 51: Powierzchnie naturalne bez zieleni (tereny piaszczyste, skaliste, plaże).
- 102: Zieleń niska (tereny trawiaste i uprawne).
- 153: Zieleń średnio-niska (krzewy, zarośla, wrzosowiska, winnice).
- 204: Zieleń średnio-wysoka (rodzinne ogrody działkowe, parki).
- 255: Zieleń wysoka (lasy).

Trzecia mapa reprezentuje układ komunikacyjny. Układ komunikacyjny w założeniu ma pomóc sieci neuronowej powiązać geometryczną kompozycję tworzoną przez zabudowę, topografię i zielen z perspektywą użytkownika przestrzeni – zarówno pieszego, jak i zmotoryzowanego. W skali dużych miast układ komunikacyjny, ciągi, ulice wlotowe i bramy same w sobie stanowią także istotne elementy kompozycyjne (Pluta, 2001). Ulice i ścieżki determinują istnienie osi widokowych (Wejchert, 1984), przecięcie ciągów komunikacyjnych stanowi podkreślenie kompozycyjne (Ching, 2007; Alexander, i in., 1977) lub wręcz definiuje punkty miastotwórcze (Ratzel, 1882). Układ komunikacyjny podobnie jak zielen i zabudowa został przedstawiony na mapie w poprzez gradient intensywności:

- 0: Tło.
- 31: Woda (rzeki, jeziora, kanały).
- 63: Drogi w budowie.
- 95: Chodniki, ścieżki, przejścia.
- 127: Drogi bez klasyfikacji OSM.
- 159: Drogi klasy „street” i „road” OSM.
- 191: Drogi klasy „secondary” i „tertiary” OSM.
- 223: Drogi klasy „primary” OSM.
- 255: Drogi klasy „motorway” OSM.

Elementy układu komunikacyjnego znajdujące się pod ziemią zostały przedstawione kolorem odpowiadającej jej klasy i linią przerywaną.

Tak zdefiniowane trzy monochromatyczne mapy przestrzenne połączyłem w trójwymiarową macierz, którą można przedstawić także w postaci obrazu RGB o ośmiobitowej głębi koloru (Rys. 61).



Rys. 61. Mapa przestrzeni cech wykorzystanej w części eksperymentalnej. Mapa składa się z trzech, nałożonych na siebie map dwuwymiarowych przedstawionych na potrzeby wizualizacji w trzech kolorach podstawowych. Kolorem niebieskim oznaczono mapę topografii i zabudowy terenu, zielonym – intensywność zieleni, czerwonym – układ komunikacyjny. Opracowanie własne.

W rzeczywistej przestrzeni trójwymiarowej taka mapa cech może być także przedstawiona jako pięciowymiarowy model zwokselizowany, w którym pierwszy wymiar stanowi długość geograficzna, drugi – szerokość geograficzna, trzeci – wysokość względna topografii i zabudowy, czwarty – intensywność zieleni, a piąty – intensywność układu komunikacyjnego (Rys. 62).



Rys. 62. Diagram przedstawiający możliwość przedstawienia przestrzeni cech w macierzowej postaci trójwymiarowej (po lewej) oraz w zwokselizowanej postaci pięciowymiarowej (po prawej). W obu interpretacjach osie *X* i *Y* odpowiadają współrzędnym geograficznym. Opracowanie własne.

Zakres pola percepcji

Ze względu na planowane wykorzystanie w eksperymencie głównym sieci typu GAN, opartej na nie w pełni splotowych sieciach neuronowych, pole percepcji na wejściu sieci neuronowej musiało mieć stałe wymiary – zarówno na etapie treningu, jak i inferencji. Konieczne było więc zdefiniowanie jednej skali i obszaru percepcji wykorzystanego w badaniu. Uwzględniając fraktalny charakter kompozycji przestrzennych, nie można obiektywnie przyjąć jakiegokolwiek skali analizy przestrzennej za bardziej trafną, bez uwzględnienia docelowego, interesującego nas problemu kompozycyjnego. W badaniu głównym zdecydowałem się na aplikację narzędzia do lokalnej skali urbanistycznej, która najczęściej pojawiała się w literaturze przedmiotu. Na potrzeby części eksperymentalnej niniejszej pracy za obszar pola percepcji w skali urbanistycznej przyjąłem kwadrat o bokach długości 1 km. Według Gehla obszar ten odpowiada średnim rozmiarom większości centrów miejskich takich miast jak Zurych, Brisbane, Pittsburg czy Kopenhaga. Promień tego obszaru, wynoszący 450 - 500 m, wyznaczony jest przez zasięg pięciominutowego spaceru. Powyżej tej wartości miejsce docelowe wydaje się odległe (Gehl, 2014). Obszar ten wpisuje się w także zakres rozpiętości obszarów zajmowanych przez „Społeczności 7000” definiowanych przez Christophera Alexandra. Według Alexandra w grupie 7000 osób pojedynczy człowiek nadal ma siłę oddziaływania na całą społeczność. Zbiorowiska ludzi powyżej granicy 10000 osób zaczynają działać dehumanizująco, głos pojedynczego człowieka przestaje być słyszalny i ważny.

Za przestrzenne granice obszaru społeczności 7000 Alexander przyjmuje od 0,25 do 2 mil (czyli 0,4 do 3,2 km). Gdy uwzględnimy rozmiary sąsiedztw centrów ekscentrycznych Alexandra (Rys. 25) wraz z tworzącymi je barierami i ścieżkami naprowadzającymi, zakres ich oddziaływania będzie wynosił około 1 km (Alexander, i in., 1977).

Dodatkowo ze względu na ograniczenia sprzętowe, konieczne było operowanie na niewielkich obrazach. Przyjęte mapy cech mają więc wymiary 512 x 512 pikseli²⁶, co odpowiada obszarowi o przybliżonych rozmiarach 1 x 1 km. Zakres taki pozwala na uchwycenie kompozycji przestrzennych w lokalnych skalach urbanistycznych przy jednoczesnym uchwyceniu sporych fragmentów głównych osi naprowadzających w skalach większych. Pokrycie analizą większych obszarów jest możliwe poprzez sekwencyjne przetwarzanie wielu, nakładających się na siebie kafelków pól percepcji. Mapy tworzone w serwisie Mapbox domyślnie korzystają z odwzorowania Merkatora, w którym im bardziej zbliżamy się do biegunów, tym bardziej zwiększa się deformacja powierzchni (PWN, 2024). Dla obszarów zastosowanych w badaniu (wszystkie znajdują się w strefie umiarkowanej) 1 piksel odpowiada średnio kwadratowi o bokach 2 x 2 m. Wartość ta jest zgodna z także wertykalnym skokiem jednostek w mapie topografii i zabudowy, który wynosi równo 2 m co jednostkę. Dwumetrowy limit skoku jednostek topografii wynika z ograniczenia serwisu Mapbox, który w momencie przygotowywania prototypu nie umożliwiał wyświetlania map topograficznych w wyższej rozdzielczości.

Wybór modelowych punktów głównych, tworzenie zbioru treningowego i testowego

Do wytrenowania sieci neuronowej potrzebny był możliwie jak największy zbiór treningowy oparty na różnych przykładach punktów głównych w kompozycjach o skali 1 km kwadratowego. Współczesne sieci neuronowe operują na ogromnych zbiorach danych składających się nawet z miliardów przykładów treningowych (Schuhmann, i in., 2022).

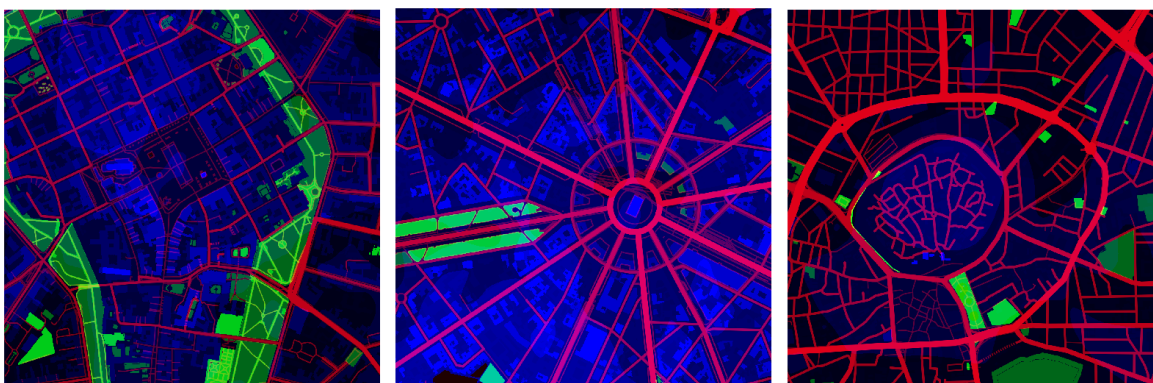
²⁶ W ramach strojenia hiperparametrów przetestowałem także algorytmy oparte na polach percepcji 256 x 256 px oraz 768 x 768 px. W modelu 256 x 256 px przestrzeń cech była zbyt mocno zniekształcona przez skalowanie, przez co poszczególne cechy stały się nieczytelne. We wstępnych eksperymentach przy treningu zbyt często następował kolaps sieci skutkujący generowaniem przez nią jedynie czarnych obrazów. Sieć 768 x 768 px wykazała obiecującą skuteczność przy wstępnych testach ze względu na pole percepcji rozszerzone do 1,5 x 1,5 km, jednak trening sieci był zdecydowanie zbyt wolny (nastąpiło ponad dwukrotne spowolnienie), przez co strojenie hiperparametrów na dostępnym sprzęcie okazało się zbyt długotrwałe.

Ze względu na konieczność ręcznego wyboru modelowych punktów głównych i ich ręczne oznaczenie, opracowanie dużego zbioru treningowego w czasie przeznaczonym na badanie nie było możliwe. Na potrzeby prototypu przygotowałem niewielki zestaw danych złożony z 75 przykładów treningowych i 32 przykładów testowych. Zastosowałem technikę rozszerzenia zbioru danych metodami odbicia lustrzanego, losowego obrotu i kadrowania obrazów. W ten sposób uzyskałem 3000 przykładów treningowych i 160 przykładów testowych. Przykłady treningowe zostały wybrane z przeanalizowanych źródeł dotyczących architektoniczno-urbanistycznej kompozycji przestrzennej. Początkowa lista złożona była ze 143 przykładów. Ich pełna lista załączona jest w Aneksie (Załącznik 4). Przykłady obejmowały:

- 14 przykładów z *Obrazu Miasta* Kevina Lyncha (Lynch, 1960).
- 51 przykładów z *Elementów Kompozycji Urbanistycznej* Kazimierza Wejcherta (Wejchert, 1984).
- 27 przykładów z *O Budowie Formy Architektonicznej* Juliusza Żórawskiego (Żórawski, 1962).
- 27 przykładów z *Architecture Form, Space, and Order* Francisa D.K. Chinga (Ching, 2007).
- 15 przykładów ze *Studium Uwarunkowań i Kierunków Zagospodarowania Przestrzennego Miasta Łodzi* (Prezydent miasta Łodzi, 2018).
- 2 przykłady z archiwalnego studium Łodzi (Prezydent miasta Łodzi, 2010).
- 8 własnych przykładów znanych mi z rodzinnego miasta Łodzi.

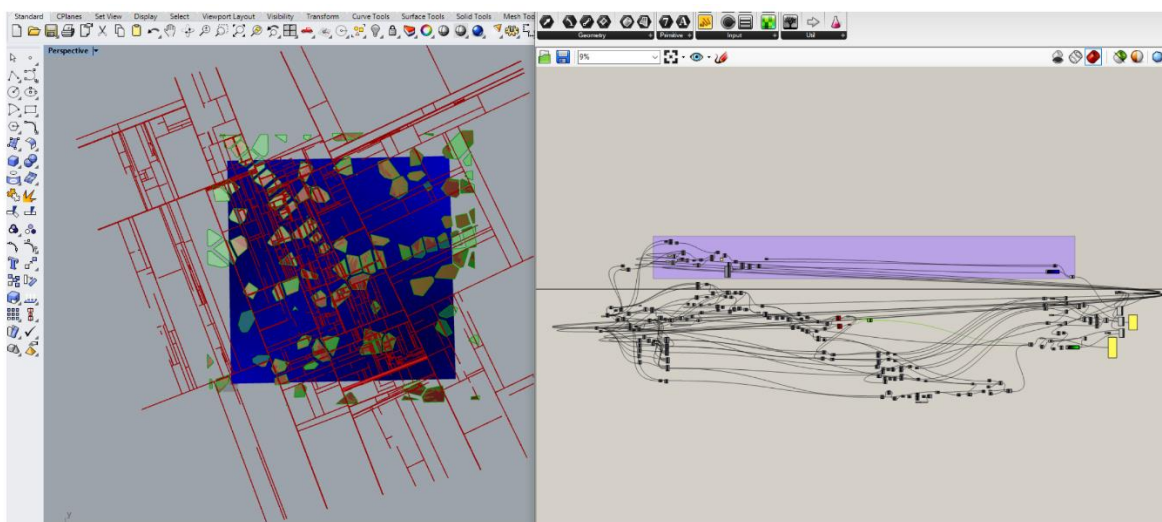
Lista została następnie zredukowana do 75 pozycji. Usuniętych zostało 13 dominant niebędących punktami głównymi, 9 przykładów już nieistniejących lub zdezaktualizowanych przez zmiany w okolicznej zabudowie oraz 46 przykładów punktów formalnie podkreślonych niebędących punktami głównymi, lub punktów głównych manifestujących się w innych skalach. Do zbioru testowego wybrałem 32 miejsca z krajobrazu Warszawy reprezentujące zróżnicowane strukturalno-przestrzennie fragmenty zabudowy obejmujące tereny poprzemysłowe, większe założenia kompozycyjne, zabudowę jednorodziną, zabudowę mieszkaniową wysokościową, historyczną tkankę śródmiejską, okolice parków i lasy miejskie, tereny wzdłuż rzeki, monofunkcyjną zabudowę biurową, tereny o dużym gradiencie topograficznym wzdłuż skarpy, urbanizowane tereny porolne, tereny z dominantami przestrzennymi oraz zabudowę otaczającą forty Twierdzy Warszawa.

- Plac Unii Lubelskiej (fragment założenia ujazdowskiego).
- Plac Zbawiciela + place okoliczne (fragment założenia ujazdowskiego).
- Pałac w Wilanowie (barokowe założenie kompozycyjne).
- Plac Narutowicza (radialne założenie kompozycyjne).
- Okolice muzeum Fryderyka Chopina (zabudowa na skarpie).
- Osiedle Zacisze (zabudowa jednorodzinna).
- Saska Kępa, ul. Francuska (zabudowa jednorodzinna).
- Boernerowo (zabudowa jednorodzinna).
- Ursynów Północny (wysokościowa zabudowa mieszkaniowa).
- Ursynów Południowy, Natolin (wysokościowa zabudowa mieszkaniowa).
- Okolice stacji metra Młociny (tereny przemysłowe, zabudowa wielorodzinna).
- Sady Żoliborskie (historyczna zabudowa śródmiejska).
- Stary Mokotów (historyczna zabudowa śródmiejska).
- Okolice Cerkwi Praskiej (historyczna zabudowa śródmiejska).
- Stara Praga (historyczna zabudowa śródmiejska) .
- Okolice muzeum Powstania Warszawskiego (zróżnicowana zabudowa z nową zabudową biurową).
- Okolice Domaniewskiej (monofunkcyjna zabudowa biurowa).
- Zakłady FSO (tereny przemysłowe).
- Ul. Modlińska na Białołęce (zabudowa porolna i przemysłowa).
- Łazienki i Pałac na Wyspie (park).
- Północna krawędź Pola Mokotowskiego (park).
- Las Bielański (las miejski).
- Bulwary Wiślane (teren przy rzece).
- Okolice Stadionu Narodowego (dominanta przestrzenna).
- Fort VIIA Służewiec (fort Twierdzy Warszawa).
- Fort Szczęśliwice (fort Twierdzy Warszawa).
- Fort Chrzanów (fort Twierdzy Warszawa).
- Fort Blizne (fort Twierdzy Warszawa).
- Fort Wawrzyszew (fort Twierdzy Warszawa).
- Fort Bema (fort Twierdzy Warszawa).
- Fort Służew (fort Twierdzy Warszawa).



Rys. 63. Przykłady trzech map cech ze zbioru treningowego sieci. Kanał niebieski przejawskrawiony na potrzeby wizualizacji. Od lewej Stare Miasto w Krakowie, Łuk Triumfalny w Paryżu i Cytadela w Irbilu. Opracowanie własne. Mapa cech przygotowana w oparciu o serwis Mapbox (Mapbox, 2023) i Open Street Map (OpenStreetMap contributors, 2023).

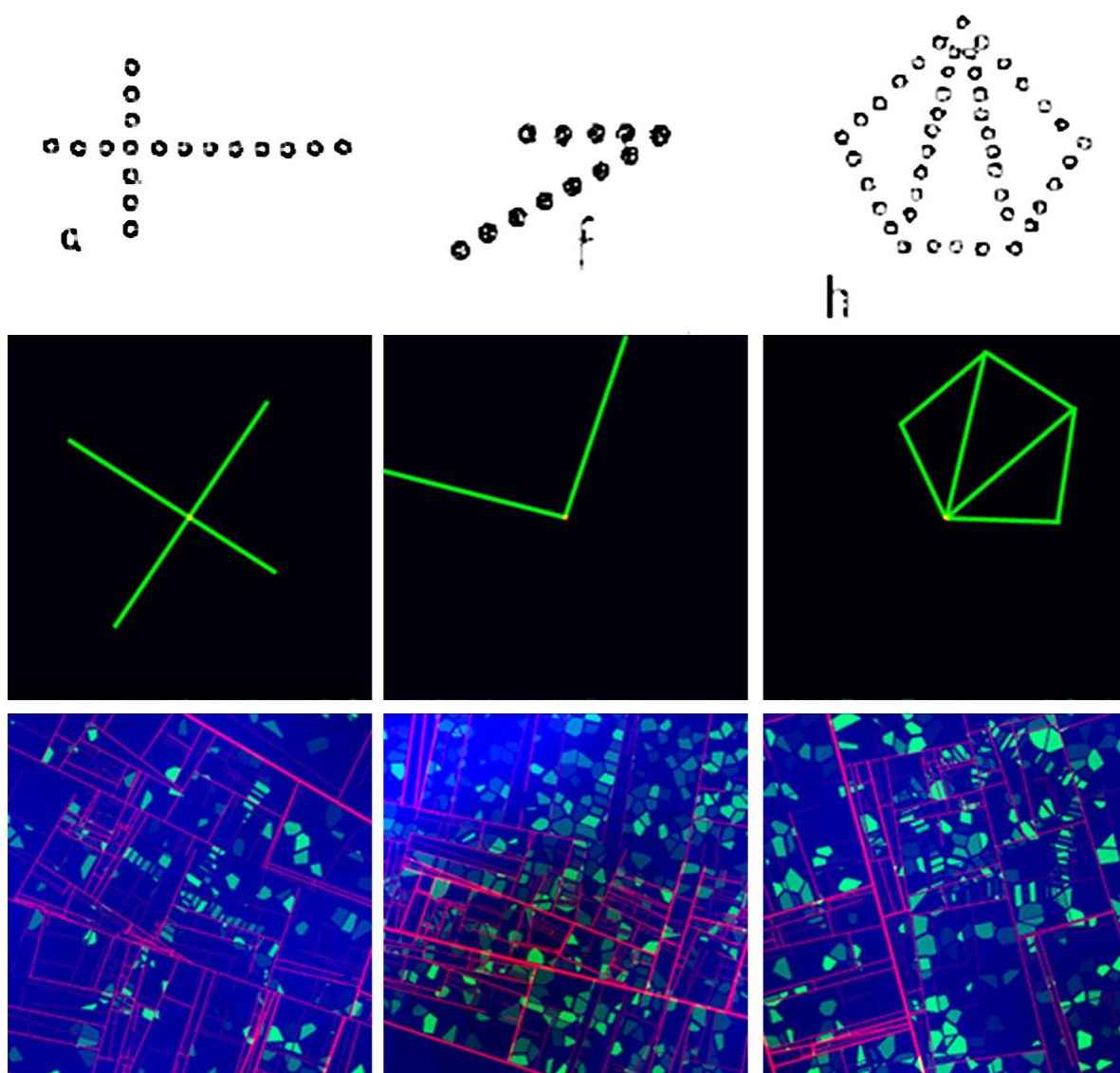
Obrazy do zbioru treningowego i testowego zostały pobrane z serwisu Mapbox przy pomocy skryptu API napisanego w języku Python. Skrypt pobierał dla zadanych współrzędnych poszczególne mapy cech i łączył je w jeden obraz RGB. Pobrane obrazy miały rozmiar 1024 x 1024 px i obejmowały obszar 2 x 2 km, dzięki czemu możliwe było późniejsze losowe kadrowanie kompozycji do docelowego rozmiaru 512 x 512 px i losowy obrót obrazu (Rys. 64). W celu uniknięcia sytuacji, w których treningowy punkt główny znajdowałby się na samej krawędzi obrazu, kadrowanie zostało ograniczone przez nieprzekraczalny margines o szerokości 32 px. By nie zaburzyć kompozycyjnych relacji północ-południe, losowy obrót obrazów został ograniczony do zakresu kątów od -15° do 15° . Obrazy testowe nie były losowo obracane.



Rys. 64. Próby stworzenia algorytmu rozszerzającego zestaw treningowy o przykłady syntetyczne. Środowisko Grasshopper. Ostatecznie algorytm nie został wykorzystany.

Opracowanie własne.

Ze względu na bardzo mały zestaw treningowy, oprócz wykorzystania powyższych technik początkowo planowałem rozszerzenie zestawu treningowego o przykłady wygenerowane syntetycznie, podobnie jak w przypadku eksperymentu wstępnego (Dzieduszyński, 2022a). W tym celu przygotowałem w środowisku Grasshopper algorytm parametryczny tworzący kompozycję przestrzenną udającą formalnie rzeczywistą kompozycję miejską poprzez zadane zasady kompozycyjne i serię generatorów pseudolosowych gwarantujących zróżnicowane wyniki (Rys. 64). Wykorzystując zasady kompozycyjne zdefiniowane przez Żórawskiego (Żórawski, 1962), algorytm próbował generować obrazy jak najlepiej wpisujące się w przestrzeń cech. W momencie opracowywania tego narzędzia, przestrzeń cech miała trochę inną strukturę, w której w kanale zielonym obrazu zamiast zieleni zapisana była zabudowa, a kanał niebieski odpowiadał jedynie topografii terenu. Kanał czerwony, tak jak w docelowej przestrzeni cech, opisywał układ komunikacyjny (Rys. 65).



Rys. 65. W górnym rzędzie – przykłady trzech zasad kompozycyjnych podporządkowanych punktom głównym zdefiniowane przez Żórawskiego (Żórawski, 1962). W środkowym rzędzie – osie kompozycyjne i punkty główne stanowiące punkty wyjścia dla algorytmu tworzącego syntetyczne przykłady treningowe. W dolnym rzędzie – wygenerowane przez algorytm syntetyczne konteksty przestrzenne. Ostatecznie algorytm nie został wykorzystany.
Opracowanie własne.

W pierwszej kolejności przy pomocy generatorów pseudolosowych algorytm definiował środkowy punkt główny i podporządkowane mu główne osie naprowadzające. W drugiej kolejności na podstawie zdefiniowanych osi i punktu głównego, generowana była sieć komunikacyjna oraz symulowana zabudowa wpisująca się w utworzone kwartały. W ostatnim kroku generowany był pseudolosowy gradient topografii terenu. Powstałe obrazy niestety były

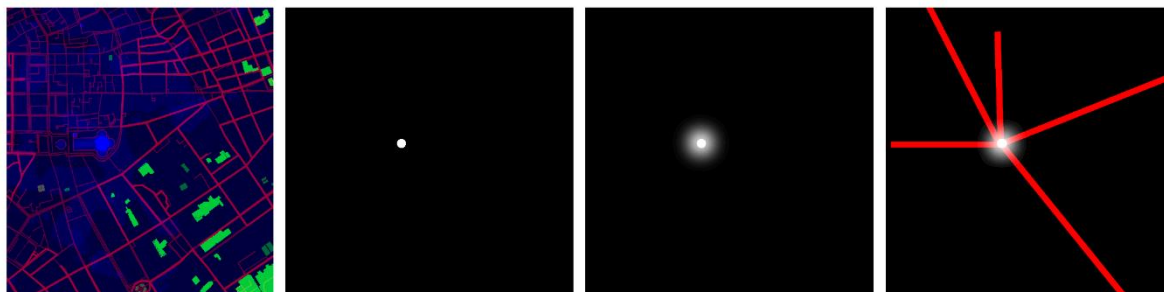
zbyt uproszczone i nie symulowały w odpowiednio realistyczny sposób rzeczywistych map kontekstów przestrzennych. Innymi słowy, w przestrzeni cech grupa obrazów syntetycznych była zbyt odległa od grupy obrazów treningowych. Wykorzystanie tych obrazów do treningu sieci neuronowej prowadziłoby do problemu przesunięcia dystrybucji (ang. covariate shift), w którym sieć zamiast uczyć się o kompozycji miast, uczyłaby się o syntetycznych obrazach stworzonych w Grasshopperze. Dodatkowo, w pierwotnej, nieoptymalizowanej wersji algorytmu, wygenerowanie pojedynczego przykładu treningowego zajmowało dwie minuty, co znacząco spowalniało iteracyjne poszukiwanie najlepszych parametrów. Ostatecznie ze względu na napotkane trudności algorytm do syntetycznego rozszerzania zbioru danych nie został wykorzystany i sieć neuronowa była wytrenowana jedynie na rzeczywistych przykładach punktów głównych zidentyfikowanych w analizowanej literaturze.

Oznaczanie punktów głównych – mapa celu

Omówione mapy cech stanowiły dla projektowanej sieci neuronowej dane wejściowe. Danymi wyjściowymi były mapy celu w przestrzeni celu (ang. target space), przedstawiające odczytywane przez sieć neuronową punkty główne. Format map celu został ustalony w oparciu o źródła literaturowe dotyczące kompozycji przestrzennej i eksperymenty na etapie strojenia hiperparametrów systemu.

Czy istnieje jakaś praktyczna granica odróżniająca pojedynczy punkt od obszaru? W dużej mierze jest to kwestia umowna. W *Miastach dla ludzi* Jan Gehl definiuje trzy wartości graniczne wyznaczające odległości „publiczne społeczne”. W najbardziej intymnej odległości 10 m mamy możliwość pełnego odbierania drugiego człowieka wszystkimi zmysłami. Od 10 m do 25 m nadal możemy odbierać emocje i ruch drugiej osoby, ale ze zmysłów pozostają nam do dyspozycji jedynie wzrok i słuch. Odległość 100 m to maksymalny dystans wzrokowego odbioru ruchu innej osoby. Powyżej tej wielkości postrzegany człowiek przestaje być czytelny, odbierany jest wtedy jedynie jako niewyraźna plamka. Odległość ta wyznacza także maksymalną rozpiętość pojedynczego miejsca, np. placu miejskiego (Gehl, 2014). Można uznać, że w przypadku punktów głównych stanowiących jednocześnie przestrzeń publiczną jego maksymalny promień może wynosić połowę tego dystansu czyli 50 m. Powyżej tej granicy ciężko mówić już w kontekście urbanistycznym o pojedynczym punkcie. Należałoby raczej zacząć nazywać daną strukturę obszarem. W praktyce wymieniane w literaturze punkty główne o skali urbanistycznej wpisujące się w obszar percepcji 1 x 1 km rzadko mają aż takie rozmiary (Załącznik 4). Średnica większości z nich zawiera się w zakresie od 10 do 50 m. Za średnicę

punktu głównego w przestrzeni celu przyjąłem medianę tych wartości wynoszącą 32 m. W przestrzeni celu o zakresie 512 x 512 px przekłada się to na średnicę punktu wynoszącą 16 pikseli.



Rys. 66. Ewolucja mapy celu dla sieci neuronowej na przykładzie florenckiej katedry Santa Maria del Fiore. Od lewej do prawej: mapa cech (kanał niebieski przejaskrawiony na potrzeby wizualizacji); pierwsza generacja mapy celu z pojedynczym punktem głównym; druga generacja mapy celu uzupełniona o gradient dookoła punktu (gradient przejaskrawiony na potrzeby wizualizacji) oraz ostateczna mapa celu uzupełniona o osie kompozycyjne (gradient przejaskrawiony na potrzeby wizualizacji). Opracowanie własne. Mapa cech przygotowana w oparciu o serwis Mapbox (Mapbox, 2023) i Open Street Map (OpenStreetMap contributors, 2023).

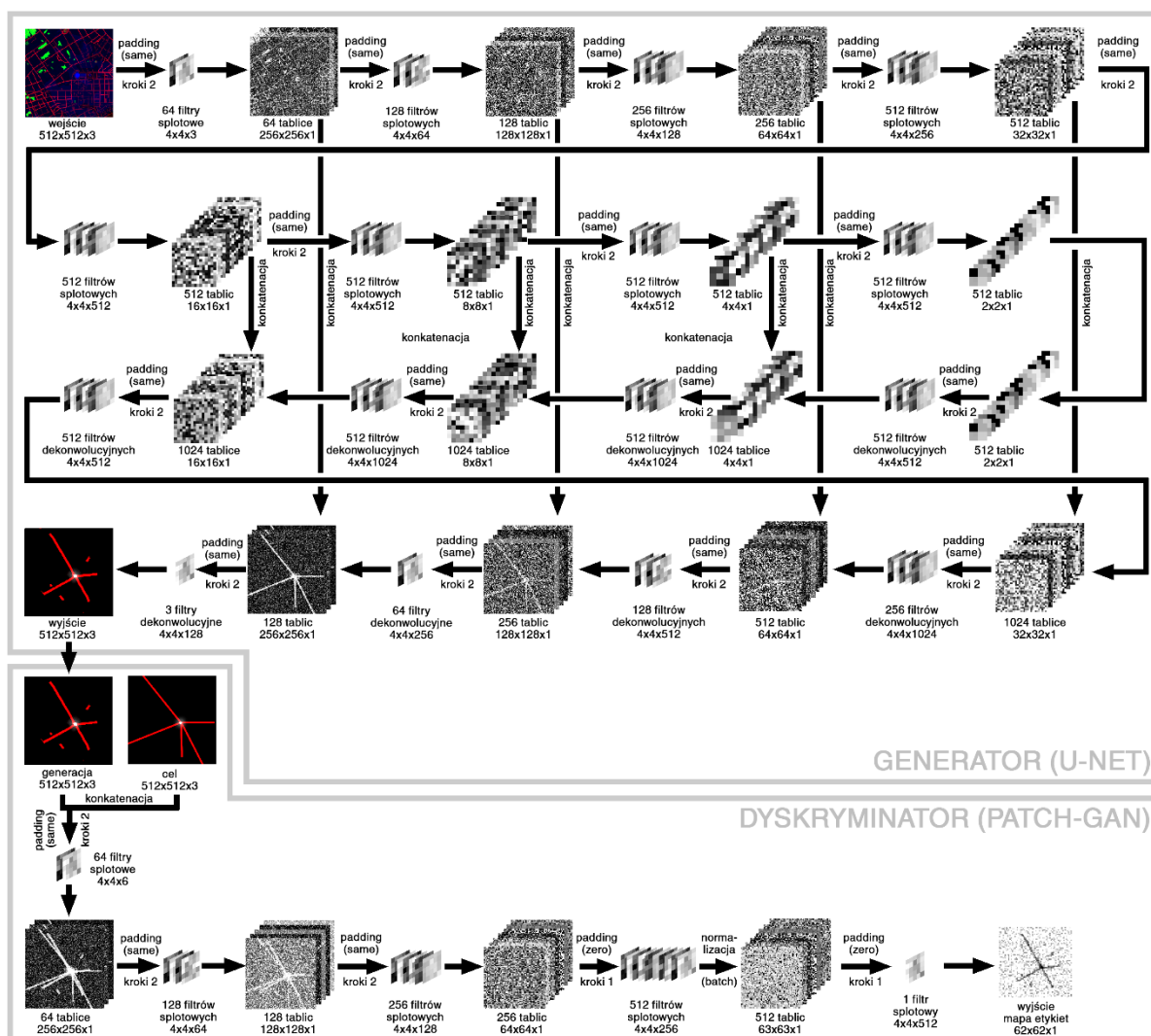
Początkowo planowałem przedstawić w przestrzeni celu jedynie monochromatyczne punkty główne, jednak przy tak sformatowanych danych treningowych testowane sieci nie uczyły się prawidłowo. Waga czarnego tła względem drobnych punktów okazywała się tak duża, że w większości przypadków następował kolaps²⁷ sieci do generowania pustych, czarnych obrazów. W celu zwiększenia wagi punktów dookoła każdego punktu dodałem warstwę z gaussowskim rozmyciem punktu o średnicy 86 pikseli. Dzięki utworzonemu w ten sposób gradientowi nie tylko zwiększyła się waga niepustych pikseli w reprezentacji celu, ale jednocześnie funkcja straty sieci neuronowej została odwrotnie skorelowana z odległością wygenerowanego przez sieć punktu od punktu docelowego. Nawet gdy wygenerowany przez sieć punkt znajdował się poza 16-pikselowym obrysem punktu docelowego, sieć dostawała

²⁷ Kolaps sieci typu GAN oznacza sytuację, w której generator przestaje tworzyć zróżnicowane dane wyjściowe i zaczyna generować bardzo podobne lub identyczne przykłady. Prowadzi to zazwyczaj do znacznej dysproporcji między stratą generatora i dyskryminatora, co efektywnie zatrzymuje dalszy trening sieci.

pewną nagrodę, o ile punkt znalazł się w obrębie 86-pikselowego gradientu. Im bliżej docelowego punktu znajdował się wygenerowany punkt, tym większą nagrodę otrzymywała sieć. Zmiana ta znacząco ułatwiła sieci znajdowanie docelowej pozycji punktu głównego. Dodatkową zmianą było dodanie czerwonego kanału, przedstawiającego osie kompozycyjne mające wpływ na dany punkt główny. Warstwa ta nie tylko zwiększyła skuteczność treningu sieci przez zwiększenie wagi niepustych pikseli w przestrzeni celu, ale jednocześnie może być wykorzystana pomocniczo przy interpretacji wygenerowanych przez sieć propozycji punktów głównych (Rys. 66).

Struktura sieci neuronowej

Korzystając z doświadczeń płynących z eksperymentu wstępnego i przeprowadzonego rozpoznania stosowanych w CAAD algorytmów, zdecydowałem się oprzeć strukturę sieci neuronowej na sieci typu CNN. Ze względu na większą złożoność problemu względem eksperymentu wstępnego, wykorzystałem bardziej złożony algorytm generatywnych sieci przeciwstawnych GAN inspirowany siecią pix2pix (Isola, i in., 2017), będącą jednym z najszerszej stosowanych algorytmów typu GAN. Opracowany algorytm składał się z dwóch części zbudowanych z prostszych sieci splotowych (Rys. 67). Generators opartego na sieci U-Net (Ronneberger, i in., 2015) i dyskriminatora opartego na patchGAN (Li i Wand, 2016). W czasie treningu obie sieci składowe uczyły się jednocześnie. Generator był odpowiedzialny za tworzenie obrazów wyjściowych w przestrzeni celu na podstawie kontekstów wejściowych. Dyskriminator próbował odróżnić obrazy tworzone przez generator od obrazów celu pochodzących ze zbioru treningowego. Generator był nagradzany za oszukanie dyskriminatora, a dyskriminator za rozróżnienie prawdziwych danych treningowych od tych syntetycznych (funkcja straty GAN loss). Dodatkowo generator był nagradzany za tworzenie obrazów, w których wartości pikseli były jak najbardziej zbliżone do wartości pikseli obrazów treningowych (funkcja straty L1 loss).



Rys. 67. Diagram modelu sieci neuronowej typu GAN wykorzystanej w eksperymencie głównym. Sieć inspirowana jest modelem pix2pix (Isola, i in., 2017). Generator tworzy mapy celu na podstawie map cech. Dyskryminator porównuje wyjście generatora z treningową mapą celu i próbuje oznaczać, które fragmenty obrazów są stworzone przez generator, a które pochodzą ze zbioru treningowego. Generator jest nagradzany za obrazy potrafiące oszukać dyskryminator. Obie sieci uczą się wspólnie, zwiększając wzajemnie swoją skuteczność. Przy inferencji wytrenowanego modelu wykorzystywany jest tylko generator. Opracowanie własne.

W celu zdefiniowania najlepszych hiperparametrów algorytmu, wykorzystałem metodę przeszukiwania losowego (ang. random search), zgodnie z którą losowo próbkowałem zestawy hiperparametrów z przedziałów najczęściej stosowanych w algorytmach wykorzystujących pix2pix rozpoznanych w kwerendzie. Zestawy hiperparametrów były następnie wykorzystane

do rozpoczęcia treningu sieci. W początkowej fazie treningu możliwe było odrzucenie dużej części zestawów, ponieważ prowadziły one do nieprawidłowego uczenia się sieci skutkującego jej kolapsem, znaczącym spowolnieniem treningu lub występowaniem zbyt silnego przetrenowania. Najlepsze zestawy hiperparamterów zostały wykorzystane do pełnego wytrenowania sieci. Ostateczny algorytm został wybrany w oparciu o jakościową ocenę generowanych obrazów, zarówno na zestawie treningowym, jak i testowym. Ze względu na ograniczone możliwości czasowe i moc obliczeniową, nie skorzystałem z bardziej systematycznej metody przeszukiwania siatki (ang. grid search).

Najważniejsze hiperparametry algorytmu obejmowały:

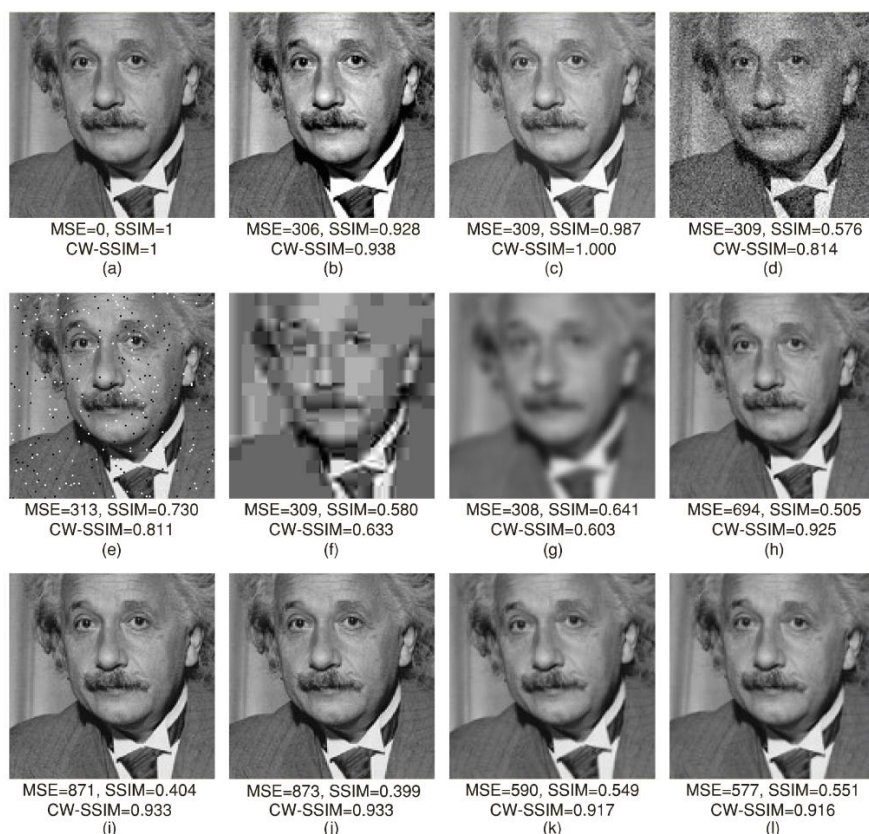
- Wymiary przestrzeni cech (512 x 512 x 3).
- Wymiary przestrzeni celu (512 x 512 x 3).
- Rozmiary obu sieci, liczbę parametrów w poszczególnych warstwach (około 53 miliony trenowalnych parametrów) (Załącznik 7).
- Liczbę i rodzaj warstw (warstwy splotowe) (Rys. 67).
- Funkcje aktywacji (ReLU, LeakyReLU i tanh).
- Parametr lambda decydujący o wagach strat GAN loss i L1 loss (parametr zmienny w trakcie treningu: $\lambda = 200$ przez pierwszą godzinę, $\lambda = 500$ przez kolejne 26h $\lambda = 1000$ przez kolejne 21h, $\lambda = 2000$ w ostatnich 84 godzinach).
- Optymalizator (Adam, $LR = 1e - 5$, $\beta_1 = 0,5$).
- Techniki regularyzacyjne (brak dropoutu).

Ostateczna wersja sieci neuronowej została wytrenowana w 132 godziny na karcie graficznej Nvidia GF RTX 2080 Super. Dyskryminator wykorzystany był jedynie w trakcie treningu. Po wytrenowaniu sieci, w momencie inferencji (rozpoznawania głównych punktów kompozycyjnych) używany jest tylko generator.

Testowanie algorytmu

Wytrenowany algorytm został przetestowany na opracowanym zbiorze testowym (Załącznik 5). W celu rozszerzenia zbioru testowego wykorzystałem metodę losowego przesuwania kadru. Zbiór testowy został rozszerzony do 160 obrazów. Ocena skuteczności została oparta na analizie ilościowej i jakościowej. Z popularnych w dziedzinie przetwarzania sygnału ilościowych metod ewaluacji obrazów warto wyróżnić błąd średniokwadratowy MSE (ang. Mean Squared Error), wskaźnik podobieństwa strukturalnego SSIM (ang. Structural Similarity

Index) oraz CW-SSIM (ang. Complex Wavelet Structural Similarity Index) (Wang i Bovik, 2008). Wszystkie te metody pozwalają na zestawienie ze sobą obrazu celu z obrazem testowanym w celu wyliczenia wartości podobieństwa między oboma obrazami. Wszystkie są szeroko stosowane i łatwe do implementacji ze względu na ich reprezentację w popularnych bibliotekach służących do przetwarzania zbiorów danych. MSE i SSIM są bardzo podatne na percepcyjnie nieistotne zmiany obrazu, takie jak drobne przesunięcia czy obroty. CW-SSIM radzi sobie z tymi sytuacjami o wiele lepiej (Rys. 68).



Rys. 68. Porównanie metod ewaluacji podobieństwa obrazów MSE, SSIM i CW-SSIM przygotowane przez Zhou Wanga i Alana C. Bovika. a) obraz źródłowy; b) zwiększony kontrast; c) przesunięcie luminancji; d) szum gaussowski; e) szum impulsowy; f) kompresja JPEG; g) rozmycie; h) oddalenie; i) przesunięcie w prawo; j) przesunięcie w lewo; k) obrót przeciwny do ruchu wskazówek zegara; l) obrót zgodny z ruchem wskazówek zegara (Wang i Bovik, 2008). W przypadku opracowanego algorytmu żadna z powyższych metod ewaluacji nie była skorelowana z rzeczywistą jakością tworzonych przez sieć neuronową obrazów.

Mimo tego, ze względu na specyfikę zbioru testowego oraz całej przestrzeni celu, żadna z tych metod nie mogła być wykorzystana w ewaluacji ilościowej wytrenowanej sieci neuronowej. Ten sam punkt główny można wskazać na podstawie wielu różnych zestawów linii naprowadzających. W zbiorze treningowym liczba pikseli linii naprowadzających jest kilkudziesięciokrotnie większa od liczby pikseli samych punktów głównych, co przełożyłoby się na wyliczenie bardzo dużego błędu w sytuacji uwzględnienia alternatywnych linii naprowadzających. Co więcej, nawet uwzględnienie jedynie kanału zawierającego sam punkt główny nie rozwiązuje problemu. Kompozycja urbanistyczna ma charakter palimpsestyczny. W obrębie jednego kadru pola percepcji mogą nakładać się różne kompozycje przestrzenne podporządkowane różnym punktom głównym. Uwzględnienie alternatywnej kompozycji bazowej skutkujące wygenerowaniem innego punktu głównego nie stanowiłoby więc błędu. Z tego powodu w ramach analizy ilościowej wziąłem pod uwagę głównie kryterium wygenerowania przynajmniej jednego punktu głównego. Po spełnieniu warunku ilościowego sieć była testowana metodą jakościową względem czterech możliwych przypadków rozpoznania punktu głównego:

- Czy zaznaczony punkt to **dominanta** lub **punkt orientacyjny** o znaczącej skali i lokalizacji podkreślonej przez kompozycję miejską i/lub topografię terenu?
- Czy zaznaczony punkt jest **zakończeniem założenia osiowego**?
- Czy zaznaczony punkt jest **punktem centralnym** w kompozycji radialnej?
- Czy zaznaczony punkt jest jednym z głównych **węzłów założenia sieciowego**?

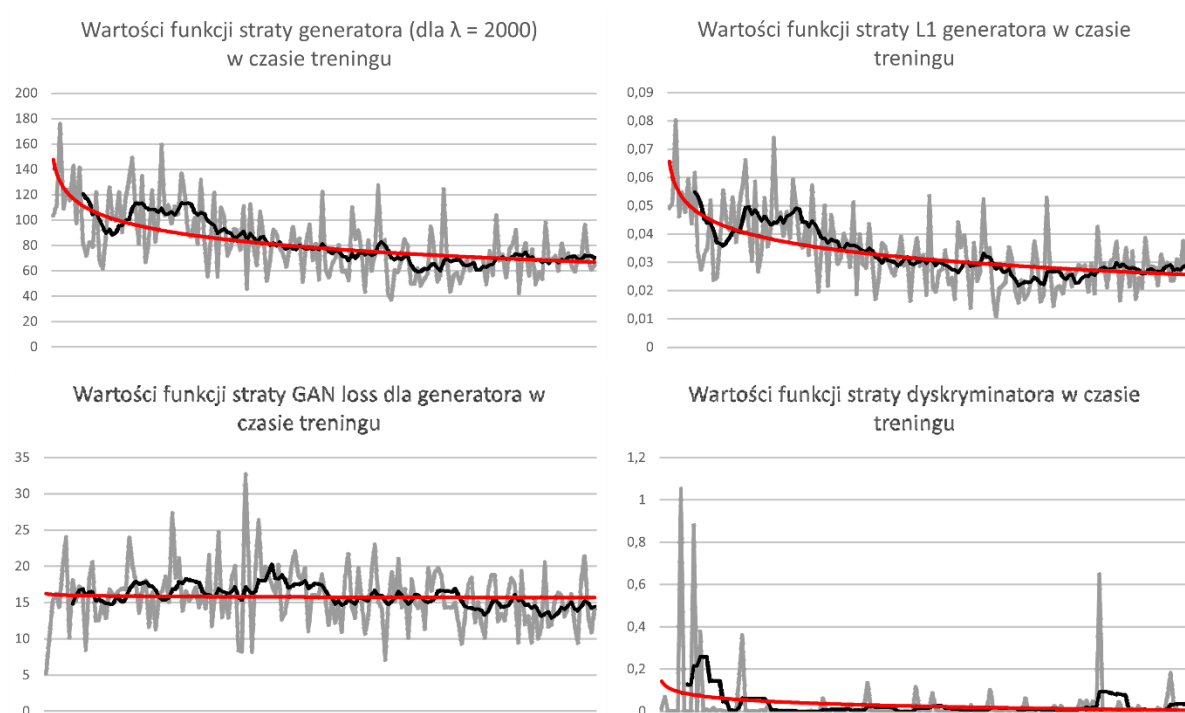
Jeśli przynajmniej jedno z kryteriów było spełnione, punkt główny zostawał uznawany za prawidłowy. Zanotowane zostały przypadki, w których pominięcie wygenerowania punktu głównego nie stanowiło błędu ze względu na charakterystykę danej mapy cech. Dodatkowo, sprawdzana była sensowność wygenerowanych linii naprowadzających, jednak ich jakość nie wpływała na ewaluację dotyczącą punktów głównych. Ewaluacja jakościowa była przeprowadzana metodą analizy eksperckiej na każdym ze 160 przykładów testowych (Załącznik 5). Pomocniczo, statystyki dotyczące liczby oznaczonych przez sieć neuronową pikseli zostały wykorzystane do ewaluacji sieci neuronowej w zakresie seryjnego uczytelniania kompozycji przestrzennych.

3.2.3. Wyniki eksperymentu głównego

Wyniki ilościowe

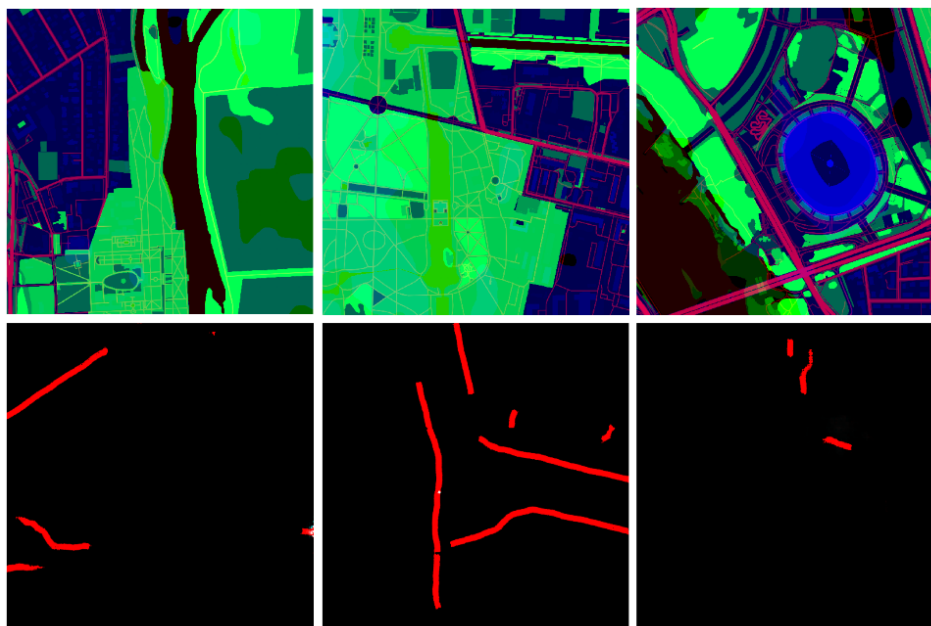
W czasie treningu trwającego 132 godziny sieć neuronowa uczyła się rozpoznawać punkty główne w zadanych kontekstach przestrzennych zbioru treningowego. Trening został zatrzymany w momencie stabilizacji funkcji straty. Dodatkowe wydłużanie treningu nie przynosiło wymiernych korzyści. Dyskryminator już w pierwszych godzinach treningu osiągnął bardzo dobrą skuteczność w rozpoznawaniu obrazów tworzonych przez generator. Z tego powodu funkcja straty „GAN loss” przez większość treningu pozostała stabilna. Mimo tego, ze względu na zwiększającą się w trakcie treningu wagę straty L1 generatora (zwiększający się parametr λ), generator stopniowo poprawiał swoją skuteczność w tworzeniu obrazów wpisujących się w przestrzeń celu. Wykresy przedstawione poniżej przedstawiają na osi pionowej wartości funkcji straty w trakcie treningu. Oś pozioma odpowiada kolejnym punktom kontrolnym (Rys. 69).

Wytrenowana sieć neuronowa została przetestowana na zbiorze testowym (Załącznik 5). W 25,6% obrazów testowych algorytm nie rozpoznał w mapach cech żadnego punktu głównego. W większości tych przypadków nierozpoznanie punktu głównego stanowiło zdecydowany błąd sieci. Szczególnie zastanawiające jest pominięcie w niektórych kadrach pola percepcji tak oczywistych punktów głównych, jak pałac w Wilanowie, pałac na wyspie w Łazienkach Królewskich czy Stadion Narodowy (Rys. 70). Zjawisko to występowało w 23,1% obrazów testowych. W niektórych obrazach wejściowych, tak jak przykładowo w przypadku Stadionu Narodowego, lekkie przesunięcie kadru skutkowało prawidłowym rozpoznaniem pominiętego punktu. W innych jednak, jak w przypadku pałacu w Wilanowie, sieć konsekwentnie odmawiała uznania punktu za główny w kilku różnych kadrach mapy cech.



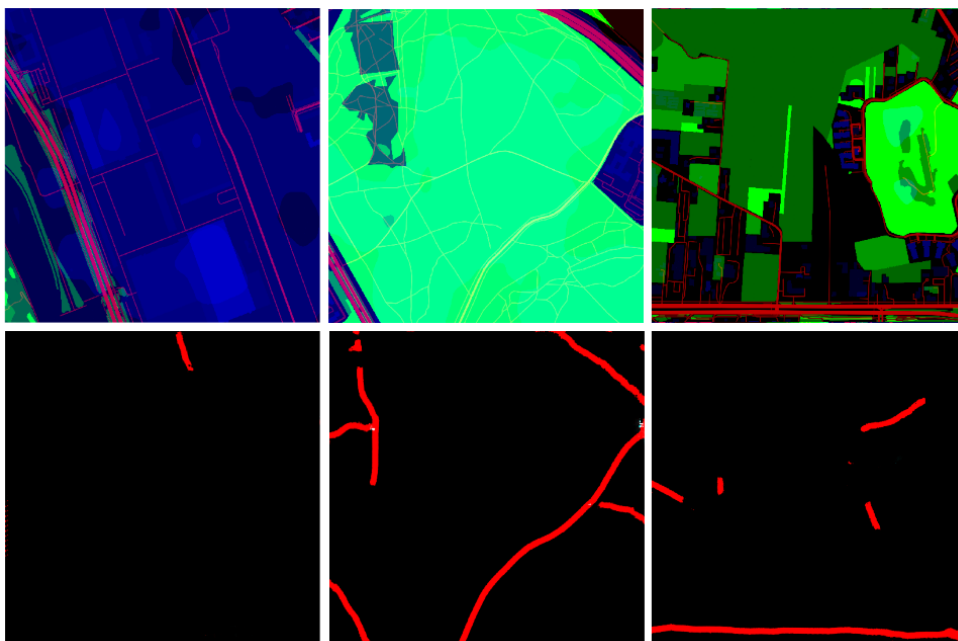
Rys. 69. Wartości funkcji straty na zbiorze treningowym dla punktów kontrolnych od 1 do 254. Oś horyzontalna odpowiada kolejnym punktom kontrolnym (ze względu na nierównomierne próbkowanie punktów kontrolnych oś ta nie stanowi osi czasu). Linia szara: wartości funkcji straty uśrednione z 4 podzbiorów treningowych. Linia czarna: średnia bieżąca z 9 kolejnych punktów kontrolnych (dla 4 podzbiorów treningowych). Linia czerwona: logarytmiczna linia trendu (dla 4 podzbiorów treningowych). Opracowanie własne.

W części zadanych kontekstów przestrzennych nie dało się jednak jednoznacznie wskazać żadnego punktu głównego. W tych przypadkach, stanowiących 2,5% zbioru testowego, pominięcie wygenerowania punktu nie stanowiło błędu. Przykładami takimi były mapy cech przedstawiające środek Lasu Bielańskiego, hale dawnych zakładów przemysłowych FSO czy pola uprawne w pobliżu Fortu Chrzanów (Rys. 71).



Rys. 70. Przykłady błędnych przypadków, w których nie został rozpoznany żaden punkt główny. Na górze mapy cech, na dole odpowiadające im obrazy stworzone przez sieć neuronową. Po lewej pałac w Wilanowie. Po środku Pałac na Wyspie w Łazienkach Królewskich. Po prawej Stadion Narodowy. Kanał niebieski map cech przejęskrawiony na potrzeby wizualizacji. Opracowanie własne.

W 74,4% zadanych kontekstów algorytm rozpoznał przynajmniej jeden punkt główny. 50,0% z nich stanowiło dominanty lub punkty orientacyjne o znaczącej skali i lokalizacji podkreślonej przez kompozycję miejską i/lub topografię terenu. 13,8% punktów głównych to zakończenia założeń osiowych. 10,6% to punkty centralne w kompozycji radialnej. Równorzędne węzły założeń sieciowych stanowiły 3,8% przypadków. Ogólna skuteczność sieci w rozpoznawaniu punktów głównych kompozycji przestrzennej, z uwzględnieniem przypadków, w których pominięcie punktu nie stanowiło błędu, wyniosła **59,4%**. Dodatkowo sprawdziłem, czy oznaczone przez sieć pomocnicze linie naprowadzające prawidłowo opisują zadane kompozycje przestrzenne. Jako ekstraktor linii naprowadzających, algorytm osiągnął skuteczność na poziomie **75,6%**.



Rys. 71. Przykłady, w których niewygenerowanie punktu głównego nie stanowiło błędu. Na górze mapy cech, na dole odpowiadające im obrazy stworzone przez sieć neuronową. Po lewej – hale przemysłowe zakładów FSO. W środku – fragment Lasu Bielańskiego. Po prawej – pola uprawne w pobliżu Fortu Chrzanów. Kanał niebieski map cech przejaśniony na potrzeby wizualizacji. Opracowanie własne.

Wsparcie projektowo-planistyczne

W celu zademonstrowania możliwości seryjnego przetwarzania większych obszarów, zaaplikowałem wytrenowaną sieć neuronową do lewobrzeżnego fragmentu zabudowy śródmiejskiej Warszawy. Obszar ten miał powierzchnię 25 km² i został podzielony na 100 kafelków odpowiadających pojedynczym polom percepcji opracowanego algorytmu (w przybliżeniu 1 km²). Kolejne kafelki były rozłożone na siatce o rozpiętości 250 m, więc każdy kolejny kafelek nakładał się na połowę powierzchni poprzedniego. 100 wynikowych obrazów wygenerowanych przez sieć neuronową zostało połączonych w jeden obraz, który nałożyłem na mapę Warszawy (Rys. 72). Algorytm na zadanym obszarze rozpoznał 64 punkty główne. 43 z nich spełniały warunki ewaluacji dla zbioru testowego. 2 punkty powtarzały się z punktami ze zbioru treningowego, więc nie zostały uwzględnione w wyliczeniu skuteczności (Załącznik 6). Na fragmencie śródmieścia Warszawy sieć osiągnęła wyższą skuteczność w rozpoznawaniu punktów głównych niż na zbiorze testowym. Skuteczność ta wyniosła **67,2%**.



Rys. 72. Demonstracja seryjnego przetwarzania większych obszarów przy pomocy opracowanej sieci neuronowej. 64 punkty główne i osie kompozycyjne rozpoznane przez sieć neuronową na obszarze 25 km² lewobrzeżnej zabudowy śródmiejskiej Warszawy.

*Po lewej – obrazy wygenerowane przez sieć nałożone na mapę przestrzeni cech.
Po prawej – nałożone na mapę Open Street Map. Czerwone linie to osie kompozycyjne.
Pełna lista punktów rozpoznanych przez sieć przedstawiona jest w aneksie (Załącznik 6).
Opracowanie własne. Mapa cech przygotowana w oparciu o serwis Mapbox (Mapbox, 2023) i Open Street Map (OpenStreetMap contributors, 2023).*

Niektóre ze wskazanych punktów na pierwszy rzut oka nie wydają się kompozycyjnie ważne, jednak po bliższym przyjrzeniu się otaczającej tkance miejskiej ich rola staje się jaśniejsza. Algorytm wskazał kilka punktów z potencjałem projektowym, którym powinna być poświęcona dodatkowa uwaga. Przykładem takiego punktu jest ostroga regulacyjna Wisły, znajdująca się na przedłużeniu ul. Ratuszowej. Znajdująca się na ostrodku ścieżka piesza nie tylko stanowi naturalne przedłużenie i zakończenie osi ul. Ratuszowej, ale podkreślona jest od południowej strony przez oś jednej z większych wysp piaszczystych i jednocześnie wyznacza bardzo atrakcyjny punkt widokowy otwierający się na Stare Miasto. Wskazana przez algorytm lokalizacja połączona jest z rekreacyjnymi sieciami pieszo-rowerowymi Parku Praskiego i warszawskiego ZOO, przez co wydaje się bardzo dobra dla lokalizacji na przykład reprezentacyjnego moła na Wiśle, mogącego dodatkowo powiązać miasto z rzeką (Rys. 73). Algorytm wskazał także szereg punktów o wysokiej wartości komercyjnej – zlokalizowanych w reprezentacyjnych obszarach biurowych i turystycznych, podkreślonych wizualnie przez

ekspozycje widokowe, powiązanych z dominantami wysokościowymi o dużej kubaturze i dobrze powiązanych z układem komunikacyjnym (Załącznik 6).



Rys. 73. Oznaczony czerwoną strzałką punkt wskazany przez sieć neuronową na ostrydze regulacyjnej Wisły na przedłużeniu ul. Ratuszowej. Punkt wydaje się dobrym miejscem na lokalizację reprezentacyjnego moła z widokiem na Stare Miasto. Opracowanie własne. Podkład przygotowany w oparciu o serwis Open Street Map (OpenStreetMap contributors, 2023).

Uczytelnianie kompozycji miejskiej

Oprócz wskazywania samych punktów głównych wytrenowana sieć neuronowa podświetla także osie naprowadzające, które zostały wykorzystane do wskazania danego punktu głównego. Pomaga to w interpretacji proponowanych przez algorytm punktów głównych. Po nałożeniu na mapę miasta oba te źródła informacji mogą być wykorzystane do uczytelniania kompozycji miejskiej. Przykład założenia ujazdowskiego w Warszawie pokazuje, że sieć prawidłowo podświetliła większość głównych osi budujących kompozycję założenia. Oznaczone zostały też niektóre z punktów głównych, takie jak południowa pierzeja placu Trzech Krzyży, skrzyżowanie osi przy placu Na Rozdrożu, północne i południowe otwarcie placu Konstytucji, skrzyżowanie Nowowiejskiej z Waryńskiego, i plac Unii Lubelskiej. Sieć pominęła kilka ważnych punktów, takich jak plac Zbawiciela czy Zamek Ujazdowski. Oś stanisławowska nie została także zaznaczona w rejonie parku Agrykola i wzdłuż Kanału Piaseczyńskiego. Mimo to, założenie ujazdowskie zostało oznaczone dosyć czytelnie w sposób umożliwiający jego identyfikację. Sieć neuronowa o wiele gorzej poradziła sobie z osią saską. Jako oś kompozycyjna podświetlony został jedynie jej krótki, zachodni fragment. Prawidłowo

podświetlona została też skośna oś poprzeczna ul. Przechodniej i Granicznej. Na samej osi saskiej został oznaczony tylko jeden punkt główny – Hala Gwardii (Rys. 74). Poza wymienionymi większymi założeniami kompozycyjnymi sieć neuronowa zidentyfikowała i oznaczyła także pewne mniejsze, warszawskie układy kompozycyjne. Wśród rozpoznanych przykładów warto wymienić bramę wyznaczoną przez wieżowce Forest i Babka Tower w obrębie strefy dominant wysokościowych ronda „Radosław”, założenie „Bramy Zachodniej”, czy zagłębienie biurowe nazywane potocznie przez mieszkańców warszawskim „Isengardem” (Załącznik 6). Warto też wymienić niektóre przykłady, w których sieć nie dała sobie rady z rozpoznaniem bardzo czytelnych kompozycji przestrzennych, jak choćby radialne założenie Placu Narutowicza, w którym algorytm oznaczył tylko części głównych osi naprowadzających, a znajdujący się na przecięciu osi kościół św. Jakuba nie został oznaczony jako punkt główny. Podobnie nie zostało rozpoznane znajdujące się w zbiorze testowym założenie wilanowskie.

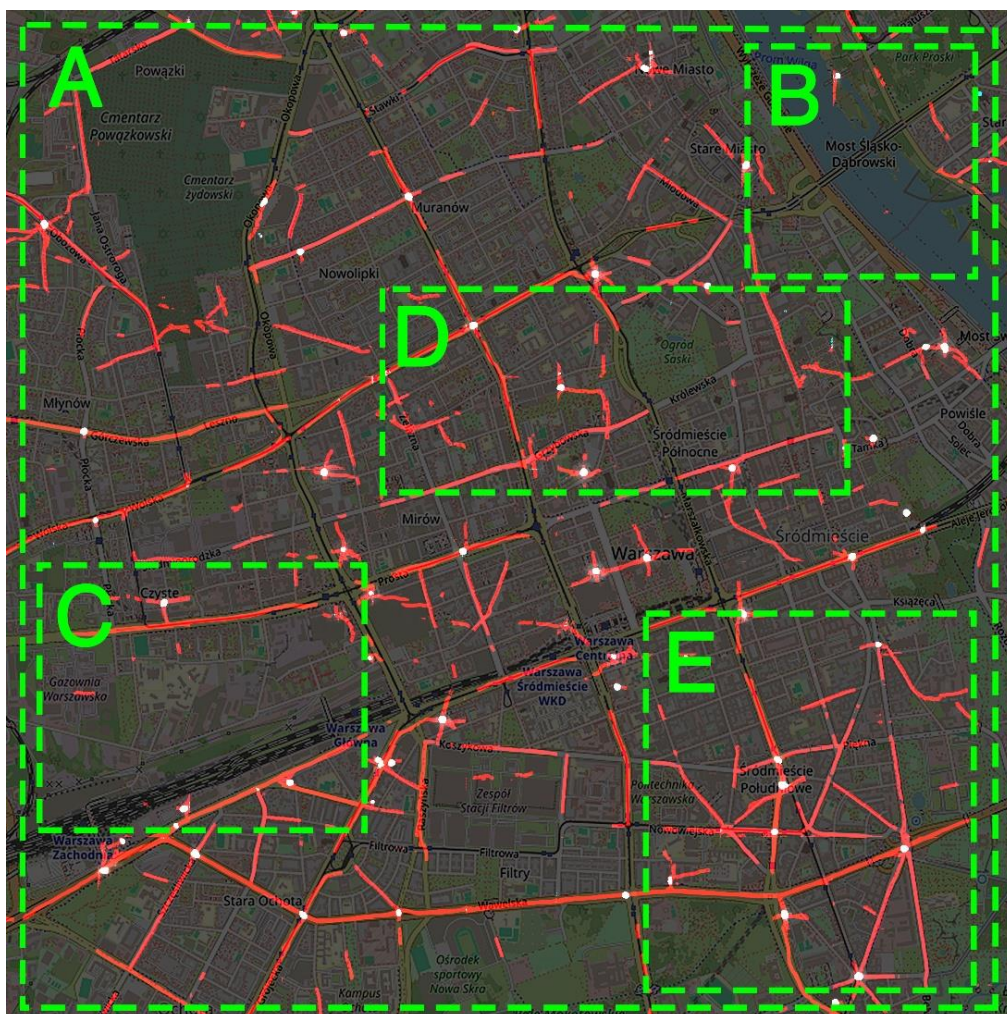


Rys. 74. Punkty główne (biały) i osie naprowadzające (czerwony) odczytane przez sieć neuronową w obszarze założenia ujazdowskiego (po lewej) i założenia osi saskiej (po prawej). Kolorem zielonym oznaczone główne osie kompozycyjne. Sieć skutecznie oznaczyła większość założenia ujazdowskiego, jednak oś saska została prawie całkowicie pominięta. Opracowanie własne. Podkład przygotowany w oparciu o Open Street Map (OpenStreetMap contributors, 2023).

W obecnej formie prototyp nie jest w stanie uchwycić wszystkich założeń kompozycyjnych. Bardzo trudno jest przygotować ilościowy test umożliwiający ujęcie tej cechy w postaci jakiejś obiektywnej metryki, jednak przykłady Wilanowa i nierozpoznanej osi saskiej demonstrują, że do interpretacji wyników wygenerowanych przez sieć neuronową trzeba podchodzić ostrożnie.

Oprócz jakościowej analizy generowanych przez sieć obrazów, w celu uczytelniania kompozycji przestrzennej możliwe jest wykorzystanie statystyk zliczających podświetlone przez sieć piksele, wpisujące się w kategorię punktów głównych i osi naprowadzających. W celu demonstracji tej metody z obszaru przetworzonego seryjnie fragmentu zabudowy śródmiejskiej Warszawy wyszczególniłem 5 obszarów o różnym „stopniu skomponowania” przestrzeni (Rys. 75):

- A: całość testowanego obszaru.
- B: okolice Wisły w rejonie mostu Śląsko-Dąbrowskiego.
- C: południowa granica Woli wzdłuż torów kolejowych.
- D: rejon osi saskiej.
- E: rejon założenia ujazdowskiego.

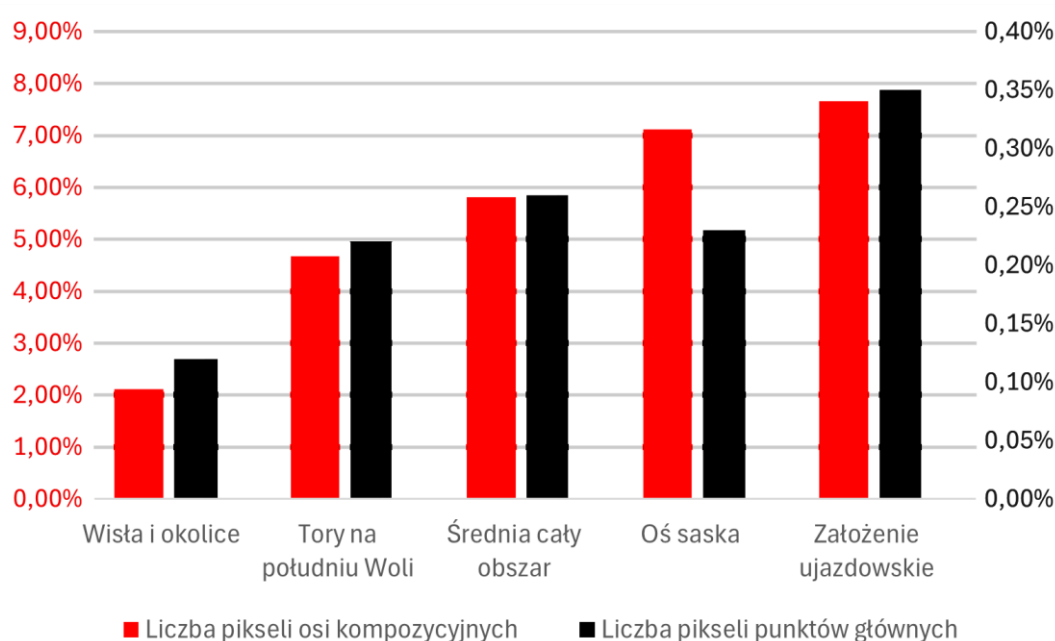


Rys. 75. Pięć obszarów przeanalizowanych statystycznie względem „stopnia skomponowania” przestrzeni. A: całość testowanego obszaru; B: okolice Wisły w rejonie mostu Śląsko-Dąbrowskiego; C: południowa granica Woli wzdłuż torów kolejowych; D: rejon osi saskiej; E: rejon założenia ujazdowskiego. Opracowanie własne. Podkład przygotowany w oparciu o Open Street Map (OpenStreetMap contributors, 2023).

Dla wyszczególnionych obszarów zostały zliczone liczby pikseli osi naprowadzających i punktów głównych. Cały obszar miał średnio 5,81% powierzchni oznaczonej jako osie naprowadzające oraz 0,26% oznaczonej jako punkty główne. Obszary słabiej skomponowane w obu kategoriach otrzymały niższe wyniki, podczas gdy obszary lepiej skomponowane otrzymały wyniki wyższe (Rys. 76). W kolejności „stopnia skomponowania” wyniki wyglądają następująco:

- Wisła i okolice: osie: 2,11%; punkty główne: 0,12%.
- Tory na południu Woli: osie: 4,68%; punkty główne: 0,22%.
- Średnia cały obszar: osie: 5,81%; punkty główne: 0,26%.
- Oś saska: osie: 7,12%; punkty główne: 0,23%.
- Założenie ujazdowskie: osie: 7,66%; punkty główne: 0,35%.

Metryki te umożliwiają automatyczne porównywanie ze sobą wybranych obszarów. Co więcej, wartości te są niezależne od powierzchni, więc możliwe jest zestawianie ze sobą obszarów o różnych rozmiarach.



Rys. 76. Liczba pikseli osi kompozycyjnych (czerwony) i liczba pikseli punktów głównych (czarny) rozpoznanych przez sieć dla wymienionych obszarów jako procent liczby pikseli całego obszaru. Opracowanie własne.

3.2.4. Wnioski z eksperymentu głównego

Opracowany w eksperymencie głównym algorytm stanowił rozwinięcie prac rozpoczętych w ramach eksperymentu wstępnego. Przygotowany prototyp zademonstrował możliwość wykorzystania głębokiego uczenia maszynowego do przetwarzania realnych, miejskich kontekstów przestrzennych. Ze względu na niszowość problemu konieczne było opracowanie autorskich metod ewaluacji jakościowej i ilościowej algorytmu. Stworzony ekstraktor głównych punktów kompozycyjnych w przeprowadzonych testach osiągnął skuteczność rzędu

59,4-67,2%. Mimo że pole percepcji narzędzia jest ograniczone do obszaru o powierzchni około 1 km², dzięki wykorzystaniu przetwarzania seryjnego możliwe jest przeprowadzanie analiz przestrzennych na obszarach większych. Dla seryjnie przetworzonych obszarów możliwe jest automatyczne wyliczenie danych statystycznych pozwalających na porównywanie ze sobą ich kompozycji przestrzennych. Opracowany prototyp pozwala na zautomatyzowane uczytelnianie kompozycji tkanki miejskiej i może być wykorzystany jako wsparcie w praktyce projektowej, planistycznej i administracyjnej. Może być także przydatny dla podmiotów i instytucji zainteresowanych komercyjną wartością reprezentacyjnych miejsc podkreślonych kompozycyjnie.

Przygotowane narzędzie wpisuje się w paradygmat projektowania hiperparametrycznego, w którym projektant i algorytm oparty na głębokim uczeniu maszynowym wspólnie odczytują konteksty przestrzenne. Interfejs narzędzia ma wysoki poziom abstrakcji, w którym operator wskazuje jedynie koordynaty interesującego go obszaru. Nie jest konieczne samodzielne, dedukcyjne oprogramowywanie parametrów punktów głównych i osi naprowadzających, ponieważ te zostały zdefiniowane indukcyjnie przez sieć neuronową na podstawie przykładów treningowych. Opracowane narzędzie umożliwia automatyzację procesu przetwarzania architektoniczno-urbanistycznych kompozycji przestrzennych, który do tej pory mógł być przeprowadzany jedynie metodą analizy eksperckiej.

Wyniki pokazują także, że w obecnej wersji opracowany prototyp ma pewne wady, a jego obsługa wymaga krytycznego myślenia. W zależności od scenariusza testowego od 32,8% do 40,6% oznaczonych punktów nie stanowiło punktu głównego ani równorzędnego punktu formalnie podkreślonego. W niektórych przypadkach oznaczone punkty były trudne w interpretacji. Decyzja sieci neuronowej nie jest oparta na zasadach wypracowanych dedukcyjnie, przez co nie można łatwo wskazać, jakie przesłanki kryły się za wskazaniem danego punktu. Zastanawiające jest także pominięcie przez algorytm pewnych oczywistych układów kompozycyjnych, takich jak oś saska, założenie wilanowskie czy plac Narutowicza. W obecnej formie algorytm zdecydowanie nie może być traktowany jako narzędzie zastępujące projektanta czy eksperta. Może za to służyć jako narzędzie zwiększające możliwości, przyspieszające i ułatwiające pracę oraz jako inspiracja do działań twórczych.

Większość problemów opracowanego narzędzia wynika z ograniczeń technicznych i czasowych przeprowadzonego eksperymentu. Jak na standardy komercyjnych narzędzi opartych na głębokim uczeniu maszynowym, wykorzystana sieć neuronowa była niewielka,

wytrenowana została na komercyjnym laptopie przy pomocy bardzo małego zbioru treningowego, a czas dostępny na strojenie hiperparametrów był krótki. Mimo to, osiągnięte rezultaty są obiecujące i stanowią zachętę do prowadzenia dalszych prac rozwojowych. W pierwszej kolejności zdecydowanie warto byłoby przy pomocy zespołu ekspertów przygotować o wiele obszerniejszy zbiór treningowy i wytrenować bardziej złożoną sieć na dedykowanym do tego sprzęcie lub przy pomocy rozwiązań chmurowych. W celu rozszerzenia zbioru treningowego można byłoby także rozwinąć rozpoczęte już prace nad generatorem syntetycznych kompozycji przestrzennych podporządkowanym punktem głównym. W trakcie prac przeprowadzanych nad eksperymentem na popularności zyskały nowsze modele sieci neuronowych oparte na transformerach. Wydają się one o wiele potężniejsze niż wykorzystana w eksperymencie generatywna sieć przeciwna. Przetestowanie najnowszych dostępnych algorytmów głębokiego uczenia maszynowego umożliwiłoby identyfikację najbardziej skutecznych rozwiązań. Selekcja taka mogłaby bazować na opracowanych na potrzeby niniejszej pracy metrykach służących do ewaluacji przygotowanego prototypu. Skuteczność sieci neuronowej prawdopodobnie poprawiłoby także rozszerzenie zakresu kontekstów przestrzennych reprezentowanych w przestrzeni cech. Ze względu na ograniczenia sprzętowe w obecnej formie rozdzielczość danych wejściowych jest niewielka, a konteksty ograniczają się do cech geometrycznych układu komunikacyjnego, zabudowy, topografii terenu i intensywności zieleni. Mapy cech mogłyby być rozszerzone o dodatkowe konteksty, w tym układ funkcjonalny, zdjęcia satelitarne, dane demograficzne, statystyczne i inne otwarte dane przestrzenne skorelowane z kompozycją przestrzenną.

4. Podsumowanie

4.1. Wnioski ogólne

Algorytmy głębokiego uczenia maszynowego są z powodzeniem stosowane w analizie i przetwarzaniu danych, generowaniu rozwiązań projektowych, klasyfikacji i uczytelnianiu danych, optymalizacji procesów, a nawet w przewidywaniu przyszłości na podstawie trendów historycznych. Szeroki katalog dostępnych narzędzi pozwala na skuteczne rozwiązywanie wielu kategorii problemów architektonicznych obejmujących przybliżanie wartości metryk i wyników symulacji, optymalizację parametrów, translację danych z jednej formy na drugą czy przetwarzanie obrazów i innych wielowymiarowych danych przestrzennych. Sieci neuronowe dobrze radzą sobie także jako algorytmy generatywne uczące się na zadanych

przykładach, algorytmy sterujące agentami w scenariuszach wymagających wykorzystania uczenia ze wzmacnianiem lub jako algorytmy służące przetwarzaniu danych sekwencyjnych, takich jak tekst naturalny czy ciągi czasowe.

Osiągnięte w eksperymentach wstępnym i głównym wyniki sugerują, że sieci neuronowe są zdolne do uczenia się zasad kompozycyjnych i do stosowania ich w praktyce. Zwłaszcza przetestowane splotowe sieci neuronowe mają duży potencjał do przetwarzania kontekstów przestrzennych. Algorytmy oparte na głębokim uczeniu maszynowym mogą wspierać architektów i urbanistów w analizie kompozycji przestrzennych, zmniejszając ryzyko przeoczenia istotnych kontekstów. Narzędzia zbliżone do opracowanego prototypu mogą skutecznie uczytelniać tkankę miejską oraz wspomagać podejmowanie decyzji projektowych i planistycznych. Wyniki badania pokazują, że sieci neuronowe nie są obecnie w stanie zastąpić architektów, ale mogą znacznie wspomagać i przyspieszać procesy projektowe i planistyczne w architekturze i urbanistyce. Przedstawiony prototyp stanowi duży krok w kierunku automatyzacji analiz kompozycji przestrzennych i otwiera nowe możliwości badawcze i praktyczne w tej dziedzinie.

4.2. Ograniczenia metody

Wyniki pokazują, że sieci neuronowe mogą być użyteczne w automatycznym przetwarzaniu danych przestrzennych, ale ich skuteczność zależy od wielu czynników, w tym jakości i ilości danych treningowych, rozmiaru i rodzaju wykorzystanej sieci i wyboru zastosowanych hiperparametrów. Ze względu na stochastyczny charakter głębokiego uczenia maszynowego i ograniczone możliwości sterowania hiperparametrycznego, obsługa sieci neuronowych zazwyczaj jest problematyczna. Algorytmy tego typu często wydają się być „czarnymi skrzynkami” (ang. black box), w których trudno jest przewidzieć, jaka decyzja zostanie podjęta, przez co wyniki są trudne do interpretacji. Duże wyzwanie stanowią także problemy związane z przesunięciem dystrybucji danych między zbiorem treningowym, a docelowymi danymi przetwarzanymi na etapie wdrożenia sieci w praktyce.

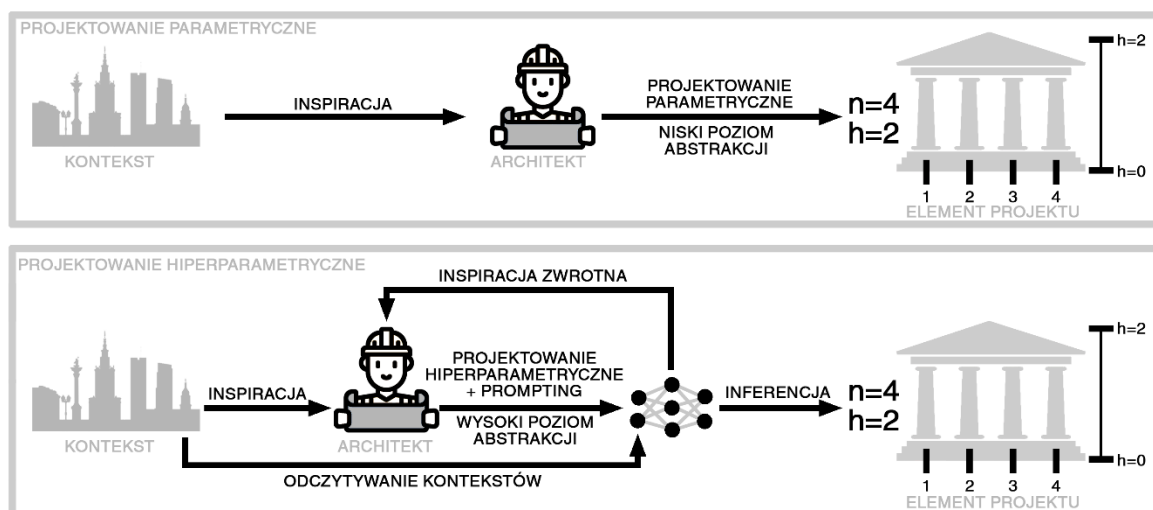
Napotkane na etapie prototypowania trudności wyznaczają jednak kierunek pod przyszłe badania, mogące obejmować rozszerzenie zbiorów treningowych i zastosowanie bardziej złożonych sieci neuronowych, testowanie nowszych modeli opartych na transformerach, czy integracja dodatkowych kontekstów o wysokiej rozdzielczości obejmujących układ funkcjonalny, zdjęcia satelitarne czy dane demograficzne. Konieczne wydaje

się też opracowanie nowszych metryk pozwalających na skuteczniejszą interpretację i ocenę otrzymywanych wyników.

4.3. Projektowanie hiperparametryczne – dyskusja

Wiele wydziałów architektury na uczelniach na całym świecie ma w programach swoich studiów elementy zaawansowanej obsługi komputerów, programowania i architektury parametrycznej. Warto wymienić między innymi ASK WAPW, ETH Zurih, MIT Media Lab, czy IAAC. Architekci są obecnie biegli w obsłudze narzędzi cyfrowych, a głębokie uczenie maszynowe staje się coraz bardziej dostępne także dla ludzi bez doświadczenia programistycznego. Na wysokim poziomie abstrakcji mamy dostęp do chatbotów LLM i intuicyjnych interfejsów online modeli generatywnych. Trochę większą kontrolę oferują platformy typu Dynamo lub Grasshopper [np. z wtyczką Pug dodającą szereg bloków, pozwalających na programowanie algorytmów uczenia maszynowego (Apellániz, 2023)]. Dzięki rozwiązaniom typu Google Colab i Jupyter Notebook możemy programować nawet złożone sieci neuronowe w prostym interfejsie przeglądarki internetowej. Największą kontrolę oferują biblioteki programistyczne, takie jak Tensorflow/Keras, PyTorch czy SciKitLearn. Wszystkie te kategorie rozwiązań są coraz częściej stosowane w architekturze, co można zauważyć między innymi w rosnącej liczbie publikacji dotyczących wykorzystania głębokiego uczenia maszynowego w CAAD.

Jesteśmy obecnie świadkami automatyzacji coraz bardziej złożonych procesów, które do tej pory uważane były za *stricte* ludzką domenę. Ten przeskok na wyższy poziom abstrakcji procesu projektowego wydaje się kontynuacją trwającego już kilkadziesiąt lat trendu. Najpierw rysunek odręczny przekształcił się w rysunek komputerowy. Rysunek CAD wyewoluował w projektowanie parametryczne. Z kolei ono zostaje obecnie stopniowo zastępowane przez projektowanie hiperparametryczne, w którym architekci definiują hiperparametry systemu uczenia maszynowego, a parametry rozwiązań problemów definiowane są już pośrednio przez samouczący się algorytm (Rys. 77). Kontrola nad hiperparametrami może mieć różnorakie formy. Od sterowania istniejącymi modelami i dostarczania im pojedynczych kontekstów wejściowych (promptów), przez dostrajanie dostępnych, wytrenowanych już modeli do indywidualnych potrzeb, aż po tworzenie własnych sieci neuronowych i własnych zbiorów danych treningowych.



Rys. 77. Diagram ilustrujący różnice między projektowaniem parametrycznym a projektowaniem hiperparametrycznym. W projektowaniu parametrycznym to architekt na podstawie odczytanego samodzielnie kontekstu decyduje o wartościach poszczególnych parametrów. Projektowanie hiperparametryczne charakteryzuje się wyższym poziomem abstrakcji. Architekt zainspirowany kontekstem definiuje hiperparametry modelu opartego na głębokim uczeniu maszynowym i dokonuje jego promptowania. Dodatkowo model także ma możliwość samodzielnego odczytywania kontekstów. Końcowe parametry definiowane są przez sieć neuronową na etapie inferencji. Wyniki inferencji mogą służyć architektowi jako inspiracja w przyszłych iteracjach. Opracowanie własne.

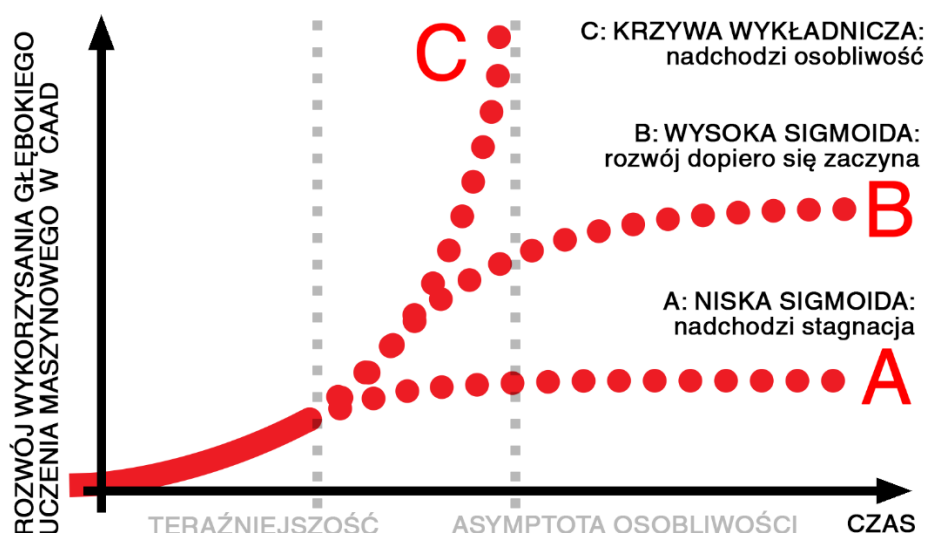
Wykorzystanie projektowania hiperparametrycznego opartego na sieciach neuronowych może być szczególnie korzystne w niektórych typach problemów architektonicznych. Głębokie uczenie maszynowe można aplikować tam, gdzie problem jest logicznie rozmyty i klasyczne metody ilościowe (np. oparte o dane importowane z systemów GIS) nie są możliwe do wykorzystania. W sytuacjach takich modele neuronowe mogą indukcyjnie nauczyć się podejmowania decyzji maksymalizujących oczekiwane korzyści. Sieci neuronowe dobrze radzą sobie w przetwarzaniu złożonych kontekstów, szczególnie tam, gdzie istnieje potrzeba równoległego lub sekwencyjnego przetworzenia wielu danych w krótkim czasie. Ze względu na źródła danych głębokie uczenie maszynowe warto wykorzystywać tam, gdzie dostępnych jest dużo danych typu big-data zgromadzonych w różnego rodzaju repozytoriach (np. bazach danych statystycznych, GIS, BIM). Obecnie istnieje wiele otwartych zbiorów danych nadających się do wykorzystania w CAAD. Wybór ciekawszych baz danych

architektoniczno-urbanistycznych, na które natknąłem się w trakcie przeprowadzonych badań, znajduje się w aneksie (Załącznik 8). Dane można zbierać także samodzielnie. Głębokie uczenie maszynowe może wspomagać przetwarzanie danych sensorycznych (np. z systemów smart city i IOT), ankietowych (np. będących wynikami procesów partycypacyjnych) czy symulacyjnych (przykładowo z symulacji energetycznych budynków). Gdy zbiory treningowe budowane są od zera, najlepiej tworzyć je w sposób jak najbardziej zautomatyzowany, ponieważ współczesne modele wymagają bardzo dużych zbiorów treningowych. Trening sieci można oprzeć także o systemy symulacyjne DRL, w których dane do treningu pochodzą z interakcji modelu ze środowiskiem symulacyjnym. Tam, gdzie nie da się zebrać wystarczająco dużo danych do wytrenowania własnej sieci, można próbować dostroić istniejące, wytrenowane już systemy do docelowego, niewielkiego zbioru treningowego. Można w takich przypadkach wykorzystać także technikę promptowania metodą wielokrotnej podpowiedzi (ang. few-shot prompting), polegającą na pokazaniu systemowi jedynie kilku przykładów danych wejściowych i wyjściowych. Few-shot prompting jest zazwyczaj stosowany w przypadku uniwersalnych asystentów opartych na dużych modelach językowych.

4.4. Okno na przyszłość

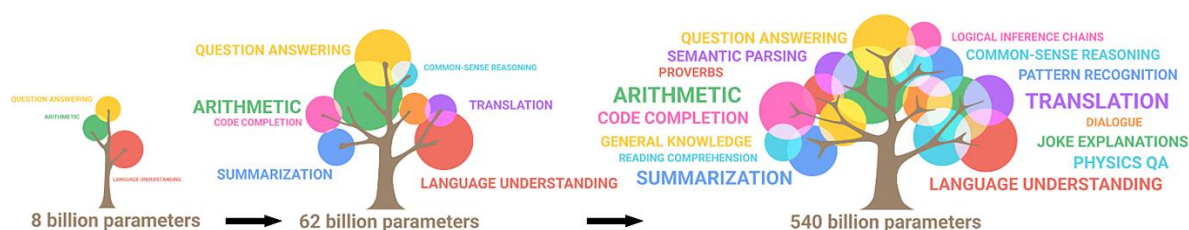
Patrząc na gwałtowny rozwój narzędzi bazujących na sieciach neuronowych w projektowaniu architektonicznym wspomaganym komputerowo, możemy zadać sobie pytanie: jaka jest przyszłość głębokiego uczenia maszynowego wykorzystywanego w architekturze? Odpowiedź na to pytanie niestety nie jest możliwa, jednak na podstawie stanu obecnego i trendów historycznych możemy wysnuć pewne oczekiwania. Rozwój technologii często przyrównywany jest do sigmoidy (Handy, 1994). Gdy nowa technologia staje się dostępna, najpierw wchodzi w fazę powolnego wzrostu, w której pierwsi użytkownicy z trudem wypracowują pierwotne metody jej wykorzystania. W pewnym momencie, gdy technologia staje się tania, ogólnodostępna i prosta w użytkowaniu, rozwój przechodzi w fazę wzrostu wykładniczego, w którym liczba użytkowników technologii rośnie lawinowo. Pojawiają się nowe metody implementacji, narzędzia, ludzie mają coraz więcej pomysłów na ich kreatywne wykorzystanie. Dzieje się tak aż do momentu dojścia do granicy tego, co jest możliwe do osiągnięcia przy istniejącej technologii. Innowacja spowalnia, liczba użytkowników się stabilizuje, a w końcu zaczyna stopniowo spadać. Czas powstawania niniejszej pracy (lata 2019-2024) zdecydowanie obejmuje moment, w którym wykorzystanie NN w CAAD

przeskoczyło w fazę wzrostu wykładniczego. Problem w tym, że nie jesteśmy w stanie rozpoznać, w którym miejscu krzywej się obecnie znajdujemy (Rys. 78). Czy jesteśmy już blisko poziomu nasycenia technologii, czy rozwój dopiero ruszył i czeka nas jeszcze wiele niespodzianek? Na dodatkową możliwość zwraca w swojej książce *The Singularity Is Near* Ray Kurzweil. Biorąc pod uwagę fakt, że bieżący postęp oparty jest na systemach samouczących się, nie możemy wykluczyć perspektywy, w której rozwój systemów AI wcale się nie ustabilizuje, a będzie ciągle przyspieszał i zacznie dążyć do nieskończoności (Kurzweil, 2005). Możliwość ta wydaje się dosyć mało prawdopodobna, jednak skoro nawet w już istniejących systemach LLM niektórzy doszukują się pierwszych sygnałów istnienia silnej sztucznej inteligencji AGI (ang. Artificial General Intelligence), nie możemy jej całkowicie zignorować (Bubeck, i in., 2023).



Rys. 78. Możliwe krzywe prognozujące rozwój wykorzystania głębokiego uczenia maszynowego w projektowaniu architektonicznym wspomagany komputerowo. Krzywa A pokazuje możliwość, że większość innowacji z wykorzystaniem NN w CAAD została już odkryta i zbliżamy się do możliwości technologicznych dalszego rozwoju technologii. Krzywa B dopuszcza możliwość, że rozwój dopiero się zaczął i pojawi się jeszcze wiele nowych narzędzi. Krzywa C opisuje sytuację, w której rozwój systemów AI doprowadzi do pojawienia się samouczącej się AGI, prowadzącej do wykładniczego rozwoju i transformacji całej ludzkości. Opracowanie własne na podstawie książek *The Empty Raincoat* (Handy, 1994) oraz *The Singularity is Near* (Kurzweil, 2005).

Przeprowadzone w niniejszej pracy badanie stanu wiedzy sugeruje, że popularność prostych implementacji sieci neuronowych służących jako narzędzia pomocnicze w rozwiązywaniu szerokiego spektrum problemów architektonicznych stopniowo rośnie. Obecnie poza ogólnie używanymi LLM i modelami dyfuzyjnymi, wykorzystywane w architekturze sieci neuronowe są dosyć niewielkie. Wydaje się jednak, że ze względu na duży potencjał ekonomiczny dyscypliny, dedykowane architekturze sieci neuronowe prawdopodobnie będą stawały się coraz większe²⁸. Jak pokazuje ewolucja dużych modeli głębokiego uczenia maszynowego, takich jak GPT 1-2-3-4... lub Dall-E 1-2-3..., im większy jest model, tym większe są jego zdolności generalizacyjne, a więc można go wykorzystać do większej liczby zastosowań (Rys. 79).



Rys. 79. Wizualizacja przygotowana przez Google, ilustrująca wzrost zdolności dużych modeli językowych wraz ze wzrostem ich wielkości na przykładzie modelu PALM. Im większa jest sieć neuronowa tym większe stają się jej kompetencje i tym bardziej zbliża się do poziomu AGI (Narang S i Chowdhery, 2022).

Przyszłość wydaje się należeć do dużych modeli, które stopniowo będą zbliżać się do poziomu silnej sztucznej inteligencji AGI. Prawdopodobnie modele takie nie będą początkowo bezpośrednio przeznaczone do zadań architektonicznych. Architektura będzie stanowiła zadanie dodatkowe, które z biegiem czasu i wzrostem zainteresowania branży będzie coraz bardziej rozwijane. Pierwsze sygnały takiego rozwoju rzeczy widzimy już obecnie, gdy ogólne modele językowe, takie jak GPT4, są aplikowane do coraz bardziej niszowych problemów.

²⁸ Przykładowo w 2017 roku firma konsultingowa McKinsey & Company oszacowała światową wartość roczną sektora konstrukcji na 10 bilionów dolarów amerykańskich z potencjałem wzrostu o dodatkowe 1,6 biliona (McKinsey Global Institute, 2017); według raportu IEA z 2019 roku sektor budowlany odpowiada za 36% światowego zapotrzebowania na energię końcową i 36% światowej emisji CO₂ (GlobalABC International Energy Agency, 2019).

W obliczu dynamicznie rozwijającego się paradygmatu projektowania hiperparametrycznego oraz rosnącego znaczenia sieci neuronowych w projektowaniu wspomaganym komputerowo, przyszłość architektury i urbanistyki jawi się wyjątkowo ciekawie. Perspektywy automatyzacji złożonych procesów (takich jak przetwarzania kontekstów kompozycyjnych) otwiera nowe możliwości usprawnienia i zwiększenia wartości pracy architektów i urbanistów. Na razie wydaje się, że narzędzia te nie zastąpią ludzkiej kreatywności, możliwości krytycznej oceny wyników i integracji złożonych procesów, ale mogą stanowić istotne wsparcie przyspieszające procesy projektowe i minimalizujące ryzyko popełnienia błędów. Wraz z dalszym rozwojem technologii i większym dostępem do danych i metod ich przetwarzania, możemy oczekiwać, że architektura stanie się jeszcze bardziej interdyscyplinarna i będzie coraz intensywniej korzystać z osiągnięć dziedziny głębokiego uczenia maszynowego. Rozpoznane w niniejszej pracy trendy rozwoju paradygmatu projektowania hiperparametrycznego i przedstawiony prototyp podkreślają ogromny potencjał architektury jako dziedziny praktycznej, badawczej i artystycznej, oraz wskazują na jej ogromny potencjał rozwojowy.

Bibliografia

- Abidoye, R. B. i Chan, A. P. C., 2017. Modeling property values in Nigeria using artificial neural network. *Journal of Property Research*, 34(1), s. 36-53.
- Achmatowicz-Otok, A. i Jarosz, A., 1996. The analysis of the land prices in Warsaw. *Miscellanea Geographica. Regional Studies on Development*, Tom 7, s. 173-180.
- Alani, M. W. i Al-Kaseem, B. R., 2021. Fill In The Blanks: Deep Convolutional Generative Adversarial Network to Investigate the Virtual Design Space of Historical Islamic Patterns. *9th ASCAAD Conference Proceedings*, s. 614-621.
- Alawadhi, M. i Yan, W., 2020. BIM Hyperreality: Data Synthesis Using BIM and Hyperrealistic Rendering for Deep Learning. *ACADIA*, s. 228-236.
- Alembics, 2021. *Disco Diffusion*. [Online]
Dostęp: <https://github.com/alembics/disco-diffusion/tree/main>
[Data uzyskania dostępu: 16 01 2024].
- Alexander, C., Ishikawa, S., Fiksdahl-King, I. i Angel, S., 1977. *A Pattern Language*. New York: Oxford University Press.
- Algeciras-Rodriguez, J., 2018. Stochastic Hybrids: From references to design options through Self Organizing Maps methodology. *eCAADe*, 1(36), s. 119-128.
- Aman, J., Matisziw, T. C., Kim, J. B. i Luo, D., 2022. Sensing The City: Leveraging geotagged social media posts and street view imagery to model urban streetscapes using deep neural networks. *Proceedings of the 27th CAADRIA Conference*, s. 595-604.
- Apellániz, D., 2023. *Pug*. [Online]
Dostęp: <https://www.food4rhino.com/en/app/pug>
[Data uzyskania dostępu: 18 01 2024].
- As, I., Pal, S. i Basu, P., 2018. Artificial intelligence in architecture: Generating conceptual design via deep learning. *International Journal of Architectural Computing*, 16(4), s. 306-327.
- Asl, M. R., Das, S., Tsai, B., Molloy, I. i Hauck, A., 2017. Energy Model Machine (EMM): Instant Building Energy Prediction using Machine Learning. *eCAADe*, 2(35), s. 277-286.
- Aubert, J. B. i Tardieu, P. F., 1788. *Varsovie superficie 1,900,000 toises*. [Sztuka] (BnF Gallieu).

- Beniaguev, D., Segev, I. i London, M., 2021. Single cortical neurons as deep artificial neural. *Neuron*, 109(17), s. 2727-2739.
- Black, W. R., 1995. Spatial interaction modeling using artificial neural networks. *Journal of Transport Geography*, 3(3), s. 159-166.
- Boulila, W., Ghandorh, H., Khan, M. A., Ahmed, F. i Ahmad, J., 2021. A novel CNN-LSTM-based approach to predict urban expansion. *Ecological Informatics*.
- Brennan, T., 1948. *Midland City: Wolverhampton-Social and Industrial Survey*. London: Dennis Dobson Ltd.
- Brown, L., Yip, M., Gardner, N., Haeusler, M. H., Khean, N., Zavoleas, Y. i Ramos, C., 2020. Drawing Recognition: Integrating Machine Learning Systems into Architectural Design Workflows. *eCAADe*, 2(38), s. 289-298.
- Bubeck, S., Chandrasekaran, V., Eldan, R., Gehrke, J., Horvitz, E., Kamar, E. i inni, 2023. Sparks of Artificial General Intelligence: Early experiments with GPT-4. *arXiv*.
- The Terminator*. 1984. [Film] Reżyser: J Cameron. USA: New World Pictures, Orion Pictures.
- Carpó, M., 2017. *The Second Digital Turn: Design Beyond Intelligence*. Cambridge: MIT Press.
- Chan, Y. H. E. i Spaeth, A. B., 2020. Architectural Visualisation with Conditional Generative Adversarial Networks (cGAN): What machines read in architectural sketches. *eCAADe*, 2(38), s. 299-308.
- Chatzikonstantinou, I. i Sariyildiz, S., 2016. Approximation of simulation-derived visual comfort indicators in office spaces: a comparative study in machine learning. *Architectural Science Review*, 59(4), s. 307-322.
- Chen, D., Luo, D., Xu, W., Luo, C., Shen, L., Yan, X. i Wang, T., 2019. Re-perceive 3D printing with Artificial Intelligence. *eCAADe*, 1(37), s. 443-450.
- Chen, M., Radford, A., Child, R., Wu, J., Jun, H., Luan, D. i Sutskever, I., 2020. Generative Pretraining From Pixels. *Proceedings of the 37th International Conference on Machine Learning, PMLR*, Tom 119, s. 1691-1703.
- Ching, F., 2007. *Architecture Form, Space, and Order, 3rd Edition*. Hoboken, New Jersey: John Willey & Sons.
- Cho, K., Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H. i Bengio, Y., 2014. Learning Phrase Representations using RNN Encoder–Decoder for

- Statistical Machine Translation. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, s. 1724–1734.
- Çolakoğlu, M. B. i Uzun, C., 2019. Architectural Drawing Recognition: A case study for training the learning algorithm with architectural plan and section drawing images. *eCAADe*, 2(37), s. 29-34.
- Creswell, J. W., 2013. *Projektowanie badań naukowych: Metody jakościowe, ilościowe i mieszane*. I red. Kraków: Wydawnictwo Uniwersytetu Jagiellońskiego.
- Croce, V., Caroti, G., De Luca, L., Piemonte, A., Veron, P. i Bevilacqua, M. G., 2021. Artificial Intelligence and H-BIM for the Semantic Description of Cultural Heritage: the Pisa Charterhouse. *42° Convegno Internazionale dei Docenti delle Discipline della Rappresentazione Congresso della Unione Italiana per il Disegno*, s. 608-625.
- Cudzik, J. i Radziszewski, K., 2018. Artificial Intelligence Aided Architectural Design. *eCAADe*, 1(36), s. 77-84.
- David, J. i Leita, A., 2022. Getting a Handle on Floor Plan Analysis. *10th International Conference of the Arab Society for Computation in Architecture, Art and Design (ASCAAD 2022 Proceedings)*, s. 221-236.
- de Miguel, J., Villafane, M. E., Piskorec, L. i Sancho-Caparrini, F., 2019. Deep Form Finding: Using Variational Autoencoders for deep form finding of structural typologies. *eCAADe*, 1(23), s. 71-80.
- del Campo, M., Carlson, A. i Manninger, S., 2020. Towards Hallucinating Machines - Designing with Computational Vision. *International Journal of Architectural Computing*, 19(1), s. 88-103.
- del Campo, M., Carlson, A. i Manninger, S., 2019. Imaginary Plans: The potential of 2D to 2D Style transfer in planning processes. *ACADIA*, s. 412-418.
- del Campo, M., Carlson, A. i Manninger, S., 2020. How Machines Learn to Plan: A Critical Interrogation of Machine Vision Techniques in Architecture. *ACADIA*, s. 272-281.
- del Campo, M., Carlson, A. i Manninger, S., 2020. 3D Graph Convolutional Neural Networks in Architecture Design. *ACADIA*, s. 688-696.
- del Castillo y López, J. L. G., 2019. Robot Ex Machina A Framework for Real Time Robot Programming and Control. *ACADIA 2019*, s. 40-49.
- Devadass, P., Heimig, T., Stumm, S., Kerber, E. i Brell-Cokcan, S., 2019. Robotic Constraints Informed Design Process. *ACADIA*, s. 130-139.

- Dias, J. L., Silva, A., Chai, C. i Gaspar, P. L. d. B. J., 2013. Neural networks applied to service life prediction of exterior painted surfaces. *Building Research &*, 42(3), s. 371-380.
- Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T. i inni, 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *International Conference on Learning Representations*.
- Duany Plater-Zyberk & Company, 2003. *SmartCode*. brak miejsca:brak nazwiska
- Duany, A. i Talen, E., 2002. Transect Planning. *Journal of the American Planning Association*, 68(3), s. 245-266.
- Dzieduszyński, T., 2022b. Convolutional Neural Networks as Context-Scraping Tools in Architecture and Urban Planning. *Builder*, 296(3), s. 79-81.
- Dzieduszyński, T., 2022a. Machine learning and complex compositional principles in architecture: Application of convolutional neural networks for generation of context-dependent spatial compositions. *International Journal of Architectural Computing*, 20(2), s. 196-215.
- Dzieduszyński, T., Gizelska, J., Marek, W. i Tan, V., 2020. *Próby generowania rozwiązań architektoniczno-urbanistycznych przy pomocy algorytmu PatchMatch. Silne i słabe strony uzyskanych rezultatów*. Kraków, CreativeTime.
- Eisenstadt, V., Arora, H., Ziegler, C., Bielski, J., Langenhan, C., Althoff, K. D. i Dengel, A., 2021. Comparative Evaluation of Tensor based Data Representations for Deep Learning Methods in Architecture. *eCAADe*, 1(39), s. 45-54.
- Eisenstadt, V., Langenhan, C. i Althoff, K. D., 2019. Generation of Floor Plan Variations with Convolutional Neural Networks and Case-based Reasoning: An approach for transformative adaptation of room configurations within a framework for support of early conceptual design phases. *eCAADe*, 2(23), s. 79-84.
- Fang, Z., Wu, Y., Hassonjee, A., Bidgoli, A. i Cardoso-Llach, D., 2020. Towards a Distributed Robotically Assisted Construction Framework: Using Reinforcement Learning to Support Scalable Multi-Drone Construction in Dynamic Environments. *ACADIA*, s. 320-329.
- Feyerabend, P. K., 2001. *Przeciw Metodzie*. Wrocław: Wydawnictwo Siedmioróg.
- Fischer, M. M. i Gopal, S., 1994. Artificial neural networks: A new approach to modeling interregional telecommunication flows. *Journal of Regional Science*, 34(4), s. 503-527.

- Freud, S., 1950/1895. Entwurf einer Psychologie. W: *Aus den anfangen der psychoanalyse. briefe an Wilhelm Fliess, abhandlungen und notizen aus de jahren 1887-1902..*
London: Williams, Lea & Co, s. 371-469.
- Fukuda, T., Kuwamuro, Y. i Yabuki, N., 2017. Optical Integrity of Diminished Reality Using Deep Learning. *eCAADe*, 1(35), s. 241-250.
- Fukushima, K., 1980. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, Tom 36, s. 193-202..
- Fuller, R. B., 1962. *Tensile-integrity structures*. The United States Patent and Trademark Office, Patent nr US 3063521.
- Fuller, R. B., 1961. *Aspension (Suspension Building)*. The United States Patent and Trademark Office, Patent nr US 3139957.
- Gao, X. i Pishdad-Bozorgi, P., 2019. A framework of developing machine learning models for facility life-cycle cost analysis. *Building Research & Information*, 48(5), s. 501-525.
- Gatys, L., Ecker, A. i Bethge, M., 2016. A Neural Algorithm of Artistic Style. *Journal of Vision*, 16(326).
- Gehl, J., 1971. *Life Between Buildings*. VI, 2010 red. Copenhagen: Danish Architectural Press.
- Gehl, J., 2014. *Miasta dla Ludzi*. Kraków: Wydawnictwo RAM.
- Giannopoulou, E., Baquero, P., Warang, A., Orciuoli, A. i Estevez, A. T., 2019. Stripe Segmentation for Branching Shell Structures: A Data Set Development as a Learning Process for Fabrication Efficiency and Structural Performance. *eCAADe*, 3(37), s. 63-70.
- GlobalABC International Energy Agency, 2019. *2019 Global Status Report for Buildings and Construction*, brak miejsca: United Nations Environment Programme.
- Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S. i inni, 2014. Generative Adversarial Nets. *Advances in Neural Information Processing Systems (NIPS)*, 27.
- Google, Gemini Team, 2023. Gemini: A Family of Highly Capable Multimodal Models. *arXiv*.
- Google, 2023. *Google Maps*, brak miejsca: Google.

- Gumin, M., 2016. *Wave Function Collapse Algorithm*. [Online]
Dostęp: <https://github.com/mxgmn/WaveFunctionCollapse>
[Data uzyskania dostępu: 11 04 2022].
- Guo, J., Liu, Y., Yang, Q. K. i Wang, Y., 2019. GPS-based citywide traffic congestion forecasting using CNN-RNN and C3D hybrid model. *Transportmetrica A: Transport Science*, 17(2), s. 190-211.
- Guranowska-Gruszecka, K., 2018. Zabudowa wysoka Warszawy. W: K. Guranowska-Gruszecka i M. Guranowska-Gruszecka, redaktorzy *Węzły Miasta*. Warszawa: Fundacja Wydziału Architektury Politechniki Warszawskiej, s. 101-129.
- Guranowska-Gruszecka, K. i Łaskarzewska, M., 2018. Słowo wstępne. W: K. Guranowska-Gruszecka i M. Łaskarzewska, redaktorzy *Węzły miasta*. Warszawa: Fundacja Wydziału Architektury Politechniki Warszawskiej, s. 6-13.
- Gzell, S., 2013. Praktyka planistyczna a naukowe metody badania miasta. W: P. Lorens i I. Mironowicz, redaktorzy *Wybrane teorie współczesnej urbanistyki*. Gdańsk: Akapit-DTP, s. 224-245.
- Handy, C. B., 1994. *The Empty Raincoat: Making Sense Of The Future*. London: Hutchinson.
- Han, Z., Yan, W. i Liu, G., 2020. A Performance-Based Urban Block Generative Design Using Deep Reinforcement Learning and Computer Vision. *CDRF 2020, Proceedings of the 2020 DigitalFUTURES*, s. 134-143.
- Harding, J., 2016. Dimensionality Reduction for Parametric Design Exploration. *Advances in Architectural Geometry*.
- Hawkins, J. K., 1961. Self-Organizing Systems-A Review and Commentary. *Proceedings of the IRE*, 49(1), s. 31-48.
- Hay, J. C., Murray, A. E. i Rosenblatt, F., 1960. *Mark I Perceptron Operators' Manual (Project PARA)*. Buffalo: Cornell Aeronautical Laboratory.
- Heartney, E., 2013. *Kenneth Snelson: Art and Ideas*. brak miejsca: Kenneth Snelson.
- He, K., Zhang, X., Ren, S. i Sun, J., 2016. Deep Residual Learning for Image Recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, s. 770-778.
- Herman, K., 2015. Zieleń – element niezbędny w tworzeniu centrum lokalnego jako przestrzeni wzmoczonych integracji międzyludzkich. W: *Studium koncepcyjne dotyczące centrów lokalnych w Warszawie*. Warszawa: OW SARP, s. 135-137.

- Heschong, L., Wymelenberg, V. D., Keven, Andersen, M., Digert, N., Fernandes, L. i inni, 2012. *Approved Method: IES Spatial Daylight Autonomy (sDA) and Annual Sunlight Exposure (ASE)*. New York: Illuminating Engineering Society.
- Hillier, B., 2007. *Space is the machine: A configurational theory of architecture*. Londyn: Space Syntax.
- Hinton, G. E. i Ng, A., 2017. *Heroes of Deep Learning: Andrew Ng interviews Geoffrey Hinton*, *deeplearning.ai*. [Online]
Dostęp: <https://youtu.be/-eyhCTvrEtE>
[Data uzyskania dostępu: 15 02 2022].
- Hinton, G. E., Rumelhart, D. E. i Williams, R. J., 1986. Learning Representations by Back-Propagating Errors. *Nature*, 323(9), s. 533-536.
- Hinton, G. i van der Mateen, L., 2008. Visualizing data using t-SNE. *Journal of Machine Learning Research*, Tom 9, s. 2579-2605.
- Ho, J., Jain, A. i Abbeel, P., 2020. Denoising Diffusion Probabilistic Models. *arXiv*.
- Ho, B., 2020. Making a New City Image. *ACADIA*, s. 658-667.
- Hochreiter, S. i Schmidhuber, J., 1997. Long Short-term Memory. *Neural Computation*, 9(8), s. 1735-1780.
- Hosmer, T. i Tigas, P., 2019. Deep Reinforcement Learning for Autonomous Robotic Tensegrity (ART). *ACADIA*, s. 16-29.
- Hosmer, T., Tigas, P., Reeves, D. i He, Z., 2020. Spatial Assembly with Self Play Reinforcement Learning. *ACADIA*, s. 382-393.
- Hu, K., 2023. *ChatGPT sets record for fastest-growing user base - analyst note*. [Online]
Dostęp: <https://www.reuters.com/technology/chatgpt-sets-record-fastest-growing-user-base-analyst-note-2023-02-01/>
[Data uzyskania dostępu: 07 01 2024].
- Hurley, S., 2008. *AutoCAD Release History: AutoCAD 2.0 (5) October 1984*. [Online]
Dostęp: https://autodesk.blogs.com/between_the_lines/ACAD_R5.html
[Data uzyskania dostępu: 02 02 2024].
- Iino, S., Ito, R., Doi, K., Imaizumi, T. i Hikosaka, S., 2018. CNN-based generation of high-accuracy urban distribution maps utilising SAR satellite imagery for short-term change monitoring. *International Journal of Image and Data FUion*, 9(4), s. 302-318.

- Ikeno, K., Fukuda, T. i Yabuki, N., 2020. Automatic Generation of Horizontal Building Mask Images by Using a 3D Model with Aerial Photographs for Deep Learning. *eCAADe*, 2(38), s. 271-278.
- Isola, P., Zhu, J.-Y., Zhou, T. i Efros, A., 2017. Image-to-Image Translation with Conditional Adversarial Networks. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, s. 5967-5976.
- Jachimowicz, A. M., 2023. *Cyfrowe metody wspomagania analiz struktury przestrzennej miast na przykładzie Warszawy*. Warszawa: Wydział Architektury Politechniki Warszawskiej.
- Jachimowicz, A. M., 2022. Morphology of Warsaw City Structure Using Urban Indexes and GIS Tools. *Land*, 11(135).
- Jeziorska, A., 2018. Węzły miejskości w aspekcie życia miejskiego i jego atrakcyjności ekonomiczno-kulturalnej. W: K. Guranowska-Gruszecka i M. Łaskarzewska, redaktorzy *Węzły Miasta*. Warszawa: Fundacja Wydziału Architektury Politechniki Warszawskiej, s. 36-51.
- Jørgensen, J., Tamke, M. i Poulsgaard, K. S., 2020. Occupancy-informed: Introducing a method or flexible behavioural mapping in architecture using machine vision. *eCAADe*, Tom 2, s. 251-258.
- Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O. i inni, 2021. Highly accurate protein structure prediction with AlphaFold. *Nature*, 596, s. 583–589.
- Kampani, A. i Varoudis, T., 2020. Perceptive Machine Visuospatial Configurations Through Machine Intuition. *eCAADe*, 38, s. 419-428.
- Karpathy, A., 2014. *ConvNetJS Deep Learning in your browser*. [Online]
Dostęp: <https://cs.stanford.edu/people/karpathy/convnetjs/demo/classify2d.html>
[Data uzyskania dostępu: 17 02 2022].
- Karras, T., Laine, S. i Aila, T., 2021. A Style-Based Generator Architecture for Generative Adversarial Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(12), s. 4217-4228.
- Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J. i Aila, T., 2020. Analyzing and Improving the Image Quality of StyleGAN. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, s. 8107-8116.
- Kartografii, G. U. G. i., 2023. *Baza Danych Obiektów Topograficznych (BDOT10k)*, Warszawa: Geoportal.gov.pl.

- Kauko, T., Hooimeijer, P. i Hakfoort, J., 2002. Capturing Housing Market Segmentation: An Alternative Approach based on Neural Network Modelling. *Housing Studies*, 17(6), s. 875-894.
- Kikuchi, T., Fukuda, T. i Yabuki, N., 2021. Automatic Diminished Reality Based Virtual Demolition Method using Semantic Segmentation and Generative Adversarial Network for Landscape Assessment. *eCAADe*, 2(39), s. 529-538.
- Kim, E. i Rosenwasser, D., 2020. Urban Emotion The interrogation of social media and its implications within urban context. *eCAADe*, 38, s. 475-482.
- Kindangen, J. I., 1996. Artificial neural networks and naturally ventilated buildings. *Building Research & Information*, 24(4), s. 203-208.
- Kinugawa, H. i Takizawa, A., 2019. Deep Learning Model for Predicting Preference of Space by Estimating the Depth Information of Space using Omnidirectional Images. *eCAADe*, 2(37), s. 61-68.
- Kirova, N. i Markopoulou, A., 2020. Pedestrian Flow: Monitoring and Prediction: Monitoring and prediction of pedestrian flows through smart material sensing surfaces towards informed dynamic mobility reconfiguration strategies. *ACADIA*, s. 84-93.
- Koshelyuk, D., Talaei, A., Garivani, S., Markopoulou, A., Chronis, A., Leon, D. A. i Krenmuller, R., 2019. Alive: A Multi Layered Flexible and Elastic Shape Aware Graphene Based Interface. *ACADIA*, s. 664-673.
- Krizhevsky, A., Hinton, G. i Sutskever, I., 2012. ImageNet Classification with Deep Convolutional Neural Networks. *Advances in Neural Information Processing Systems*, 25(2).
- 2001: *A Space Odyssey*. 1968. [Film] Reżyser: S Kubrick. UK, USA: Stanley Kubrick Production, Metro Goldwyn Mayer.
- Kurzweil, R., 2005. *The Singularity Is Near: When Humans Transcend Biology*. New York: Penguin Books.
- Lam, K. C., Yu, C. Y. i Lam, K. Y., 2008. An Artificial Neural Network and Entropy Model for Residential Property Price Forecasting in Hong Kong. *Journal of Property Research*, 25(4), s. 321-342.
- Law, S., Seresinhe, C. I., Shen, Y. i Gutierrez-Roig, M., 2018. Street-Frontage-Net: urban image classification using deep convolutional neural networks. *International Journal of Geographical Information Science*, 34(4), s. 681-707.

- Le Conseil de Paris, 2023. *Plan Local d'Urbanisme de Paris, Dispositions Generales Applicables Au Territoire Couvert Par Le PLU*. 55 red. brak miejsca:brak nazwiska
- Le Conseil de Paris, 2023. *Plan Local d'Urbanisme de Paris, Troisième partie : Les choix retenus pour établir*. 55 red. Paris: brak nazwiska
- Le Conseil de Paris, 2015. *Plan Local d'Urbanisme, Atlas général - Planches thématiques: Fuseaux de protection du site de Paris*. 35 red. Paris: brak nazwiska
- LeCun, Y., Boser, B., Denker, J., Henderson, D., Howard, R., Hubbard, W. i Jackel, L., 1989. Backpropagation Applied to Handwritten Zip Code Recognition. *Neural Computation*, 1(4), s. 541-551.
- LeCun, Y. i Fridman, L., 2022. *Yann LeCun: Dark Matter of Intelligence and Self-Supervised Learning | Lex Fridman Podcast #258*. [Online]
Dostęp: <https://youtu.be/SGzMELJ11Cc>
[Data uzyskania dostępu: 15 01 2022].
- Li, C. i Wand, M., 2016. Precomputed Real-Time Texture Synthesis with Markovian Generative Adversarial Networks. *Computer Vision - ECCV 2016*, Tom 9907, s. 702-716.
- Li, J. i Chen, H., 2021. Optimization and Prediction of Design Variables Driven by Building Energy Performance A Case Study of Office Building in Wuhan. *CDRF 2020: Proceedings of the 2020 DigitalFUTURES*, s. 229-242.
- Lillicrap, T., Hunt, J., Pritzel, A., Heess, N., Erez, T., Tassa, Y. i inni, 2015. Continuous control with deep reinforcement learning. *arXiv*.
- Li, P. i Zhu, W., 2020. Clustering and Morphological Analysis of Campus Context Based on a Convolutional Autoencoder Model. *ACADIA*, s. 170-177.
- Liu, C., Shen, J., Ren, Y. i Zheng, H., 2020. Pipes of AI - Machine Learning Assisted 3D Modeling Design. *The International Conference on Computational Design and Robotic Fabrication, Proceedings of the 2020 DigitalFUTURES*, s. 17-26.
- Liu, H., Liao, L. i Srivastava, A., 2019. An Anonymous Composition: Design Optimization Through Machine Learning Algorithm. *ACADIA*, s. 404-411.
- Liu, X. i Lathrop Jr, R. G., 2002. Urban change detection based on artificial neural network. *International Journal of Remote Sensing*, 23(11), s. 2513-2518.
- Liu, Y., Luo, Y., Deng, Q. i Zhou, X., 2020. Exploration of Campus Layout Based on Generative Adversarial Network Discussing the Significance of Small Amount

- Sample Learning for Architecture. *CDRF 2020: Proceedings of the 2020 DigitalFUTURES*, s. 169–178.
- Liu, Z., Mao, H., Wu, C., Feichtenhofer, C., Darrell, T. i Xie, S., 2022. A ConvNet for the 2020s. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, s. 11976-11986.
- Li, Y., Yabuki, N., Fukuda, T. i Zhang, J., 2020. A big data evaluation of urban street walkability using deep learning and environmental sensors. *eCAADe*, 2(38), s. 319-328.
- Long, J., Shelhamer, E. i Darrell, T., 2015. Fully convolutional networks for semantic segmentation. *IEEE CVPR*, Tom 1, s. 3431-3440.
- Luo, D., Wang, J. i Xu, W., 2018. Applied Automatic Machine Learning Process for Material Computation. *eCAADe*, 1(36), s. 109-118.
- Lynch, K., 1960. *The Image of the City*. Cambridge: MIT Press.
- Lyu, Y., Vosselman, G., Xia, G. S., Yilmaz, A. i Yang, M. Y., 2018. UAVid: A Semantic Segmentation Dataset for UAV Imagery. *arXiv*.
- Mahankali, R., Johnson, B. R. i Anderson, A. T., 2018. Deep learning in design workflows The elusive design pixel. *International Journal of Architectural Computing*, 16(4), s. 328-340.
- Maithani, S., Arora, M. K. i Jain, R. K., 2010. An artificial neural network based approach for urban growth zonation in Dehradun city, India. *Geocarto International*, 25(8), s. 663-681.
- Małkiński, M. i Mańdziuk, J., 2022. Deep Learning Methods for Abstract Visual Reasoning: A Survey on Raven's Progressive Matrices. *arXiv*.
- Mapbox, 2023. *Mapbox Studio*. [Online]
Dostęp: <https://www.mapbox.com/about/maps/>
[Data uzyskania dostępu: 19 02 2024].
- Markusiewicz, J. i Balerdi, A. G., 2020. LOTI Using Machine Learning to simulate subjective opinions in design. *eCAADe*, 1(38), s. 439-448.
- Marsillo, L., Suntorachai, N., Karthikeyan, K., Voinova, N., Khairallah, L. i Chronis, A., 2022. Context Decoder: Measuring urban quality through artificial intelligence. *Proceedings of the 40th Conference on Education and Research in Computer Aided Architectural Design in Europe (eCAADe 2022)*, s. 237-246.

- Maryńczuk, P., 2012. Composition of a Plan of an Architectural Complex. *Civil And Environmental Engineering Reports*, 9, s. 63-72.
- Masad, O. i Shamir, A., 2017. *Automatic Image Composition*. USA, Patent nr US20170208243A1.
- Mazuroski, W., Berger, J., Oliveira, R. C. L. F. i Mendes, N., 2018. An artificial intelligence-based method to efficiently bring CFD to building simulation. *Journal of Building Performance Simulation*, 11(5), s. 588-603.
- McCarley, R., 1998. Dreams: Disguise of Forbidden Wishes or Transparent Reflections of a Distinct Brain State?. *Annals of the New York Academy of Sciences*, 846, s. 116-133.
- McCulloch, W. S. i Pitts, W., 1943. A Logical Calculus of the Ideas Immanent in Nervous Activity. *Bulletin of Mathematical Biophysics*, Tom 5, s. 115-133.
- McKinsey Global Institute, 2017. *Reinventing Construction: A Route To Higher Productivity*, brak miejsca: McKinsey & Company.
- Meeran, A. i Joyce, S. C., 2020. Machine Learning for Comparative Urban Planning at Scale: An Aviation Case Study. *ACADIA 2020: Distributed Proximities*, Tom 1, s. 178-187.
- Mehmood, H. i Tripathi, N. K., 2013. Optimizing artificial neural network-based indoor positioning system using genetic algorithm. *International Journal of Digital Earth*, 6(2), s. 158-184.
- Meta AI, 2023. Llama 2: Open Foundation and Fine-Tuned Chat Models. *arXiv*.
- Minsky, M. i Papert, S., 1969. *Perceptrons An Introduction to Computational Geometry*. Cambridge: MIT Press.
- Minsky, M. i Sykes, C., 2011. *Marvin Minsky - Show and tell: My neural network machine (137/151)*. [Online]
Dostęp: <https://youtu.be/waYt6fGyayE>
[Data uzyskania dostępu: 02 02 2024].
- Minsky, M. i Sykes, C., 2017. *Marvin Minsky - How I became interested in neural networks (22/151)*. [Online]
Dostęp: <https://youtu.be/2jgji5LMDzQ>
[Data uzyskania dostępu: 02 02 2024].
- Mintrone, A. i Erioli, A., 2021. Training Spaces: Fostering machine sensibility for spatial assemblages through wave function collapse and reinforcement learning. *eCAADe*, 1(39), s. 17-39.

- Mironowicz, I., 2013. Miasto, jego struktura i kompozycja - Definicje, schematy, relacje przestrzenne. W: P. Lorens i I. Mironowicz, redaktorzy *Wybrane teorie współczesnej urbanistyki*. Gdańsk: Akapit-DTP, s. 93-116.
- Mironowicz, I. i Ossowicz, T., 1997. Koncepcja teoretyczna analizy kompozycyjnej układów przestrzennych. W: E. Bagiński, red. *Techniki i metody badawcze w planowaniu przestrzennym*. Wrocław: Oficyna Wydawnicza Politechniki Wrocławskiej, s. 101-113.
- Mitchell, B., 2017. Chapter 5: Spatial Bias in Convolutional Networks. W: *The Spatial Inductive Bias of Deep Learning*. Baltimore, Maryland: John Hopkins University, s. 151-156.
- Mitchell, W. J., 1977. *Computer-aided architectural design*. New York: Petrocelli/Charter.
- Mitchell, W. J., 1975. The Theoretical Foundation of Computer-Aided Architectural Design. *Environment and Planning B: Urban Analytics and City Science*, Volume 2, s. 127-150.
- Mohammad, A., Beorkrem, C. i Ellinger, J., 2019. Hybrid Elevations Using GAN Networks. *ACADIA*, s. 370-379.
- Mordvintsev, A., Olah, C. i Tyka, M., 2015. *Inceptionism: Going Deeper into Neural Networks*. [Online]
Dostęp: 2022
- Muzeum Architektury we Wrocławiu, 2019. <http://ma.wroc.pl/pl/kolekcja/>. [Online]
Dostęp: <http://ma.wroc.pl/pl/kolekcja/>
[Data uzyskania dostępu: 02 02 2024].
- Nair, V. i Hinton, G. E., 2010. Rectified Linear Units Improve Restricted Boltzmann Machines. *ICML'10: Proceedings of the 27th International Conference on International Conference on Machine Learning*, s. 807-814.
- Nakabayashi, M., Fukuda, T. i Yabuki, N., 2021. Mixed Reality Landscape Visualization Method with Automatic Discrimination Process for Dynamic Occlusion Handling Using Instance Segmentation. *eCAADe*, 2(39), s. 539-546.
- Narang S i Chowdhery, A., 2022. *Pathways Language Model (PaLM): Scaling to 540 Billion Parameters for Breakthrough Performance*, brak miejsca: Google Research.
- New York Times, The, 1958. Electronic "Brain" Teaches Itself. *The New York Times*, 13 07, s. 9.

- Newton, D., 2021. Visualizing Deep Learning Models for Urban Health Analysis. *eCAADe*, 1(39), s. 527-536.
- Newton, D., 2020. Anxious Landscapes Correlating the Built Environment with Mental Health through Deep Learning. *ACADIA*, s. 130-137.
- Newton, D., 2019. Deep Generative Learning for the Generation and Analysis of Architectural Plans with Small Datasets. *eCAADe*, 2(37), s. 21-23.
- Newton, D., 2018. Multi Objective Qualitative Optimization MOQO in Architectural Design. *eCAADe*, 1(36), s. 187-196.
- Newton, D., Piatkowski, D., Marshall, W. i Tendle, A., 2020. Deep Learning Methods for Urban Analysis and Health Estimation of Obesity. *eCAADe*, 1(38), s. 297-304.
- Ng, A., 2017. *Why Non-linear Activation Functions*, brak miejsca: deeplearning.ai.
- Ng, J. M., Khean, N., Madden, D., Fabbri, A., Gardner, N., Haeusler, M. H. i Zavoileas, Y., 2019. Optimising Image Classification Implementation of Convolutional Neural Network Algorithms to Distinguish Between Plans and Sections. *CAADRIA*, 2(24), s. 795-804.
- Nicholas, P., Chen, Y., Borpujari, N., Bartov, N. i Refsgaard, A., 2021. A Chained Machine Learning Approach to Motivate Retro Cladding of Residential Buildings. *eCAADe*, 1(39), s. 55-64.
- Niezabitowska, E. D., 2014. *Metody i techniki badawcze w architekturze*. Gliwice: Wydawnictwo Politechniki Śląskiej.
- Olah, C., Mordvintsev, A. i Schubert, L., 2017. *Feature Visualization: How neural networks build up their understanding of images*. [Online]
Dostęp: <https://distill.pub/2017/feature-visualization/>
[Data uzyskania dostępu: 02 02 2022].
- Olazaran, M., 1996. A Sociological Study of the Official History of the Perceptrons Controversy. *Social Studies of Science*, Tom 26, s. 611-659.
- Oliver, J. L., Tortosa, L. i Vicent, J. F., 2011. A neural network model to develop actions in urban complex systems represented by 2D meshes. *International Journal of Computer Mathematics*, 88(16), s. 3361-3379.
- OpenAI, 2023. GPT-4 Technical Report. *arXiv*.
- Openshaw, S., 1993. Modelling spatial interaction using a neural net. W: P. N. M.M. Fischer, red. *Geographic Information Systems, Spatial Modeling, and Policy Evaluation*. Berlin, Heidelberg and New York: Springer, s. 147-164.

- OpenStreetMap contributors, 2023. *OpenStreetMap*, brak miejsca: brak nazwiska
- Özel, G. i Ennemoser, B., 2019. Interdisciplinary AI: A Machine Learning System for Streamlining External Aesthetic and Cultural Influences in Architecture. *ACADIA*, s. 380-391.
- Pan, Y., Chen, X., Sun, Q. i Zhang, X., 2021. Monitoring Asphalt Pavement Aging and Damage Conditions from Low-Altitude UAV Imagery Based on a CNN Approach. *Canadian Journal of Remote Sensing*, 47(3), s. 432-449.
- Pan, Y., Qian, J. i Hu, Y., 2021. A Preliminary Study on the Formation of the General Layouts on the Northern Neighborhood Community Based on GauGAN Diversity Output Generator. *Proceedings of the 2020 DigitalFUTURES*, s. 179-188.
- Pasquero, C. i Poletto, M., 2020. DeepGreen: Coupling Biological and Artificial Intelligence in Urban Design. *ACADIA*, s. 668-677.
- Paszkowski, Z., 2007. Mieszkać na krawędzi. *Czasopismo Techniczne. Architektura*, 104(11), s. 203 - 214.
- Patel, S. V., Tchakerian, R., Morais, R. L., Zhang, J. i Cropper, S., 2020. The Emoting City: Designing feeling and artificial empathy in mediated environments. *eCAADe*, 2(38), s. 261-270.
- Pearson, K., 1901. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2(11), s. 559-572.
- Peng, W., Zhang, F. i Nagakura, T., 2017. Machines' Perception of Space Employing 3D Isovist Methods and a Convolutional Neural Network in Architectural Space Classification. *ACADIA*, 37, s. 474- 481.
- Pluta, K., 2014. *Przestrzenie publiczne miast europejskich: Projektowanie urbanistyczne*. Warszawa: Oficyna Wydawnicza Politechniki Warszawskiej.
- Pluta, K., 2001. Rola kompozycji urbanistycznej w kształtowaniu przestrzeni publicznych współczesnych miast. *Biuletyn Polska Akademia Nauk. Komitet Przestrzennego Zagospodarowania Kraju*, 194, s. 87-104.
- Płoszaj-Mazurek, M., 2023. *SLAD.AI Easy Carbon Footprint Calculator*. [Online]
Dostęp: <https://slad.ai/>
[Data uzyskania dostępu: 16 01 2024].
- Popper, K. R., 1977. *Logika odkrycia naukowego*. Warszawa: Państwowe Wydawnictwo Naukowe.

- Pourazar, H., Samadzadegan, F. i Javan, F. D., 2021. A deep 2D/3D Feature-Level fusion for classification of UAV multispectral imagery in urban areas. *Geocarto International*.
- Prezydent m.st. Warszawy, 2023. *Projekt Studium Uwarunkowań i Kierunków Zagospodarowania Przestrzennego m.st. Warszawy*. Warszawa: Rada m.st. Warszawy.
- Prezydent m.st. Warszawy, 2006 ze zmianami. *Studium Uwarunkowań i Kierunków Zagospodarowania Przestrzennego Miasta Stołecznego Warszawy*. Warszawa: Rada m. st. Warszawy.
- Prezydent miasta Łodzi, 2010. *Studium uwarunkowań i kierunków zagospodarowania przestrzennego miasta Łodzi*. Łódź: Rada Miejska w Łodzi.
- Prezydent miasta Łodzi, 2018. *Studium uwarunkowań i kierunków zagospodarowania przestrzennego miasta Łodzi*. brak miejsca: Rada Miejska w Łodzi.
- PWN, 2022. *Szereg Czasowy*. [Online]
Dostęp: <https://encyklopedia.pwn.pl/haslo/szereg-czasowy;3889308.html>
[Data uzyskania dostępu: 03 2022].
- PWN, 2024. *Merkatora Odwzorowanie*. [Online]
Dostęp: <https://encyklopedia.pwn.pl/haslo/Merkatora-odwzorowanie;3939813.html>
[Data uzyskania dostępu: 19 02 2024].
- Qayyum, A., Malik, A., Saad, N. M. i Mazher, M., 2019. Designing deep CNN models based on sparse coding for aerial imagery: a deep-features reduction approach. *European Journal of Remote Sensing*, 52(1), s. 221-239.
- Qi, Y., Drolma, S. C., Zhang, X., Liang, J., Jiang, H., Xu, J. i Ni, T., 2020. An investigation of the visual features of urban street vitality using a convolutional neural network. *Geo-spatial Information Science*, 23(4), s. 341-351.
- Radford, A., Kim, J., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S. i inni, 2021. Learning Transferable Visual Models From Natural Language Supervision. *International Conference on Machine Learning*.
- Rajkowska, J., 2002. *Pozdrowienia z Alej Jerozolimskich*. Warszawa: Soul-utions.Com.
- Ramesh, A., Dhariwal, P., Nichol, A., Chu, C. i Chen, M., 2022. Hierarchical Text-Conditional Image Generation with CLIP Latents. *arXiv*.
- Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A. i inni, 2021. Zero-Shot Text-to-Image Generation. *Proceedings of the 38th International Conference on Machine Learning, PMLR*, Tom 139, s. 8821-8831.

- Ranjan, N., Bhandari, S., Zhao, H. P., Kim, H. i Khan, P., 2020. City-Wide Traffic Congestion Prediction Based on CNN, LSTM and Transpose CNN. *IEEE Access*, Tom 8, s. 81606-81620.
- Rashevsky, N., 1938. *Mathematical Biophysics: Physicomathematical Foundations of Biology*. Chicago: University of Chicago Press.
- Ratzel, F., 1882. *Antropogeographis*. Stuttgart: J. Engelhorn.
- Raven, J., 2000. The Raven's Progressive Matrices: Change and Stability over Culture and Time. *Cognitive Psychology*, 41(1), s. 1-48.
- Rehm, L., 2014. *Camera51 for Android guides your photo composition*. [Online]
Dostęp: <https://www.dpreview.com/news/1849973697/camera51-for-android-guides-your-composition>
[Data uzyskania dostępu: 02 02 2024].
- Rhee, J., Veloso, P. i Krishnamurti, R., 2023. Three decades of machine learning with neural networks in computer-aided architectural design (1990–2021). *Design Science*, Tom 9.
- RIBA, 2024. *RIBA AI Report 2024*, London: Royal Institute of British Architects.
- Rombach, R., Blattmann, A., Lorenz, D., Esser, P. i Ommer, B., 2022. High-Resolution Image Synthesis With Latent Diffusion Models. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, s. 10684-10695.
- Ronneberger, O., Fischer, P. i Brox, T., 2015. U-Net: Convolutional Networks for Biomedical Image Segmentation. *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, s. 234-241.
- Rosenblatt, F., 1958. The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain. *Psychological Review*, 65(6), s. 386-408.
- Rosenblatt, F., 1961. *Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms*. Buffalo: Cornell Aeronautical Laboratory, Cornell University.
- Rossi, G. i Nicholas, P., 2020. Encoded Images: Representational Protocols for Integrating cGANs in Iterative Computational Design Processes. *ACADIA*, s. 218-227.
- Rossi, G. i Nicholas, P., 2018. Modelling A Complex Fabrication System New design tools for doubly curved metal surfaces fabricated using the English Wheel. *eCAADe*, 1(36), s. 811-820.
- Rossi, G. i Nicholas, P., 2019. Haptic Learning: Towards Neural Network based adaptive Cobot Path Planning for unstructured spaces. *eCAADe*, 2(37), s. 201-210.

- Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S. i inni, 2015. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, Tom 115, s. 211-252.
- Saggio, A., 2013. *The IT Revolution in Architecture. Thoughts on a Paradigm Shift*. Morrisville: Lulu.com.
- Saha, N., Haymaker, J. i Sheldon, D., 2020. Space Allocation Techniques (SAT) Computable Design Problems and Integrated Framework of Solvers. *ACADIA*, s. 248-257.
- SARP, Oddział Warszawski, 2015. *Studium koncepcyjne dotyczące centrów lokalnych w Warszawie*. Warszawa: OW SARP.
- Sato, G., Ishizawa, T., Iseda, H. i Kitahara, H., 2020. Automatic Generation of the Schematic Mechanical System Drawing by Generative Adversarial Network. *eCAADe*, 1(38), s. 403-410.
- Schmitt, G., 1999. *Information Architecture: Basis and Future of CAAD*. Basel: Birkhaeuser.
- Schuhmann, C., Beaumont, R., Vencu, R., Gordon, C., Wightman, R., Cherti, M. i inni, 2022. LAION-5B: An open large-scale dataset for training next generation image-text models. *NeurIPS 2022 Datasets and Benchmarks*.
- Schulman, J., Wolski, F., Dhariwal, P., Radford, A. i Klimov, O., 2017. Proximal Policy Optimization Algorithms. *arXiv*.
- Sebestyen, A., Rock, J. i Hirschberg, U. L., 2021. Towards Abductive Reasoning Based Computational Design Tools: Using Machine Learning as a way to explore the combined design spaces of multiple parametric models. *eCAADe*, 1(39), s. 141-150.
- Sebestyen, A. i Tyc, J., 2020. Machine Learning Methods in Energy Simulations for Architects and Designers: The implementation of supervised machine learning in the context of the computational design process. *eCAADe*, 1(38), s. 613-622.
- Sedlmeier, A. i Feld, S., 2018. Learning indoor space perception. *Journal of Location Based Services*, s. 179-214.
- Sejm RP, 2003. Art. 2. Dz.U. 2003 nr 80 poz. 717, Ustawa z dnia 27 marca 2003 r. o planowaniu i zagospodarowaniu przestrzennym z późniejszymi zmianami z 2023. *Dziennik Ustaw*.
- Sharma, D., Gupta, N., Chattopadhyay, C. i Mehta, S., 2017. DANIEL: A Deep Architecture for Automatic Analysis and Retrieval of Building Floor Plans. *International Conference on Document Analysis and Recognition*, 14, s. 420-425.

- Shih, S. G., 2016. On the Hierarchical Construction of SL Blocks A Generative System that Builds Self- Interlocking Structures. *Advances in Architectural Geometry*, s. 124-136.
- Silva, A., Dias, J. L., Gaspar, P. L. i de Brito, J., 2011. Service life prediction models for exterior stone cladding. *Building Research & Information*, 39(6), s. 637-653.
- Silver, D., Huang, A., Maddison, C., Gues, A., Sifre, L., van den Driessche, G. i inni, 2016. Mastering the game of Go with deep neural networks and tree search. *Nature*, 529, s. 484–489.
- Silvestre, J., Ikeda, Y. i Guena, F., 2016. Artificial Imagination of Architecture with Deep Convolutional Neural Network. *Living Systems and Micro-Utopias: Towards Continuous Designing, Proceedings of the 21st International Conference of the Association for Computer-Aided Architectural Design Research in Asia CAADRIA*, s. 881-890.
- Siła-Nowicka, K., Vandrol, J., Oshan, T., Long, J. A., Demšar, U. i Fotheringham, S., 2016. Analysis of human mobility patterns from GPS trajectories and contextual information. *International Journal of Geographical Information Science*, 5(30), s. 881-906.
- Singh, M. M., Schneider-Marin, P., Harter, H., Lang, W. i Geyer, P., 2020. Applying Deep Learning and Databases for Energy-efficient Architectural Design. *eCAADe*, 2(38), s. 79-87.
- Sinke, Y., Gatz, S., Tamke, M. i Thomsen, M. R., 2021. Machine Learning for Fabrication of Graded Knitted Membranes. *CDRF 2020: Proceedings of the 2020 DigitalFUTURES*, s. 309-319.
- Słyk, J., 2012. *Źródła Architektury Informacyjnej*. Warszawa: Oficyna Wydawnicza Politechniki Warszawskiej.
- Sobczyńska, K., 2021. Percepcja i postrzeganie formy jako wytyczne dla kształtowania panoramy miasta. *TEKA Komisji Architektury, Urbanistyki i Studiów Krajobrazowych*, 3, s. 63-69.
- Song, Y. i Yuan, P. F., 2021. A Research On Building Cluster Morphology Formation Based On Wind Environmental Performance And Deep Reinforcement Learning. *eCAADe*, 1(39), s. 335-344.
- Stenson, M. W., 2017. *Architectural Intelligence: How Designers and Architects Created the Digital Landscape*. Cambridge: MIT Press.

- Steinfeld, K., 2019. GAN Loci Imagining Place using Generative Adversarial Networks. *ACADIA*, s. 392-403.
- Steinfeld, K., 2020. Drawn Together: Machine Augmented Sketching in the Design Studio. *ACADIA*, s. 282-289.
- Stevenson, W. J., 1994a. Predicting Building Energy Parameters Using Artificial Neural Nets. *Transactions of the American Society of Heating, Refrigerating and Airconditioning Engineers*, 100, s. 1081-1087.
- Stevenson, W. J., 1994b. Using Artificial Neural Nets to Predict Building Energy Parameters. *ASHRAE Transactions*, 100(2), s. 1081-1087.
- Stojanovic, V., Trapp, M., Richter, R., Hagedorn, B. i Dollner, J., 2019. Semantic Enrichment of Indoor Point Clouds: An Overview of Progress towards Digital Twinning. *eCAADe*, 2(37), s. 809-818.
- Sukel, M., Rudinac, S. i Worring, M., 2019. Multimodal Classification of Urban Micro-Events. *Proceedings of the 27th ACM International Conference on Multimedia*, s. 1455–1463.
- Sukel, M., Rudinac, S. i Worring, M., 2020b. Detecting Urban Issues With the Object Detection Kit. *Proceeding of the 28th ACM International Conference on Multimedia*, s. 4518–4520.
- Sukel, M., Rudinac, S. i Worring, M., 2020a. Urban Object Detection Kit: A System for Collection and Analysis of Street-Level Imagery. *ICMR '20*, s. 509–516.
- Sun, C., Hsiao, C. W. S. M. i Chen, H. T., 2019. HorizonNet: Learning Room Layout with 1D Representation and Pano Stretch Data Augmentation. *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, s. 1047-1056.
- Sun, Y., Jiang, L. i Zheng, H., 2020. A Machine Learning Method of Predicting Behavior Vitality Using Open Source Data. *ACADIA*, s. 160-168.
- Szwankowski, E., 1970. *Ulice i place Warszawy*. Warszawa: Państwowe Wydawnictwo Naukowe.
- Śleszyński, P., Chmielewski, T. J., Chmielewski, S. i Kułak, A., 2018. Estetyczne koszty chaosu przestrzennego. *Studia KPZK*, 182(2), s. 365-403.
- Tadeusiewicz, R. i Szaleniec, M., 2015. *Leksykon Sieci Neuronowych*. I red. Wrocław: Wydawnictwo Fundacji „Projekt Nauka”.

- Takizawa, A. i Furuta, A., 2017. 3D Spatial Analysis Method with First-Person Viewpoint by Deep Convolutional Neural Network with Omnidirectional RGB and Depth Images. *eCAADe*, 35, s. 693-702.
- Tamas, J. i Toth, Z., 2018. Classification based symbolic indoor positioning over the Miskolc IIS data set. *Journal of Location Based Services*, 12(1), s. 2-18.
- Tamke, M., Zwierzycki, M., Deleuran, A. H. i Baranovskaya, Y. S., 2017. Lace Wall Extending design intuition through machine learning. W: *Fabricate 2017*. brak miejsca:UCL Press, s. 98-105.
- Tandy, C. R. V., 1967. The isovist method of landscape survey. *Symposium: Methods of Landscape Analysis*, s. 9-10.
- Tarabishy, S., Psarras, S., Kosicki, M. i Tsigkari, M., 2020. Deep Learning surrogate models for spatial and visual connectivity. *International Journal of Architectural Computing*, 18(1), s. 53-66.
- Thomsen, M., Nicholas, P., Tamke, M., Gatz, S. i Sinke, Y., 2019. Predicting and steering performance in architectural materials. *eCAADe*, 2(37), s. 485-494.
- Tian, R. W. T. i Gün, O. Y., 2020. Data Driven Midsole Performance Oriented Midsole Design Using Computational Multi Objective Optimization. *ACADIA*, s. 188-197.
- Tian, R., 2020. Suggestive Site Planning with Conditional GAN and Urban GIS Data. *Proceedings of the 2020 DigitalFUTURES*, s. 103-113.
- Toulkeridou, V., 2019. Steps towards AI augmented parametric modeling systems for supporting design exploration. *eCAADe*, 1(37), s. 81-90.
- Vaswani, A., Shazeer, N., Parmar, J., Uszkoreit, J., Jones, L., Gomez, A. i inni, 2017. Attention Is All You Need. *Advances in Neural Information Processing Systems 30 (NIPS 2017)*.
- Veloso, P. i Krishnamurti, R., 2020. An Academy of Spatial Agents: Generating spatial configurations with deep reinforcement learning. *eCAADe*, 2(38), s. 191-200.
- Vinyals, O., Babuschkin, I., Czarnecki, W., Mathieu, M., Dudzik, A., Chung, J. i inni, 2019. Grandmaster level in StarCraft II using multi-agent reinforcement learning. *Nature*, 575, s. 350–354.
- The Matrix*. 1999. [Film] Reżyser: L Wachowski, A Wachowski. USA: Warner Bros.
- Wang, Z. i Bovik, A. C., 2008. Mean squared error: Love it or leave it? A new look at Signal Fidelity Measures.. *IEEE Signal Processing Magazine*, 26(1), s. 98–117.
- Watkins, C. J. C. H. i Dayan, P., 1992. Q-learning. *Machine Learning*, Tom 8, s. 279–292.

- Wejchert, K., 1984. *Elementy Kompozycji Urbanistycznej*. Warszawa: Arkady.
- Wibranek, B., Liu, Y., Funk, N., Belousov, B., Peters, J. i Tessmann, O., 2021. Reinforcement Learning for Sequential Assembly of SL Blocks: Self-interlocking combinatorial design based on Machine Learning. *eCAADe*, 1(39), s. 27-39.
- Widrow, B. i Hoff, M. E., 1960. Adaptive Switching Circuits. *IRE WESCON Convention Record*, Tom 4, s. 96-104.
- Wightman, R., 2019. *PyTorch Image Models*. [Online]
Dostęp: <https://github.com/rwightman/pytorch-image-models>
[Data uzyskania dostępu: 02 02 2024].
- Wu, J., Woo, H., Tamura, Y., Moro, A., Massaroli, S., Yamashita, A. i Asama, H., 2019. Pedestrian trajectory prediction using BiRNN encoder-decoder framework. *Advanced Robotics*, 33(18), s. 956-969.
- Wu, S., 2021. Approach to Auto-Recognition of Human Trajectory in Squares using Machine Learning-Based Methods An application of the Yolo-v3 and the DeepSORT algorithm. *eCAADe*, 2(39), s. 361-370.
- Xiao, Y., Chen, S., Ikeda, Y. i Hotta, K., 2020. Automatic Recognition and Segmentation of Architectural Elements From 2D Drawings By Convolutional Neural Network. *CAADRIA*, 1(25), s. 843-852.
- Yazıcıoğlu, G. i Dino, İ. G., 2021. From Streetscape to Data Semantic segmentation for the prediction of outdoor thermal comfort. *eCAADe*, 39, s. 555-562.
- Yazici, S., 2020. A machine-learning model driven by geometry, material and structural performance data in architectural design process. *eCAADe*, 1(38), s. 411-418.
- Yeh, A. G. i Xia, L., 2004. Integration of Neural Networks and Cellular Automata for Urban Planning. *Geo-spatial Information Science*, 7(1), s. 6-13.
- Yetkin, O. i Sorguç, A. G., 2019. Design Space Exploration of Initial Structural Design Alternatives via Artificial Neural Networks. 1(37), s. 55-60.
- Yu, D., 2020. Reprogramming Urban Block by Machine Creativity How to use neural networks as generative tools to design space. *eCAADe*, s. 249-258.
- Zandavali, B. A. i Garcia, M. J., 2019. Automated Brick Pattern Generator for Robotic Assembly using Machine Learning and Images. *eCAADe*, 3(37), s. 217-226.
- Zannoni, R., 1772. *Plan de Varsovie*. [Sztuka] (Śląska Biblioteka Cyfrowa).
- Zevi, B., 1993. *Architecture As Space: How to look at architecture*. New York: Da Capo Press.

- Zhai, Y., Yao, Y., Guan, Q., Liang, X., Li, X., Pan, Y. i inni, 2020. Simulating urban land use change by integrating a convolutional neural network with vector-based cellular automata. *International Journal of Geographical Information Science*, 34(7), s. 1475-1499.
- Zhang, H. i Huang, Y., 2020. Machine Learning Aided 2D 3D Architectural Form Finding at High Resolution. *Proceedings of the 2020 DigitalFUTURES. CDRF 2020*, s. 159-168.
- Zhang, H., 2020. Text-To-Form: 3D Prediction by Linguistic Descriptions. *ACADIA*, s. 238-247.
- Zhang, J., Fukuda, T. i Yabuki, N., 2020. A Large-Scale Measurement and Quantitative Analysis Method of Facade Color in the Urban Street Using Deep Learning. *CDRF 2020*, s. 93-102.
- Zheng, H. i Huang, W., 2018. Understanding and Visualizing Generative Adversarial Networks in Architectural Drawings. *CAADRIA*, 23, s. 233-238.
- Zheng, H., Wang, X., Qi, Z., Sun, S. i Akbarzadeh, M., 2020. Generating and Optimizing a Funicular Arch Floor Structure. *ACADIA*, s. 208-217.
- Zhou, B., Lapedriza, A., Xiao, J., Torralba, A. i Oliva, A., 2014. Learning Deep Features for Scene Recognition using Places Database. *Advances in Neural Information Processing Systems (NIPS)*, 27.
- Zisserman, A. i Simonyan, K., 2015. Very Deep Convolutional Networks for Large-Scale Image Recognition. *3rd International Conference on Learning Representations (ICLR 2015)*, Computational and Biological Learning Society, s. 1-14.
- Zohier, I., El Antably, A. i Madani, A. S., 2020. An AI Lens on Historic Cairo A Deep Learning Application for Minaret Classification. *ACADIA*, s. 426-434.
- Żórawski, J., 1962. *O Budowie Formy Architektonicznej*. Warszawa: Arkady.

Spis ilustracji

- Rys. 1. Wizualizacja 37 cech (oś pionowa) wyodrębnionych ze 149 publikacji (oś pozioma) dotyczących zastosowania sieci neuronowych w projektowaniu wspomaganim komputerowo. Biały kolor oznacza powiązanie danej cechy z publikacją. Wyodrębnione cechy pozwoliły na organizację zebranych źródeł i przeprowadzenie ich analizy ilościowej. Opracowanie własne.
- Rys. 2. Stosowanie sieci neuronowych w projektowaniu w różnych skalach w publikacjach dotyczących komputerowego wspomagania projektowania architektonicznego. Opracowanie własne.
- Rys. 3. Wizualizacja skal, w których stosowane są sieci neuronowe wykonana przy pomocy algorytmu t-SNE na podstawie 37 wyodrębnionych cech badanych artykułów. Każdy punkt reprezentuje odrębną publikację, kolor punktu odpowiada skali zastosowania sieci neuronowej. Opracowanie własne.
- Rys. 4. Nieistniejący, syntetyczny krajobraz wygenerowany przez opracowany przez Kyle’a Steinfelda neuronowy algorytm GAN Loci wizualizujący „genius loci” dzielnicy Blijdorp w Rotterdamie (Steinfeld, 2019).
- Rys. 5. Algorytm GAN_Physarum (oparty na neuronowym transferze stylu) opracowany przez Pasquero i Poletto zastosowany do przetworzenia satelitarnych zdjęć tkanki urbanistycznej w skali 1 x 1 km (po lewej) i 10 x 10 km (po prawej) (Pasquero i Poletto, 2020).
- Rys. 6. Algorytm GAN_Physarum (oparty na neuronowym transferze stylu) opracowany przez Pasquero i Poletto zastosowany do przetworzenia Centre Pompidou sfotografowanego z lotu ptaka (Pasquero i Poletto, 2020).
- Rys. 7. Przykładowe zdjęcia z drona pobrane z bazy danych UAVid stworzonej przez Lyu, Vosselmana, Xia i in. służącej do trenowania neuronowych algorytmów semantycznej segmentacji. Poszczególne klasy obiektów widoczne na zdjęciu oznaczone są na wizualizacjach różnymi kolorami wyznaczającymi ich obrysy (Lyu, i in., 2018).
- Rys. 8. Fantastyczne krajobrazy „wyśnzione” przez opracowany przez Google algorytm DeepDream, zaaplikowany do sieci neuronowej służącej klasyfikacji obrazów. Krajobrazy zostały wygenerowane na podstawie czystego szumu poprzez iteracyjne wzmacnianie na obrazie wejściowym cech zwiększających siłę aktywacji wybranych neuronów i warstw sieci neuronowej (Mordvintsev, i in., 2015).

- Rys. 9. Sekwencja generowania warstwy nowych detali architektonicznych na zadanym zdjęciu domu jednorodzinnego. Metoda zaproponowana przez Silvestra, Ikedę i Guenę z Keio University, oparta na opracowanym przez Google algorytmie DeepDream (Silvestre, i in., 2016).
- Rys. 10. Wizualizacje stworzone przez Güvença Özela i Benjamin Ennemosera na podstawie neuronowego transferu stylu. Wygenerowane przez sieć neuronową dwuwymiarowe obrazy zostały przekształcone proceduralnie w modele trójwymiarowe i osadzone kompozytowo w oryginalnych kontekstach (Özel i Ennemoser, 2019).
- Rys. 11. Defamiliaryzacja rzutu architektonicznego będąca skutkiem prac Matiasa del Campo, Sandry Carlson i Alexandry Manninger. Po lewej rezultat transferu stylu między rzutem modernistycznym a barokowym (del Campo, i in., 2019), w centrum z lewej rezultat transferu stylu między rzutem barokowym a modernistycznym (del Campo, i in., 2020), po prawej abstrakcyjne „rzuty” stworzone przy pomocy generatywnych sieci przeciwstawnych (del Campo, i in., 2019).
- Rys. 12. Demonstracja skuteczności systemu opracowanego przez Asa i in. Po lewej wygenerowany przez sieć neuronową graf programu funkcjonalnego domu jednorodzinnego, po prawej automatycznie wygenerowany na podstawie grafu parametryczny, koncepcyjny model kubaturowy (As, i in., 2018).
- Rys. 13. Bryły wygenerowane przez system opracowany przez Liu, Liao i Srivastavę. Przeciwna sieć neuronowa generowała takie parametry dla algorytmu parametrycznego, by wynikowe bryły w jak największym stopniu spełniały wymagane cechy. Po lewej u góry – bryła z wysokim wynikiem „monumentalności”, po prawej u góry – „solidności”, po lewej u dołu – „delikatności” i po prawej u dołu – „dynamiczności” (Liu, i in., 2019).
- Rys. 14. Możliwość przeprowadzania semantycznych operacji arytmetycznych na wektorach w przestrzeni ukrytej zademonstrowana przez Sebestyena i in. Ściśnięty sześciąt minus sześciąt plus kula = ściśnięta kula (Sebestyen, i in., 2021).
- Rys. 15. System Sketch2Pix opracowany przez Kyle’a Steinfelda służący do przetwarzania linearnych szkiców aksonometrycznych na ich fotorealistyczne wizualizacje. Po lewej przykłady prostych modułów zwizualizowanych przez algorytm, po prawej dwa przykłady kolaży stworzonych przez studentów architektury korzystających z algorytmu Sketch2Pix w trakcie zajęć projektowych (Steinfeld, 2020).

- Rys. 16. Projekt plomby architektonicznej wygenerowany przez algorytm PatchMatch wspomagany przez zespół pięciu klasyfikacyjnych sieci neuronowych dokonujących selekcji najlepszych elementów projektowych (Dzieduszyński, i in., 2020).
- Rys. 17. Proces treningu algorytmu służącego do redukcji liczby wymiarów definiujących projekt parametryczny stworzony przez Johna Hardinga. Na ilustracji przedstawiono proces redukcji parametrów z pięciu do dwóch, możliwych do zwizualizowania w postaci dwuwymiarowej płaszczyzny (Harding, 2016).
- Rys. 18. Wspomagane przez sieć neuronową metody fabrykacji giętych, blaszanych paneli elewacyjnych opracowane przez Gabriellę Rossi i Paula Nicholasa. Po lewej – druk 3D wzmocnień wygenerowanych przez sieć neuronową w odpowiedzi na przewidywane obciążenia wywołane parciem wiatru. Po prawej – wytwarzanie podobnych wzmocnień w formie wytłoczonych w blasze podłużnych żeber (Rossi i Nicholas, 2020).
- Rys. 19. Instalacja przestrzenna „Lace Wall” autorstwa Tamke, Zwierzyckiego i in. W „Lace Wall” sieci neuronowe zostały wykorzystane do klasyfikacji modułów, które ze względu na zbyt duże deformacje w skali mikro uniemożliwiały stworzenie docelowej formy w skali makro (Tamke, i in., 2017).
- Rys. 20. Artykuły pośrednio dotyczące morfologii urbanistycznej, kompozycji przestrzennej oraz artykuły przeglądowe w publikacjach dotyczących komputerowego wspomagania projektowania architektonicznego przy wykorzystaniu sieci neuronowych. Opracowanie własne.
- Rys. 21. Złota kopuła State House w Bostonie ginąca na tle nowszych wieżowców, m.in. One Beacon Street z roku 1972 (fot. Google Maps, 2023)
- Rys. 22. Widok na plac Zbawiciela będący punktem głównym założenia gwiazdowego (fot. Google Maps, 2023)
- Rys. 23. Zlewiska centrów lokalnych Brennana. Centra lokalne przesunięte są w kierunku głównego centrum miasta [na podstawie (Brennan, 1948)]
- Rys. 24. Centra ekscentryczne Alexandra. Znajdujące się na granicach między obszarami społeczności [na podstawie (Alexander, i in., 1977)]
- Rys. 25. Zbliżenie na centrum ekscentryczne Alexandra. Naprowadzają na nie ścieżki i bariery wyznaczające obszar społeczności. Opracowanie własne
- Rys. 26. Strefy ochronne z wyraźnie oznaczonymi punktami głównymi pochodzące z trzydziestej piątej wersji paryskiego planu urbanistycznego PLU dla obszaru Pól Elizejskich (Le Conseil de Paris, 2015).

- Rys. 27. Legenda do rysunku stref ochronnych z trzydziestej piątej wersji PLU (Le Conseil de Paris, 2015).
- Rys. 28. Zrzut ekranu z trybu deweloperskiego programu Camera51, pokazujący rozpoznane przez aplikację elementy kompozycyjne. Ikona w centrum ekranu pokazuje, w jaki sposób przesunąć aparat, by polepszyć kompozycję kadru (Masad i Shamir, 2017).
- Rys. 29. Wizualizacja wybranych głowic uwagi sieci neuronowej przetwarzającej proste kompozycje przestrzenne naszkicowane przez Juliusza Żórawskiego (Żórawski, 1962). Kolorem czerwonym oznaczone są miejsca w obrazie, które dana głowica uwagi uznała za istotne. Wykorzystana sieć to Vision Transformer (Dosovitskiy, i in., 2021) wytrenowany przez Rossa Wightmana (Wightman, 2019) do rozpoznawania klas ze zbioru obrazów ImageNet (Russakovsky, i in., 2015). Sieć nie próbuje odczytywać kontekstów kompozycyjnych, tylko stara się rozpoznać, co znajduje się na obrazie (w tym przypadku wszystkie szkice Żórawskiego zostały rozpoznane jako zdjęcia łańcuchów). Sieć zwraca uwagę na różne fragmenty obrazu, jednak dużą wagę przykłada także do punktów formalnie podkreślonych. Opracowanie własne.
- Rys. 30. Dwie metody wyznaczania rdzeni integracji zaproponowane przez Billa Hilliera. U góry – metoda analizy nakładających się elementów wypukłych, u dołu – metoda analizy jednowymiarowych elementów liniowych. Metody zostały zaprezentowane na dwóch modelach tkanki miejskiej, jednej z czytelnym punktem głównym, drugiej bez takiego punktu. Obie metody dają zbliżone rezultaty (Hillier, 2007).
- Rys. 31. Szkic Sigmunta Freuda z 1885 roku, będący prawdopodobnie pierwszym koncepcyjnym modelem sieci neuronowej. Rysunek przedstawia przepływ energii od bodźca $Q'\eta$ przez neuron a do neuronu b (Freud, 1950/1895). Teoria Freuda była dosyć niejasna, niepoparta empirycznie i ostatecznie okazała się błędna (McCarley, 1998), jednak stała się jedną z wielu inspiracji dla twórców pierwszych funkcjonalnych sieci neuronowych z lat 50. dwudziestego wieku (Minsky i Sykes, 2017).
- Rys. 32. *Diagram perceptronu elementarnego z jednym neuronem dokonującym klasyfikacji binarnej. Dane wejściowe zespolone są stałym połączeniem z jednostkami asocjacyjnymi „A”. Połączenia między jednostkami „A” a jednostką responsywną „R” są skalowane względem wag, w których zapisana jest „zdolność” perceptronu do rozwiązania danego problemu. Jednostka „R”, którą dzisiaj nazwalibyśmy neuronem, liczy średnią ważoną sygnałów wejściowych i przepuszcza ją przez schodkową funkcję*

aktywacji. Efektem jest dana wyjściowa: klasa przyjmująca wartość -1 lub 1. Ilustracja przygotowana na podstawie (Rosenblatt, 1961).

- Rys. 33. Porównanie rozwiązania problemu alternatywy wykluczającej przez jednowarstwową i dwuwarstwową sieć neuronową. Klas XOR nie da się oddzielić przy pomocy jednej prostej. Jednowarstwową sieć neuronową nieprawidłowo klasyfikuje wyjście y dla $x_1 = -1$ i $x_2 = -1$. Dwuwarstwową sieć neuronową rozwiązuje problem bez trudu. Zamiast aktywacji schodkowej zastosowano jej różniczkowalne przybliżenie: funkcję sigmoidalną. Ilustracja oparta na wynikach symulacji przeprowadzonej w przeglądarkowej implementacji jednokierunkowej sieci neuronowej ConvNetJS (Karpathy, 2014).
- Rys. 34. Sieć neuronowa przekształcająca wektor wejściowy X na wektor wyjściowy Y . Przykładowo: możemy wytrenowanej sieci pokazać zdjęcie kota i na wyjściu otrzymać prawidłowo zidentyfikowaną etykietę „KOT!”. Opracowanie własne.
- Rys. 35. Diagram propagacji w przód na poziomie pojedynczego neuronu. Sygnały wejściowe x_n sumowane są zgodnie z ich wagami w'_n , do sumy dodawana jest wartość progu b' . Wynik przepuszczany jest przez nieliniową, różniczkowalną funkcję aktywacji. Wartość funkcji stanowi sygnał dla neuronów w kolejnej warstwie. Opracowanie własne na podstawie (Hinton, i in., 1986).
- Rys. 36. Diagram propagacji wstecznej na poziomie pojedynczego neuronu. Parametry w''_n oraz b'' są aktualizowane proporcjonalnie do szybkości uczenia α i pochodnej cząstkowej funkcji kosztu J względem danego parametru. Globalnie, z każdym cyklem propagacji wstecznej, funkcja kosztu schodzi w kierunku lokalnego minimum. Opracowanie własne na podstawie (Hinton, i in., 1986).
- Rys. 37. Rodzaje sieci neuronowych w publikacjach dotyczących komputerowego wspomaganie projektowania architektonicznego. Opracowanie własne.
- Rys. 38. Wizualizacja rodzajów sieci neuronowych w publikacjach CAAD wykonana przy pomocy algorytmu t-SNE na podstawie 37 wyodrębnionych cech badanych artykułów. Każdy punkt reprezentuje odrębną publikację. Opracowanie własne.
- Rys. 39. Diagram klasycznej, jednokierunkowej sieci w pełni połączonej (po lewej) oraz sieci typu autoenkoder (po prawej). Wykorzystywane w praktyce sieci zazwyczaj mają więcej neuronów oraz warstw. Opracowanie własne.
- Rys. 40. Diagram warstwy splotowej w splotowej sieci neuronowej. Dane wejściowe skanowane są sekwencyjnie przez filtry splotowe obliczające wartości funkcji splotu na podstawie

odczytanych sygnałów. Dane na wyjściu mają formę wielowymiarowych tablic, po jednej tablicy na jeden filtr splotowy. Trenowalne parametry znajdują się w filtrach splotowych. Opracowanie własne.

Rys. 41. Diagram ilustrujący pętlę działania agenta DRL w środowisku symulacyjnym. Działanie agenta zostaje nagrodzone przez funkcję straty i skutkuje zmianą jego stanu. Opracowanie własne.

Rys. 42. Diagram ilustrujący sposób działania sieci RNN typu LSTM. W następujących po sobie krokach sieć otrzymuje wejściowe dane z sekwencji wejściowej X i przetwarza je na sekwencję wyjściową Y. Dodatkowo sieć otrzymuje na wejściu stany ukryte sieci z poprzedniego kroku. W modelu LSTM sieć otrzymuje także mechanizm pamięci, pozwalający jej nauczyć się możliwości zapisywania pewnych istotnych danych w komórkach pamięci, których stan sieć może odczytać w kolejnych krokach iteracji. Opracowanie własne.

Rys. 43. Wstępne testy skuteczności (%) FNN oraz CNN w przetwarzaniu wielowymiarowych zadań przestrzennych (po lewej) oraz przykłady rozwiązań wygenerowanych przez oba typy sieci (po prawej). Oba typy sieci miały za zadanie wygenerowanie trzech wysp spełniających serię zasad kompozycyjnych reprezentowanych w zbiorze treningowym. CNN osiąga skuteczność o 4 - 14% lepszą niż FNN o porównywalnej liczbie parametrów (Dzieduszyński, 2022b).

Rys. 44. Proste zasady kompozycyjne składające się na złożoną zasadę kompozycyjną wykorzystaną w eksperymencie wstępnym (Dzieduszyński, 2022a).

Rys. 45. Dozwolone osie kompozycyjne dla zbioru treningowego i zbioru testowego o dystrybucji treningowej (Dzieduszyński, 2022a).

Rys. 46. Dozwolone osie kompozycyjne dla zbioru walidacyjnego i zbioru testowego (Dzieduszyński, 2022a).

Rys. 47. Trzy scenariusze testowane w prototypie wstępnym polegające na generowaniu całości kompozycji, jej fragmentu oraz pojedynczego piksela (Dzieduszyński, 2022a).

Rys. 48. Przykład przeprowadzonej optymalizacji hiperparametru znormalizowanej wartości progu aktywacji pikseli wyjściowych. Optymalizacja przeprowadzona została metodą grid search w zakresie od 0,35 do 0,65 na zbiorach walidacyjnych dla scenariusza I i II względem skuteczności sieci na badanych zasadach kompozycyjnych. Optymalna wartość wyniosła 0,5 dla scenariusza I (po lewej) i 0,4 dla scenariusza II (po prawej) (Dzieduszyński, 2022a).

- Rys. 49. Diagram sieci neuronowej wykorzystanej w scenariuszu I (Dzieduszyński, 2022a).
- Rys. 50. Wykres przedstawiający ewolucję skuteczności sieci w scenariuszu I w miarę treningu.
By zapobiec przetrenowaniu sieci, trening został zatrzymany w epoce trzeciej (technika regularyzacyjna early stopping) (Dzieduszyński, 2022a).
- Rys. 51. Wybór przykładów pozytywnych rozwiązanych prawidłowo i negatywnych, w których sieć miała problemy (Dzieduszyński, 2022a).
- Rys. 52. Diagram sieci neuronowej wykorzystanej w scenariuszu II (Dzieduszyński, 2022a).
- Rys. 53. Wykres przedstawiający ewolucję skuteczności sieci w scenariuszu II w miarę treningu.
By zapobiec przetrenowaniu sieci, trening został zatrzymany w epoce 75 (Dzieduszyński, 2022a).
- Rys. 54. Wybór przykładów pozytywnych, z którymi sieć scenariusza II radziła sobie bardzo dobrze i negatywnych, w których sieć miała problemy (Dzieduszyński, 2022a).
- Rys. 55. Diagram sieci neuronowej wykorzystanej w scenariuszu III (Dzieduszyński, 2022a).
- Rys. 56. Wykres przedstawiający ewolucję skuteczności sieci w scenariuszu III w miarę treningu.
W tym przypadku doprowadzono trening do końca, early stopping nie był konieczny (Dzieduszyński, 2022a).
- Rys. 57. Diagram przedstawiający wyniki działania sieci w scenariuszu. Każdy z pikseli w kompozycji był sekwencyjnie zakrywany i generowany przez sieć osobno. W górnym rzędzie – kompozycje wejściowe, w środkowym – mapa aktywacji dla każdego z 64 pikseli, w dolnym – decyzje podjęte przez sieć dla każdego piksela (czerwony kolor oznacza decyzję o narysowaniu białego piksela, niebieski kolor oznacza decyzję o wygenerowaniu czarnego piksela). Wybór przykładów pozytywnych i negatywnych (Dzieduszyński, 2022a).
- Rys. 58. Wyniki sieci dla ręcznie rysowanych przykładów brzegowych. Każdy z pikseli w kompozycji był sekwencyjnie zakrywany i generowany przez sieć osobno. W górnym rzędzie – kompozycje wejściowe, w środkowym – mapa aktywacji dla każdego z 64 pikseli, w dolnym – decyzje podjęte przez sieć dla każdego piksela (czerwony kolor oznacza decyzję o narysowaniu białego piksela, niebieski kolor oznacza decyzję o wygenerowaniu czarnego piksela) (Dzieduszyński, 2022a).
- Rys. 59. Przykłady białółękich struktur urbanistycznych zaproponowanych przez sieć neuronową wytrenowaną na szwarceplanach zabudowy Śródmieścia Warszawy. Zabudowa proponowana przez sieć znajduje się w czerwonych ramkach. Nawet przy niewielkiej rozdzielczości wynikającej z ograniczeń strukturalnych sieci i bardzo

ograniczonych kontekstach reprezentowanych w przetwarzanych danych, sieć wydaje się respektować gęstość i osiowość zabudowy, skalę obrysów budynków i konieczność kontynuacji pierzei (Dzieduszyński, 2022a).

Rys. 60. Diagram przedstawiający centrowanie map topograficzny względem wartości minimalnej. Przy pobieraniu danych z serwisu Mapbox opracowany skrypt API rozpoznaje najmniejszą wartość piksela na obrazie i odejmuje tę wartość od wartości pozostałych pikseli. Dzięki temu mapa topograficzna przedstawia tylko wysokości względne i pozostawia więcej przestrzeni dla mapy zabudowy. Opracowanie własne.

Rys. 61. Mapa przestrzeni cech wykorzystanej w części eksperymentalnej. Mapa składa się z trzech, nałożonych na siebie map dwuwymiarowych przedstawionych na potrzeby wizualizacji w trzech kolorach podstawowych. Kolorem niebieskim oznaczono mapę topografii i zabudowy terenu, zielonym – intensywność zieleni, czerwonym – układ komunikacyjny. Opracowanie własne.

Rys. 62. Diagram przedstawiający możliwość przedstawienia przestrzeni cech w macierzowej postaci trójwymiarowej (po lewej) oraz w zwokselizowanej postaci pięciowymiarowej (po prawej). W obu interpretacjach osie X i Y odpowiadają współrzędnym geograficznym. Opracowanie własne.

Rys. 63. Przykłady trzech map cech ze zbioru treningowego sieci. Kanał niebieski przejaszczony na potrzeby wizualizacji. Od lewej Stare Miasto w Krakowie, Łuk Triumfalny w Paryżu i Cytadela w Irbilu. Opracowanie własne. Mapa cech przygotowana w oparciu o serwis Mapbox (Mapbox, 2023) i Open Street Map (OpenStreetMap contributors, 2023).

Rys. 64. Próby stworzenia algorytmu rozszerzającego zestaw treningowy o przykłady syntetyczne. Środowisko Grasshopper. Ostatecznie algorytm nie został wykorzystany. Opracowanie własne.

Rys. 65. W górnym rzędzie – przykłady trzech zasad kompozycyjnych podporządkowanych punktom głównym zdefiniowane przez Żórawskiego (Żórawski, 1962). W środkowym rzędzie – osie kompozycyjne i punkty główne stanowiące punkty wyjścia dla algorytmu tworzącego syntetyczne przykłady treningowe. W dolnym rzędzie – wygenerowane przez algorytm syntetyczne konteksty przestrzenne. Ostatecznie algorytm nie został wykorzystany. Opracowanie własne.

Rys. 66. Ewolucja mapy celu dla sieci neuronowej na przykładzie florenckiej katedry Santa Maria del Fiore. Od lewej do prawej: mapa cech (kanał niebieski przejaszczony

na potrzeby wizualizacji); pierwsza generacja mapy celu z pojedynczym punktem głównym; druga generacja mapy celu uzupełniona o gradient dookoła punktu (gradient przejaśniony na potrzeby wizualizacji) oraz ostateczna mapa celu uzupełniona o osie kompozycyjne (gradient przejaśniony na potrzeby wizualizacji). Opracowanie własne. Mapa cech przygotowana w oparciu o serwis Mapbox (Mapbox, 2023) i Open Street Map (OpenStreetMap contributors, 2023).

Rys. 67. Diagram modelu sieci neuronowej typu GAN wykorzystanej w eksperymencie głównym. Sieć inspirowana jest modelem pix2pix (Isola, i in., 2017). Generator tworzy mapy celu na podstawie map cech. Dyskryminator porównuje wyjście generatora z treningową mapą celu i próbuje oznaczać, które fragmenty obrazów są stworzone przez generator, a które pochodzą ze zbioru treningowego. Generator jest nagradzany za obrazy potrafiące oszukać dyskryminator. Obie sieci uczą się wspólnie, zwiększając wzajemnie swoją skuteczność. Przy inferencji wytrenowanego modelu wykorzystywany jest tylko generator. Opracowanie własne.

Rys. 68. Porównanie metod ewaluacji podobieństwa obrazów MSE, SSIM i CW-SSIM przygotowane przez Zhou Wanga i Alana C. Bovika. a) obraz źródłowy; b) zwiększony kontrast; c) przesunięcie luminancji; d) szum gaussowski; e) szum impulsowy; f) kompresja JPEG; g) rozmycie; h) oddalenie; i) przesunięcie w prawo; j) przesunięcie w lewo; k) obrót przeciwny do ruchu wskazówek zegara; l) obrót zgodny z ruchem wskazówek zegara (Wang i Bovik, 2008). W przypadku opracowanego algorytmu żadna z powyższych metod ewaluacji nie była skorelowana z rzeczywistą jakością tworzonych przez sieć neuronową obrazów.

Rys. 69. Wartości funkcji straty na zbiorze treningowym dla punktów kontrolnych od 1 do 254. Oś horyzontalna odpowiada kolejnym punktom kontrolnym (ze względu na nierównomierne próbkowanie punktów kontrolnych oś ta nie stanowi osi czasu). Linia szara: wartości funkcji straty uśrednione z 4 podzbiorów treningowych. Linia czarna: średnia bieżąca z 9 kolejnych punktów kontrolnych (dla 4 podzbiorów treningowych). Linia czerwona: logarytmiczna linia trendu (dla 4 podzbiorów treningowych). Opracowanie własne.

Rys. 70. Przykłady błędnych przypadków, w których nie został rozpoznany żaden punkt główny. Na górze mapy cech, na dole odpowiadające im obrazy stworzone przez sieć neuronową. Po lewej pałac w Wilanowie. Po środku Pałac na Wyspie w Łazienkach

Królewskich. Po prawej Stadion Narodowy. Kanał niebieski map cech przejawiony na potrzeby wizualizacji. Opracowanie własne.

Rys. 71. Przykłady, w których niewygenerowanie punktu głównego nie stanowiło błędu.

Na górze mapy cech, na dole odpowiadające im obrazy stworzone przez sieć neuronową. Po lewej – hale przemysłowe zakładów FSO. W środku – fragment Lasu Bielańskiego. Po prawej – pola uprawne w pobliżu Fortu Chrzanów. Kanał niebieski map cech przejawiony na potrzeby wizualizacji. Opracowanie własne.

Rys. 72. Demonstracja seryjnego przetwarzania większych obszarów przy pomocy opracowanej sieci neuronowej. 64 punkty główne i osie kompozycyjne rozpoznane przez sieć neuronową na obszarze 25 km² lewobrzeżnej zabudowy śródmiejskiej Warszawy. Po lewej – obrazy wygenerowane przez sieć nałożone na mapę przestrzeni cech. Po prawej – nałożone na mapę Open Street Map. Czerwone linie to osie kompozycyjne. Pełna lista punktów rozpoznanych przez sieć przedstawiona jest w aneksie (Załącznik 6). Opracowanie własne. Mapa cech przygotowana w oparciu o serwis Mapbox (Mapbox, 2023) i Open Street Map (OpenStreetMap contributors, 2023).

Rys. 73. Oznaczony czerwoną strzałką punkt wskazany przez sieć neuronową na ośrodku regulacyjnej Wisły na przedłużeniu ul. Ratuszowej. Punkt wydaje się dobrym miejscem na lokalizację reprezentacyjnego moła z widokiem na Stare Miasto. Opracowanie własne. Podkład przygotowany w oparciu o serwis Open Street Map (OpenStreetMap contributors, 2023).

Rys. 74. Punkty główne (biały) i osie naprowadzające (czerwony) odczytane przez sieć neuronową w obszarze założenia ujazdowskiego (po lewej) i założenia osi saskiej (po prawej). Kolorem zielonym oznaczone główne osie kompozycyjne. Sieć skutecznie oznaczyła większość założenia ujazdowskiego, jednak oś saska została prawie całkowicie pominięta. Opracowanie własne. Podkład przygotowany w oparciu o Open Street Map (OpenStreetMap contributors, 2023).

Rys. 75. Pięć obszarów przeanalizowanych statystycznie względem „stopnia skomponowania” przestrzeni. A: całość testowanego obszaru; B: okolice Wisły w rejonie mostu Śląsko-Dąbrowskiego; C: południowa granica Woli wzdłuż torów kolejowych; D: rejon osi saskiej; E: rejon założenia ujazdowskiego. Opracowanie własne. Podkład przygotowany w oparciu o Open Street Map (OpenStreetMap contributors, 2023).

Rys. 76. Liczba pikseli osi kompozycyjnych (czerwony) i liczba pikseli punktów głównych (czarny) rozpoznanych przez sieć dla wyszczególnionych obszarów jako procent liczby pikseli całych obszarów. Opracowanie własne.

Rys. 77. Diagram ilustrujący różnice między projektowaniem parametrycznym a projektowaniem hiperparametrycznym. W projektowaniu parametrycznym to architekt na podstawie odczytanego samodzielnie kontekstu decyduje o wartościach poszczególnych parametrów. Projektowanie hiperparametryczne charakteryzuje się wyższym poziomem abstrakcji. Architekt zainspirowany kontekstem definiuje hiperparametry modelu opartego na głębokim uczeniu maszynowym i dokonuje jego promptowania. Dodatkowo model także ma możliwość samodzielnego odczytywania kontekstów. Końcowe parametry definiowane są przez sieć neuronową na etapie inferencji. Wyniki inferencji mogą służyć architektowi jako inspiracja w przyszłych iteracjach. Opracowanie własne.

Rys. 78. Możliwe krzywe prognozujące rozwój wykorzystania głębokiego uczenia maszynowego w projektowaniu architektonicznym wspomagany komputerowo. Krzywa A pokazuje możliwość, że większość innowacji z wykorzystaniem NN w CAAD została już odkryta i zbliżamy się do możliwości technologicznych dalszego rozwoju technologii. Krzywa B dopuszcza możliwość, że rozwój dopiero się zaczął i pojawi się jeszcze wiele nowych narzędzi. Krzywa C opisuje sytuację, w której rozwój systemów AI doprowadzi do pojawienia się samouczącej się AGI, prowadzącej do wykładniczego rozwoju i transformacji całej ludzkości. Opracowanie własne na podstawie książek *The Empty Raincoat* (Handy, 1994) oraz *The Singularity is Near* (Kurzweil, 2005).

Rys. 79. Wizualizacja przygotowana przez Google, ilustrująca wzrost zdolności dużych modeli językowych wraz ze wzrostem ich wielkości na przykładzie modelu PALM. Im większa jest sieć neuronowa tym większe stają się jej kompetencje i tym bardziej zbliża się do poziomu AGI (Narang S i Chowdhery, 2022).

Aneks

Załącznik 1. Tabela źródeł kwerendy głównej wykorzystania sieci neuronowych w CAAD (do 2021 roku). Zastosowania.

Kodowana techniką multi-hot:

Publikacja					Zastosowanie					
Tytuł	Autorzy	Rok	Czasopismo/ Konferencja	Źródło danych	Eks- trakcja i uczy- telnia-nie	Ewalu- acja i prze-twa- rzanie da-nych	Klasy- fikacja	Prze- wydy- wanie przysz- łości	Opty- mali- zacja	Genero- wanie
PROCENT:				100%	15,4%	30,9%	17,4%	5,4%	8,7%	22,1%
SUMA:				149	23	46	26	8	13	33
Modelling spatial interaction using a neural net	Openshaw, S	1993	GISSSMaP	Tabele opisujące wzorce podróży do pracy w miastach amerykańskich i powiązane z nimi koszty	0	1	0	0	0	0
Using Artificial Neural Nets to Predict Building Energy Parameters	Stevenson, W J	1994	ASHRAE	Szeregi czasowe danych pomiarowych dotyczących wydajności energetycznej budynku	0	1	0	0	0	0
Artificial Neural Networks A New Approach to Modelling Interregional Telecommunication Flows	Fischer, M M; Gopal, S	1994	WSG	Dane historyczne o wykorzystaniu sieci telekomunikacyjnej	0	0	0	1	0	0
Artificial neural networks and naturally ventilated buildings	Kindangen, J I	1996	BRaI	Symulacja	0	0	0	0	1	0
Urban change detection based on artificial neural network	Liu, X; Lathrop Jr, R G	2002	UoRS	Zdjęcia satelitarne	1	0	0	0	0	0
Capturing Housing Market Segmentation An Alternative Approach based on Neural Network Modelling	Kauko, T; Hooimeijer, P; Hakfoort, J	2002	HS	10 zmiennych opisujących nieruchomości	0	0	1	0	0	0
Integration of neural networks and cellular automata for urban planning	Yeh, A G; Xia, L	2004	GSIS	Historyczne dane GIS	0	0	0	1	0	0
An Artificial Neural Network and Entropy Model for Residential Property Price Forecasting in Hong Kong	Lam, K C; Yu, C Y; Lam, K Y	2008	JoPR	Zmienne opisujące historycznie sprzedane budynki (liczba sypialni, lokalizacja, widok na morze, typ budynku etc)	0	1	0	0	0	0
An artificial neural network based approach for urban growth zonation in Dehradun city, India	Maithani, S; Arora, M K; Jain, R K	2010	GI	Zbiór map i zdjęć satelitarnych przetwarzanych przez systemy GIS	0	0	0	1	0	0
Service life prediction models for exterior stone cladding	Silva, A; Dias, J L; Gaspar, P L; de Brito, J	2011	BRaI	Zbiór danych opisujących elewację z paneli kamiennych	0	1	0	0	0	0
A neural network model to develop actions in urban complex systems represented by 2D meshes	Oliver, J L; Tortosa, L; Vicent, J F	2011	UoCM	Plan w formie sieci/grafu	0	0	0	0	1	0
Brains machines and buildings towards a neuromorphic architecture	Arbib, M A	2012	IBI	Przeglądowe/Inne	0	0	0	0	0	0

Optimizing artificial neural network-based indoor positioning system using genetic algorithm	Mehmood, H; Tripathi, N K	2013	IJoDE	Sygnal WiFi (RSSI z poszczególnych Access Pointów)	0	1	0	0	0	0
Neural networks applied to service life prediction of exterior painted surfaces	Dias, J L; Silva, A; Chai, C; Gaspar, P L, de Brito, J	2014	BRal	Zbiór danych opisujących zewnętrzną powierzchnię malowaną	0	1	0	0	0	0
Approximation of simulation-derived visual com...a comparative study in machine learning	Chatzikonstantinou, I; Sariyildiz, S	2015	ASR	Symulacja	0	1	0	0	0	0
Analysis of human mobility patterns from GPS trajectories and contextual information	Sila-Nowicka, K; Vandrol, J; Oshan, T; Long, J A; Demsar, U; Fotheringham, S	2016	IJoGIS	Trajektorie GPS	1	1	0	0	0	0
Artificial Imagination of Architecture with Deep Convolutional Neural Network	Silvestre, J; Ikeda, Y; Guena, F	2016	Computer Science	Kamera	0	1	0	0	0	1
Dimensionality Reduction for Parametric Design Exploration	Harding, J	2016	AiAG	Projekty parametryczne	1	0	0	0	0	0
Energy Model Machine EMM Instant Building Energy Prediction using Machine Learning	Asl, M R; Das, S; Tsai, B; Molloy, I; Hauck, A	2017	ECAADE	Symulacja	0	1	0	0	0	0
Modeling property values in Nigeria using artificial neural network	Abidoye, R B; Chan, A P C	2017	JoPR	Zmienne opisujące historycznie sprzedane budynki (liczba sypialni, lokalizacja, widok na morze, typ budynku etc)	0	1	0	0	0	0
Lace Wall Extending design intuition through machine learning	Tamke, M; Zwierzycki, M; Deleuran, A H; Baranovskaya, Y S	2017	FABRICATE	Złożony projekt parametryczny	0	0	1	0	0	0
3D Spatial Analysis Method with First-Person Viewpoint by Deep Convolutional Neural Network with Omnidirectional RGB and Depth Images	Takizawa, A; Furuta, A	2017	ECAADE	Zdjęcia panoramiczne (z symulacji modelu 3D) i opinie ludzi o "preferencji" przestrzeni	0	1	0	0	0	0
DANIEL A Deep Architecture for Automatic Analysis and Retrieval of Building Floor Plans	Sharma, D; Gupta, N; Chattopadhyay, C; Mehta, S	2017	ICoDAaR	Plany architektoniczne	1	0	0	0	0	0
Optical Integrity of Diminished Reality Using Deep Learning	Fukuda, T; Kuwamuro, Y; Yabuki, N	2017	ECAADE	Zdjęcia panoramiczne	0	0	1	0	0	0
Machines' Perception of Space Employing 3D Isovist Methods and a Convolutional Neural Network in Architectural Space Classification	Peng, W; Zhang, F; Nagakura, T	2017	MIT	Isovist z punktów wewnątrz i na zewnątrz budynku (pawilony parterowe)	0	0	1	0	0	0
Computing brains learning algorithms and neurocomputation in the smart city	Williamson, B	2017	ICaS	Przeglądowe/Inne	0	0	0	0	0	0
Machine learning for architectural design Practices and infrastructure	Tamke, M; Nicholas, P; Zwierzycki M	2018	IJAC	Przeglądowe/Inne	0	0	0	0	0	0

Learning Machine Learning as an Architect How to	Khean, N; Fabbri, A; Haeusler, M H	2018	ECAADE	Przeglądowe/Inne	0	0	0	0	0	0
The introspection of Deep Neural Networks - Towards Illuminating the black box	Khean, N; Kim, L; Martinez, J; Doherty, B; Fabbri, A; Gardner, N; Haeusler, H	2018	CAADRIA	Przeglądowe/Inne	0	0	0	0	0	0
Learning indoor space perception	Sedlmeier, A; Feld, S	2018	JoLBS	Isovist z punktów we wnętrzach	0	0	1	0	0	0
Classification based symbolic indoor positioning over the Miskolc IIS data set	Tamas, J; Toth, Z	2018	JoLBS	Dane hybrydowe nawigacyjne nagrane przy pomocy systemu ILONA (Indoor Localisation and Navigation); multimodalne z różnych sensorów w telefonie komórkowym	0	0	1	0	0	0
Applied Automatic Machine Learning Process for Material Computation	Luo, D; Wang, J; Xu, W	2018	ECAADE	Wartości przekrojów próbki elastomeru wyginanej przez ramię robotyczne, lub zadana forma geometryczna	0	1	0	0	0	0
Artificial Intelligence Aided Architectural Design	Cudzik, J; Radziszewski, K	2018	ECAADE	Depthmapy kapiteli kolumn korynckich	0	0	0	0	0	1
Artificial intelligence in architecture Generating conceptual design via deep learning	As, I; Pal, S; Basu, P	2018	IJAC	Rozkład funkcjonalny w formie grafu	0	0	0	0	0	1
Street Frontage Net Urban image classification using deep convolutional neural networks	Law, S; Seresinhe, C I; Shen, Y; Gutierrez-Roig, M	2018	IJoGIS	Zdjęcia panoramiczne	0	0	1	0	0	0
Multi Objective Qualitative Optimization MOQO in Architectural Design	Newton, D	2018	ECAADE	Symulacja (wygenerowane automatycznie modele wokselowe reprezentujące poszczególne cechy, jak rymiczność, heterogeniczność, horyzontalność etc)	0	0	0	0	1	0
CNN based generation of high accuracy urban distribution maps utilising SAR satellite imagery for short term change monitoring	Iino, S; Ito, R; Doi, K; Imaizumi, T; Hikosaka, S	2018	IJoIaDF	Zdjęcia satelitarne	1	0	0	0	0	0
Modelling A Complex Fabrication System New design tools for doubly curved metal surfaces fabricated using the English Wheel	Rossi, G; Nicholas, P	2018	ECAADE	Ręcznie generowane pary geometrii i instrukcji	0	1	0	0	0	0
Understanding and Visualizing Generative Adversarial Networks in Architectural Drawings	Zheng, H; Huang, W	2018	LPaA	Plany architektoniczne	0	0	1	0	0	0
An artificial intelligence-based method to efficiently bring CFD to building simulation	Mazurowski, W; Berger, J; Oliveira, R C L F; Mendes, N	2018	JoBPS	Symulacje (BPS: airflow, 3D heat exchange, HVAC systems simulation)	0	0	0	0	0	0
Design in THE Age of Artificial Intelligence	Leach, N	2018	LAF	Przeglądowe/Inne	0	0	0	0	0	0
Stochastic Hybrids From references to design options through Self Organizing Maps methodology	Algeciras-Rodriguez, J	2018	ECAADE	Modele w formie chmury punktów (wierzchołki poligonów)	0	0	0	0	0	1
Deep learning in design workflows The elusive design pixel	Mahankali, R; Johnson, B R; Anderson, A T	2018	IJAC	Modele BIM	0	0	0	0	0	0

Deep Form Finding Using Variational Autoencoders for deep form finding of structural typologies	de Miguel, J; Villafane, M E; Piskorec, L; Sancho-Caparrini, F	2019	ECAADE	Symulacja	0	1	0	0	0	0
A framework of developing machine learning models for facility life cycle cost analysis	Gao, X; Pishdad-Bozorgi, P	2019	BRaI	Modele BIM, dokumentacja projektowa, IWMS (integrated workplace management system), Dane o zużyciu mediów	0	1	0	0	0	0
Deep Reinforcement Learning for Autonomous Robotic Tensegrity ART	Hosmer, T; Tigas, P	2019	ACADIA	Symulacja	0	0	0	0	1	0
Alive A Multi Layered Flexible and Elastic Shape Aware Graphene Based Interface	Koshelyuk, D; Talaei, A; Garivani, S; Markopoulou, A; Chronis, A; Leon, D A; Krenmuller, R	2019	ACADIA	Wartości oporu elektrycznego (na wejściu) i geometria odkształcenia na wyjściu	0	1	0	0	0	0
Stripe Segmentation for Branching Shell Structures A Data Set Development as a Learning Process for Fabrication Efficiency and Structural Performance	Giannopoulou, E; Baquero, P; Warang, A; Orciuoli, A; Estevez, A T	2019	ECAADE	Projekty parametryczne (parametry na wejściu, cechy modelu na wyjściu zamiast czasochłonnego generowania modeli parametrycznych)	0	1	0	0	0	0
Re perceive 3D printing with Artificial Intelligence	Chen, D; Luo, D; Xu, W; Luo, C; Shen, L; Yan, X; Wang, T	2019	ECAADE	Szeregi czasowe (gcode) i rezultat w postaci krzywej wydrukowanej przez drukarkę 3d	0	1	0	0	0	0
Design Space Exploration of Initial Structural Design Alternatives via Artificial Neural Networks	Yetkin, O; Sorguç, A G	2019	ECAADE	Zbiór projektów parametrycznych kratownic na wejściu i wyniki symulacji przemieszczenia i masy struktury na wyjściu	0	0	0	0	1	1
Haptic Learning Towards Neural Network based adaptive Cobot Path Planning for unstructured spaces	Rossi, G; Nicholas, P	2019	ECAADE	Symulacja (trening sieci) + feedback haptyczny z rzeczywistego ramienia (finetuning)	0	0	0	0	1	0
Multimodal Classification of Urban Micro Events	Sukel, M; Rudinac, S; Worring, M	2019	ACMIC	Kamera/zgłoszenia tekstowe/koordynaty + Przeglądowe	1	0	0	0	0	0

Interdisciplinary AI A Machine Learning System for Streamlining External Aesthetic and Cultural Influences in Architecture	Özel, G; Ennemoser, B	2019	ACADIA	Kamera (zdjęcia budynków)	0	1	0	0	0	1
Imaginary Plans The potential of 2D to 2D Style transfer in planning processes	del Campo, M; Carlson, A; Manninger, S	2019	ACADIA	Plany architektoniczne	0	0	0	0	0	1
Generation of Floor Plan Variations with Convolutional Neural Networks and Case-based Reasoning	Eisenstadt, V; Langenhan, C; Althoff, K D	2019	ECAADE	Rozkład funkcjonalny w formie grafu	0	0	0	0	0	1
HorizonNet Learning Room Layout with 1D Representation and Pano Stretch Data Augmentation	Sun, C; Hsiao, C W, Sun, M; Chen, H T	2019	CVF	Zdjęcia panoramiczne (wnętrza)	1	1	0	0	0	0
Architectural Drawing Recognition A case study for training the learning algorithm with architectural plan and section drawing images	Çolakoğlu, M B; Uzun, C	2019	ECAADE	Plany i przekroje	0	0	1	0	0	0
Optimising Image Classification Implementation of Convolutional Neural Network Algorithms to Distinguish Between Plans and Sections	Ng, J M Y; Khean, N; Madden, D; Fabbri, A; Gardner, N; Haeusler, M H; Zavoleas, Y	2019	CAADRIA	Plany i przekroje	0	0	1	0	0	0
Semantic Enrichment of Indoor Point Clouds An Overview of Progress towards Digital Twinning	Stojanovic, V; Trapp, M; Richter, R; Hagedorn, B; Dollner, J	2019	ECAADE	Chmura punktów	0	0	1	0	0	0
Deep Learning Model for Predicting Preference of Space by Estimating the Depth Information of Space using Omnidirectional Images	Kinugawa, H; Takizawa, A	2019	ECAADE	Zdjęcia panoramiczne	0	1	0	0	0	0
Designing deep CNN models based on sparse coding for aerial imagery a deep features reduction approach	Qayyum, A; Malik, A; Saad, N M; Mazher, M	2019	EJoRS	Zdjęcia z drona	1	0	0	0	0	0
Deep Generative Learning for the Generation and Analysis of Architectural Plans with Small Datasets	Newton, D	2019	ECAADE	Plany architektoniczne	0	0	0	0	0	1
Hybrid Elevations Using GAN Networks	Mohammad, A; Beorkrem, C; Ellinger, J	2019	ACADIA	Parametrycznie wygenerowane elewacje, ale przetwarzane płaskie rysunki elewacji (czerwony)	0	0	0	0	0	1
An Anonymous Composition Design Optimization Through Machine Learning Algorithm	Liu, H; Liao, L; Srivastava, A	2019	ACADIA	Ręcznie kategoryzowane, losowo generowane projekty parametryczne względem "monumentalności" "delikatności" "dynamiczności" etc.	0	0	0	0	0	1
GAN Loci Imaging Place using Generative Adversarial Networks	Steinfeld, K	2019	ACADIA	Zdjęcia panoramiczne (street view)	1	0	0	0	0	0
Predicting and steering performance in architectural materials	Thomsen, M; Nicholas, P; Tamke, M; Gatz, S; Sinke, Y	2019	ECAADE	Docelowe zdjęcia zacielenia przez dzianą membranę, na wyjściu pliki do cyfrowej maszyny dziewiarskiej	0	1	0	0	0	0
Automated Brick Pattern Generator for Robotic Assembly using Machine Learning and Images	Zandavali, B A; Garcia, M J	2019	ECAADE	Symulacja (wygenerowane automatycznie obrysy i odpowiadające im ułożenie cegieł)	0	0	0	0	0	1

Steps towards AI augmented parametric modeling systems for supporting design exploration	Toulkeridou, V	2019	ECAADE	Projekty parametryczne przedstawiony jako szereg czasowy lub sekwencyjny diagram bloków operacji dynamo	0	0	0	0	0	1
Robot Ex Machina A Framework for Real Time Robot Programming and Control	del Castillo y López, J L G	2019	ACADIA	Ręcznie rysowane fragmenty szkiców i ich uzupełnienia	0	0	0	0	0	1
Pedestrian trajectory prediction using BiRNN encoder-decoder framework	Wu, J; Woo, H; Tamura, Y; Moro, A; Massaroli, S; Yamashita, A; Asama, H	2019	AR	Film z kamery	0	0	0	1	0	0
Quo vadis AI in Architecture Survey of the current possibilities of AI in the architectural practice	Mrosła, L; von Both, P	2019	ECAADE	Przeglądowe/Inne	0	0	0	0	0	0
Trend Topics and Changing Concepts of Computational Design in the Last 16 Years	Guzelci, O Z; Alacam, S; Guzelci, H	2019	ECAADE	Przeglądowe/Inne	0	0	0	0	0	0
Application of AI in Urban Design	Nematollahi, M A; Shahbazi, S; Nabian, N	2019	CVaA	Przeglądowe/Inne	0	0	0	0	0	0
Do Robots Dream of Digital Sheep	Leach, N	2019	ACADIA	Przeglądowe/Inne	0	0	0	0	0	0
Machine Learning from the Past	Zwierzycki, M	2019	AIAAF	Przeglądowe/Inne	0	0	0	0	0	0
Smart spaces, information processing and the question of intelligence	Lynch, C R; del Casino Jr., V J	2019	AotAAoG	Przeglądowe/Inne	0	0	0	0	0	0
Robotic Constraints Informed Design Process	Devadass, P; Heimig, T; Stumm, S; Kerber, E; Brell-Cokcan, S	2019	ACADIA	Symulacja	0	1	0	0	0	0
Space ALocation Techniques SAT Computable Design Problems and Integrated Framework of Solvers	Saha, N; Haymaker, J; Shelden, D	2020	ACADIA	Symulacja	0	0	0	0	0	1
Spatial Assembly with Self Play Reinforcement Learning	Hosmer, T; Tigas, P; Reeves, D; He, Z	2020	ACADIA	Symulacja	0	0	0	0	0	1
A Performance Based Urban Block Generative Design Using Deep Reinforcement Learning and Computer Vision	Han, Z; Yan, W; Liu, G	2020	CDRF	Symulacja	0	1	0	0	1	1
Applying Deep Learning and Databases for Energy-efficient Architectural Design	Singh, M M; Schneider-Marín, P; Harter, H; Lang, W; Geyer, P	2020	ECAADE	Symulacja	0	1	0	0	0	0
Machine Learning Methods in Energy Simulations for Architects and Designers	Sebestyen, A; Tyc, J	2020	ECAADE	Symulacja	0	1	0	0	0	0
A machine-learning model driven by geometry, material and structural performance data in architectural design process	Yazici, S	2020	ECAADE	Symulacja (structural performance, okształcenia, właściwości materiałów)	0	0	0	0	1	0
Optimization and Prediction of Design Variables Driven by Building Energy Performance A Case Study of Office Building in Wuhan	Li, J; Chen, H	2020	CDRF	Symulacja	0	0	0	0	1	0

Perceptive Machine Visuospatial Configurations Through Machine Intuition	Kampani, A; Varoudis, T	2020	ECAADE	Konglomerat danych miejskich	1	0	0	0	0	0
Pedestrian Flow Monitoring and Prediction through smart material sensing surfaces	Kirova, N; Markopoulou, A	2020	ACADIA	Sensory w płytach chodnikowych (trening na podstawie symulacji)	0	0	0	1	0	0
LOTI Using Machine Learning to simulate subjective opinions in design	Markusiewicz, J; Balerdi, A G	2020	ECAADE	Ręcznie oznaczone przez studentów modele parametryczne krzeseł (studenci oznaczali, czy dane krzesło jest plagiatem czy nie)	0	1	0	0	0	0
Generating and Optimizing a Funicular Arch Floor Structure	Zheng, H; Wang, X; Qi, Z; Sun, S; Akbarzadeh, M	2020	ACADIA	Symulacja	0	0	0	0	1	1
Towards Hallucinating Machines Designing with Computational Vision	del Campo, M; Carlson, A; Manninger, S	2020	IJAC	Kamera (sieć opata o google deep dream wytrenowana na imagenet)	0	0	0	0	0	1
How Machines Learn to Plan A Critical Interrogation of Machine Vision Techniques in Architecture	del Campo, M; Carlson, A; Manninger, S	2020	ACADIA	Plany budynków barokowych i współczesnych	0	0	0	0	0	1
Pipes of AI Machine Learning Assisted 3D Modeling Design	Liu, C; Shen, J; Ren, Y; Zheng, H	2020	CDRF	Obrys kondygnacji	0	0	0	0	0	1
Automatic Generation of the Schematic Mechanical System Drawing by Generative Adversarial Network	Sato, G; Ishizawa, T; Iseda, H; Kitahara, H	2020	ECAADE	Modele BIM (przerabiane na rastrowe plany architektoniczne)	0	0	0	0	0	1
An Academy of Spatial Agents Generating spatial configurations with deep reinforcement learning	Veloso, P; Krishnamurti, R	2020	ECAADE	Symulacja	0	0	0	0	0	1
Deep Learning surrogate models for spatial and visual connectivity	Tarabishy, S; Psarras, S; Kosicki, M; Tsigkari, M	2020	IJAC	Symulacja	0	1	0	0	0	0
3D Graph Convolutional Neural Networks in Architecture Design	del Campo, M; Carlson, A; Manninger, S	2020	ACADIA	Ręczny dataset "Sensibility" (parę tysięcy modeli OBJ przedstawiających poszczególne klasy architektoniczne: domy i kolumny), modelowane ręcznie, data augmentation by rozszerzyć dataset	0	1	1	0	0	0
The Emoting City Designing feeling and artificial empathy in mediated environments	Patel, S V; Tchakerian, R; Morais, R L; Zhang, J; Cropper, S	2020	ECAADE	Kamera	0	1	0	0	0	0

Occupancy-informed Introducing a method or flexible behavioural mapping in architecture using machine vision	Jørgensen, J; Tamke, M; Poulsgaard, K S	2020	ECAADE	Kamera w przestrzeni zbudowanej	1	0	0	0	0	0
A Large Scale Measurement and Quantitative Analysis Method of Facade Color in the Urban Street Using Deep Learning	Zhang, J; Fukuda, T; Yabuki, N	2020	CDRF	Zdjęcia panoramiczne (street view)	1	0	0	0	0	0
HierarchyNet Hierarchical CNN Based Urban Building Classification	Taoufiq, S; Nagy, B; Benedek, C	2020	Remote Sensing	Kamera (zdjęcia z perspektywy człowieka)	0	0	1	0	0	0
Drawing Recognition Integrating Machine Learning Systems into Architectural Design Workflows	Brown, L; Yip, M; Gardner, N; Haeusler, M H; Khean, N; Zavoleas, Y; Ramos, C	2020	ECAADE	Plany architektoniczne	0	0	1	0	0	0
Automatic Recognition and Segmentation of Architectural Elements	Xiao, Y; Chen, S; Ikeda, Y; Hotta, K	2020	CAADRIA	Plany architektoniczne	0	0	1	0	0	0
An AI Lens on Historic Cairo A Deep Learning Application for Minaret Classification	Zohier, I; El Antably, A; Madani, A S	2020	ACADIA	Kamera	0	0	1	0	0	0
Automatic Generation of Horizontal Building Mask Images by Using a 3D Model with Aerial Photographs for Deep Learning	Ikeno, K; Fukuda, T; Yabuki, N	2020	ECAADE	Zdjęcia satelitarne (+ generowane w VR z modelu 3D maski do treningu czyli symulacja)	1	0	1	0	0	0
Machine Learning for Comparative Urban Planning at Scale An Aviation Case Study	Meeran, A; Joyce, S C	2020	ACADIA	Zdjęcia satelitarne	1	0	1	0	0	0
Anxious Landscapes Correlating the Built Environment with Mental Health through Deep Learning	Newton, D	2020	ACADIA	Zdjęcia satelitarne, dane medyczne (wywiad o zaburzeniach nerwowych na badanym obszarze)	1	0	0	0	0	1
Deep Learning Methods for Urban Analysis and Health Estimation of Obesity	Newton, D; Piatkowski, D; Marshall, W; Tendle, A	2020	ECAADE	Zdjęcia satelitarne	1	0	0	0	0	0
Clustering and Morphological Analysis of Campus Context	Li, P; Zhu, W	2020	ACADIA	Zdjęcia satelitarne	1	0	0	0	0	0

Urban Detection Kit A System for Collection and Analysis of Street Level Imagery	Sukel, M; Rudinac, S; Worring, M	2020	ICMR	Kamera (wideo przymocowane do pojazdów służb miejskich)	1	0	0	0	0	0
Detecting Urban Issues With the Object Detection Kit	Sukel, M; Rudinac, S; Worring, M	2020	ACMIC	Kamera (wideo przymocowane do pojazdów służb miejskich)	1	0	0	0	0	0
A big data evaluation of urban street walkability using deep learning and environmental sensors	Li, Y; Yabuki, N; Fukuda, T; Zhang, J	2020	ECAADE	Konglomerat danych, GIS (open street maps + google maps), google street view, wideo z kamer, dane o hałasie i oświetleniu	0	1	1	0	0	0
An investigation of the visual features of urban street vitality using a convolutional neural network	Qi, Y; Drolma, S C; Zhang, X; Liang, J; Jiang, H; Xu, J; Ni, T	2020	Geo-spatial Information Science	Zdjęcia panoramiczne (street view)	0	1	0	0	0	0
Monitoring Asphalt Pavement Aging and Damage Conditions from Low Altitude UAV Imagery Based on a CNN Approach	Pan, Y; Chen, X; Sun, Q; Zhang, X	2020	CJoRS	Zdjęcia z drona	0	1	0	0	0	0
Making a New City Image	Ho, B	2020	ACADIA	Schwarzplan + kamera (zdjęcia z perspektywy człowieka)	0	0	1	0	0	0
Simulating urban land use change by integrating a convolutional neural network with vector-based cellular automata	Zhai, Y; Yao, Y; Guan, Q; Liang X; Li, X; Pan, Y; Yue, H; Yuan, Z; Zhou, J	2020	IJoGIS	Zbiór zmiennych GIS opisujących przestrzeń miasta (w formie map gradientowych)	0	0	1	0	0	0
City Wide Traffic Congestion Prediction Based On CNN LSTM and Transpose CNN	Ranjan, N; Bhandari, S; Zhao, H P; Kim, H; Khan, P	2020	IEEE Access	Szereg czasowy map ulic pokolorowanych względem zakorkowania (z google maps)	0	0	0	1	0	0
GPS Based citywide traffic congestion forecasting using CNN-RNN and C3D hybrid model	Guo, J; Liu, Y; Yang, Q K; Wang, Y	2020	TATS	Szereg czasowy trajektorii GPS	0	0	0	1	0	0
Suggestive Site Planning with Conditional GAN and Urban GIS Data	Tian, R	2020	CDRF	Dane GISowe przetworzone na obrazy (segmentacja funkcjonalna) do treningu	0	0	0	0	0	1
Text To Form 3D Prediction by Linguistic Descriptions	Zhang, H	2020	ACADIA	Tekst naturalny opisujący budynek	0	1	0	0	0	1
Towards a Distributed Robotically Assisted Construction Framework Using Reinforcement Learning	Fang, Z; Wu, Y; Hassonjee, A; Bidgoli, A; Cardoso-Llach, D	2020	ACADIA	Symulacja DRL	0	0	0	0	1	0

Machine Learning for Fabrication of Graded Knitted Membranes	Sinke, Y; Gatz, S; Tamke, M; Thomsen, M R	2020	CDRF	Chmury punktów: Skany 3D próbek poddanych obciążeniom (wejście), na wyjściu "pixel-based knit fabrication files" dla maszyny dziewiarskiej	0	1	0	0	1	0
Exploration of Campus Layout Based on Generative Adversarial Network Discussing the Significance of Small Amount Sample Learning for Architecture	Liu, Y; Luo, Y; Deng, Q; Zhou, X	2020	CDRF	Ręcznie oznaczane diagramy podziału funkcjonalnego kampusów uniwersyteckich	0	0	0	0	0	1
A Preliminary Study on the Formation of the General Layouts on the Northern Neighborhood Community Based on GauGAN Diversity Output Generator	Pan, Y; Qian, J; Hu, Y	2020	CDRF	Schwarzplany istniejących osiedli	0	0	0	0	0	1
Architectural Visualisation with Conditional Generative Adversarial Networks What machines read in architectural sketches	Chan, Y H E; Spaeth, A B	2020	ECAADE	Pary ręcznie rysowanych szkiców budynków i odpowiadających perspektyw architektonicznych (zdjęć i renderów)	0	1	0	0	0	0
Drawn Together Machine Augmented Sketching in the Design Studio	Steinfeld, K	2020	ACADIA	Ręcznie trenowane pary szkic i wizualizacja aksometryczna w formie uproszczonego diagarmu	0	1	0	0	0	0
DeepGreen Coupling Biological and Artificial Intelligence in Urban Design	Pasquero, C; Poletto, M	2020	ACADIA	Plany i mapy oraz fotografie próbek słuzowca	0	1	0	0	0	0
Machine Learning Aided 2D 3D Architectural Form Finding at High Resolution	Zhang, H; Huang, Y	2020	CDRF	Sekwencje kolejnych przekrojów przez budynek	0	1	0	0	0	0
BIM Hyperreality Data Synthesis Using BIM and Hyperrealistic Rendering for Deep Learning	Alawadhi, M; Yan, W	2020	ACADIA	Modele BIM i powiązane z nimi wizualizacje (trening); potem rozpoznawanie elementów budynku na zdjęciach (aplikacja)	0	0	1	0	0	0
A Machine Learning Method of Predicting Behavior Vitality Using Open Source Data	Sun, Y; Jiang, L; Zheng, H	2020	ACADIA	Mapy GiSowe i heatmapy aktywności fizycznej	0	1	0	0	0	0
Reprogramming Urban Block by Machine Creativity How to use neural networks as generative tools to design space	Yu, D	2020	ECAADE	Do treningu - plany i przekroje z ręcznie naniesionymi funkcjami, do inferencji same plany i przekroje	0	0	0	0	0	1
Encoded Images Representational Protocols for Integrating cGANs in Iterative Computational Design Processes	Rossi, G; Nicholas, P	2020	ACADIA	Wartości opisujące geometrię enkodowane w formie obrazów (do przetwarzania przez GAN)	0	1	0	0	0	0
Data Driven Midsole Performance Oriented Midsole Design Using Computational Multi Objective Optimization	Tian, R; Wang, T; Gün, O Y	2020	ACADIA	Szeregi czasowe, Dane dotyczące rozkładu obciążeń na podeszwie stopy podczas chodzenia i biegania (dataset ARAMIS)	0	1	0	0	0	0
On AI Adoption Issues in Architectural Design Identifying the issues based on an extensive literature review	Zwierzycki, M	2020	ECAADE	Przeglądowe/Inne	0	0	0	0	0	0
Steering into the Skid Arbitrating Human and Artificial Intelligences to	Kimm, G; Burry, M	2020	ACADIA	Przeglądowe/Inne	0	0	0	0	0	0

Augment the Design Process										
Urban Emotion The interrogation of social media and its implications within urban context	Kim, E; Rosenwasser, D	2020	ECAADE	Media społecznościowe	1	0	0	0	0	0
Reinforcement Learning for Sequential Assembly of SL Blocks	Wibranek, B; Liu, Y; Funk, N; Belousov, B; Peters, J; Tessmann, O	2021	ECAADE	Symulacja	0	0	0	0	1	1
Training Spaces Fostering machine sensibility for spatial assemblages through wave function collapse and reinforcement learning	Mintrone, A; Erioli, A	2021	ECAADE	Symulacja	0	0	0	0	0	1
Towards Abductive Reasoning Based Computational Design Tools Using Machine Learning as a way to explore the combined design spaces of multiple parametric models	Sebestyen, A; Rock, J; Hirschberg, U L	2021	ECAADE	Symulacja	0	1	0	0	0	0
A Chained Machine Learning Approach to Motivate Retro Cladding of Residential Buildings	Nicholas, P; Chen, Y; Borpujari, N; Bartov, N; Refsgaard, A	2021	ECAADE	Konglomerat danych: zdjęcia fasad, symulacje energetycznej budynków (na rzutach) i numeryczne (koszt ekonomiczny i wartość pochłoniętego przez panele CO2)	1	1	0	0	0	0
Approach to Auto Recognition of Human Trajectory in Squares using Machine Learning-Based Methods	Wu, S	2021	ECAADE	Filmy z drona	1	0	0	0	0	0
A Research On Building Cluster Morphology Formation Based On Wind Environmental Performance And Deep Reinforcement Learning	Song, Y; Yuan, P F	2021	ECAADE	Symulacja	0	0	0	0	0	1
Automatic Diminished Reality Based Virtual Demolition Method using Semantic Segmentation and Generative Adversarial Network for Landscape Assessment	Kikuchi, T; Fukuda, T; Yabuki, N	2021	ECAADE	Kamera (telefon)	0	0	1	0	0	0
Mixed Reality Landscape Visualization Method with Automatic Discrimination Process for Dynamic Occlusion Handling Using Instance Segmentation	Nakabayashi, M; Fukuda, T; Yabuki, N.	2021	ECAADE	Kamera (telefon/CCTV)	0	0	1	0	0	0
Comparative Evaluation of Tensor based Data Representations for Deep Learning Methods in Architecture	Eisenstadt, V; Arora, H; Ziegler, C; Bielski, J; Langenhan, C; Althoff, K D; Dengel, A	2021	ECAADE	Tensorowe reprezentacje semantyczne projektów architektonicznych	0	0	1	0	0	0
Visualizing Deep Learning Models for Urban Health Analysis	Newton, D	2021	ECAADE	Zdjęcia satelitarne i dane o zdrowiu	1	0	0	0	0	0

From Streetscape to Data Semantic segmentation for the prediction of outdoor thermal comfort	Yazicioglu, G; Dino, I G	2021	ECAADE	Zdjęcia panoramiczne street view + symulacje nasłonecznienia w ladybug do treningu	0	1	0	0	0	0
A deep 2d 3d Feature Level fusion for classification of UAV multispectral imagery in urban areas	Pourazar, H; Samadzadegan, F; Javan, F D	2021	Geocarto International	Zdjęcia z drona, różne spektra światła widzialnego i podczerwonego	0	0	1	0	0	0
A novel CNN LSTM based approach to predict urban expansion	Boulila, W; Ghandorh, H; Khan, M A; Ahmed, F; Ahmad, J	2021	Ecological Informatics	Szereg czasowy zdjęć satelitarnych	0	0	0	1	0	0
Fill In The Blanks Deep Convolutional Generative Adversarial Network to Investigate the Virtual Design Space of Historical Islamic Patterns	Alani, M W; Al-Kaseem, B R	2021	ASCAAD	Przekroje przez rozety	0	0	0	0	0	1
Limits to Applied ML in Planning and Architecture Understanding and defining extents and capabilities	Joyce, S C; Nazim, I	2021	ECAADE	Przeglądowe/Inne	0	0	0	0	0	0

Załącznik 2. Tabela źródeł kwerendy głównej wykorzystania sieci neuronowych w CAAD (do 2021 roku). Rodzaj sieci neuronowej, skala zastosowania.

Kodowana techniką multi-hot:

Publikacja	Rodzaj sieci neuronowej								Skala zastosowania				
Tytuł	FNN i VAE	CNN	RNN	LSTM	GAN	TRANS-FORMER ORAZ NLP	INNE NN, np. SOM	DRL	WPOMAGAJĄCO DODATKOWE ML	DETAL, MATERIAŁY, CAM	ARCHITEKTURA	URBANISTYKA	BEZSKALOWE
PROCENT:	32,9%	41,6%	4,7%	6,0%	20,1%	2,7%	3,4%	6,0%	14,8%	25,5%	49,0%	39,6%	2,7%
SUMA:	49	62	7	9	30	4	5	9	22	38	73	59	4
Modelling spatial interaction using a neural net	1	0	0	0	0	0	0	0	0	0	0	1	0
Using Artificial Neural Nets to Predict Building Energy Parameters	1	0	0	0	0	0	0	0	0	0	1	0	0
Artificial Neural Networks A New Approach to Modelling Interregional Telecommunication Flows	1	0	0	0	0	0	0	0	0	0	0	1	0
Artificial neural networks and naturally ventilated buildings	1	0	0	0	0	0	0	0	0	0	1	0	0
Urban change detection based on artificial neural network	1	0	0	0	0	0	0	0	1	0	0	1	0
Capturing Housing Market Segmentation An Alternative Approach based on Neural Network Modelling	0	0	0	0	0	0	1	0	0	0	0	1	0

Integration of neural networks and cellular automata for urban planning	1	0	0	0	0	0	0	0	1	0	0	1	0
An Artificial Neural Network and Entropy Model for Residential Property Price Forecasting in Hong Kong	1	0	0	0	0	0	0	0	1	0	0	1	0
An artificial neural network based approach for urban growth zonation in Dehradun city, India	1	0	0	0	0	0	0	0	0	0	0	1	0
Service life prediction models for exterior stone cladding	1	0	0	0	0	0	0	0	1	1	0	0	0
A neural network model to develop actions in urban complex systems represented by 2D meshes	0	0	0	0	0	0	1	0	0	0	0	1	0
Brains machines and buildings towards a neuromorphic architecture	1	0	0	0	0	0	0	0	0	1	1	0	0
Optimizing artificial neural network-based indoor positioning system using genetic algorithm	1	0	0	0	0	0	0	0	0	0	1	0	0
Neural networks applied to service life prediction of exterior painted surfaces	1	0	0	0	0	0	0	0	0	1	0	0	0
Approximation of simulation-derived visual com...a comparative study in machine learning	1	0	0	0	0	0	0	0	1	0	1	0	0
Analysis of human mobility patterns from GPS trajectories and contextual information	1	0	0	0	0	0	0	0	0	0	0	1	0
Artificial Imagination of Architecture with Deep Convolutional Neural Network	0	1	0	0	1	0	0	0	0	0	1	0	0
Dimensionality Reduction for Parametric Design Exploration	0	0	0	0	0	0	1	0	1	0	1	0	0
Energy Model Machine EMM Instant Building Energy Prediction using Machine Learning	1	0	0	0	0	0	0	0	1	0	1	0	0
Modeling property values in Nigeria using artificial neural network	1	0	0	0	0	0	0	0	0	0	0	1	0
Lace Wall Extending design intuition through machine learning	1	0	0	0	0	0	0	0	0	1	0	0	0

3D Spatial Analysis Method with First-Person Viewpoint by Deep Convolutional Neural Network with Omnidirectional RGB and Depth Images	0	1	0	0	0	0	0	0	0	0	1	1	0
DANIEL A Deep Architecture for Automatic Analysis and Retrieval of Building Floor Plans	0	1	0	0	0	0	0	0	0	0	1	0	0
Optical Integrity of Diminished Reality Using Deep Learning	0	1	0	0	0	0	0	0	0	0	1	1	0
Machines' Perception of Space Employing 3D Isosvist Methods and a Convolutional Neural Network in Architectural Space Classification	0	1	0	0	0	0	0	0	0	0	1	0	0
Computing brains learning algorithms and neurocomputation in the smart city	0	0	0	0	0	0	0	0	0	0	0	1	0
Machine learning for architectural design Practices and infrastructure	1	0	0	0	0	0	0	0	0	1	0	0	0
Learning Machine Learning as an Architect How to	1	0	0	0	0	0	0	0	0	0	0	0	1
The introspection of Deep Neural Networks - Towards Illuminating the black box	1	0	0	0	0	0	0	0	0	0	0	0	1
Learning indoor space perception	1	0	0	0	0	0	0	0	1	0	1	0	0
Classification based symbolic indoor positioning over the Miskolc IIS data set	1	0	0	0	0	0	0	0	1	0	1	0	0
Applied Automatic Machine Learning Process for Material Computation	1	0	1	1	0	0	0	0	0	1	0	0	0
Artificial Intelligence Aided Architectural Design	1	0	0	0	0	0	0	0	0	1	0	0	0
Artificial intelligence in architecture Generating conceptual design via deep learning	0	1	0	0	1	0	0	0	0	0	1	0	0
Street Frontage Net Urban image classification using deep convolutional neural networks	0	1	0	0	0	0	0	0	0	0	1	1	0
Multi Objective Qualitative Optimization MOQO in Architectural Design	0	1	0	0	0	0	0	0	0	0	1	0	0
CNN based generation of high accuracy urban distribution maps utilising SAR satellite imagery for short term change monitoring	0	1	0	0	0	0	0	0	0	0	0	1	0
Modelling A Complex Fabrication System New design tools for doubly curved metal surfaces fabricated using the English Wheel	0	1	0	0	0	0	0	0	0	1	0	0	0
Understanding and Visualizing Generative Adversarial Networks in Architectural Drawings	0	0	0	0	1	0	0	0	0	0	1	0	0

An artificial intelligence-based method to efficiently bring CFD to building simulation	0	0	1	0	0	0	0	0	0	0	1	0	0
Design in THE Age of Artificial Intelligence	0	0	0	0	0	0	0	0	0	0	0	0	0
Stochastic Hybrids From references to design options through Self Organizing Maps methodology	0	0	0	0	0	0	1	0	1	1	1	0	0
Deep learning in design workflows The elusive design pixel	0	0	0	0	0	1	0	0	0	0	1	0	0
Deep Form Finding Using Variational Autoencoders for deep form finding of structural typologies	1	0	0	0	0	0	0	0	0	0	1	0	0
A framework of developing machine learning models for facility life cycle cost analysis	1	0	0	0	0	0	0	0	1	0	1	1	0
Deep Reinforcement Learning for Autonomous Robotic Tensegrity ART	1	0	0	0	0	0	0	1	0	1	1	0	0
Alive A Multi Layered Flexible and Elastic Shape Aware Graphene Based Interface	1	0	0	0	0	0	0	0	0	1	0	0	0
Stripe Segmentation for Branching Shell Structures A Data Set Development as a Learning Process for Fabrication Efficiency and Structural Performance	1	0	0	0	0	0	0	0	0	1	0	0	0
Re perceive 3D printing with Artificial Intelligence	1	1	0	1	0	0	0	0	1	1	0	0	0
Design Space Exploration of Initial Structural Design Alternatives via Artificial Neural Networks	1	0	0	0	0	0	0	0	0	1	0	0	0

Haptic Learning Towards Neural Network based adaptive Cobot Path Planning for unstructured spaces	1	0	0	0	0	0	0	0	0	1	0	0	0
Multimodal Classification of Urban Micro Events	0	1	0	1	0	0	0	0	1	0	0	1	0
Interdisciplinary AI A Machine Learning System for Streamlining External Aesthetic and Cultural Influences in Architecture	0	1	0	0	0	0	0	0	0	0	1	0	0
Imaginary Plans The potential of 2D to 2D Style transfer in planning processes	0	1	0	0	1	0	0	0	0	0	1	0	0
Generation of Floor Plan Variations with Convolutional Neural Networks and Case-based Reasoning	0	1	0	0	1	0	0	0	0	0	1	0	0
HorizonNet Learning Room Layout with 1D Representation and Pano Stretch Data Augmentation	0	1	1	1	0	0	0	0	0	0	1	0	0
Architectural Drawing Recognition A case study for training the learning algorithm with architectural plan and section drawing images	0	1	0	0	0	0	0	0	0	0	1	0	0
Optimising Image Classification Implementation of Convolutional Neural Network Algorithms to Distinguish Between Plans and Sections	0	1	0	0	0	0	0	0	0	1	1	0	0
Semantic Enrichment of Indoor Point Clouds An Overview of Progress towards Digital Twinning	0	1	0	0	0	0	0	0	1	0	1	0	0
Deep Learning Model for Predicting Preference of Space by Estimating the Depth Information of Space using Omnidirectional Images	0	1	0	0	1	0	0	0	0	0	0	1	0
Designing deep CNN models based on sparse coding for aerial imagery a deep features reduction approach	0	1	0	0	0	0	0	0	0	0	0	1	0
Deep Generative Learning for the Generation and Analysis of Architectural Plans with Small Datasets	0	0	0	0	1	0	0	0	0	0	1	0	0
Hybrid Elevations Using GAN Networks	0	0	0	0	1	0	0	0	0	0	1	0	0
An Anonymous Composition Design Optimization Through Machine Learning Algorithm	0	0	0	0	1	0	0	0	0	0	1	0	0

GAN Loci Imaging Place using Generative Adversarial Networks	0	0	0	0	1	0	0	0	0	0	0	1	0
Predicting and steering performance in architectural materials	0	0	0	0	1	0	0	0	0	1	0	0	0
Automated Brick Pattern Generator for Robotic Assembly using Machine Learning and Images	0	0	0	0	1	0	0	0	0	1	0	0	0
Steps towards AI augmented parametric modeling systems for supporting design exploration	0	0	1	1	0	1	0	0	0	0	1	1	0
Robot Ex Machina A Framework for Real Time Robot Programming and Control	0	0	1	0	0	0	0	0	0	1	0	0	0
Pedestrian trajectory prediction using BiRNN encoder-decoder framework	0	0	1	0	0	0	0	0	0	1	0	0	0
Quo vadis AI in Architecture Survey of the current possibilities of AI in the architectural practice	0	0	0	0	0	0	0	0	0	1	1	1	1
Trend Topics and Changing Concepts of Computational Design in the Last 16 Years	0	0	0	0	0	0	0	0	0	1	1	1	1
Application of AI in Urban Design	0	0	0	0	0	0	0	0	0	0	0	1	0
Do Robots Dream of Digital Sheep	0	0	0	0	0	0	0	0	0	0	0	0	0
Machine Learning from the Past	0	0	0	0	0	0	0	0	0	0	0	0	0
Smart spaces, information processing and the question of intelligence	0	0	0	0	0	0	0	0	0	0	0	0	0
Robotic Constraints Informed Design Process	0	0	0	0	0	0	1	0	1	1	0	0	0
Space Allocation Techniques SAT Computable Design Problems and Integrated Framework of Solvers	1	0	0	0	0	0	0	1	0	0	1	1	0
Spatial Assembly with Self Play Reinforcement Learning	1	0	0	0	0	0	0	1	0	0	1	0	0
A Performance Based Urban Block Generative Design Using Deep Reinforcement Learning and Computer Vision	1	0	0	0	0	0	0	1	0	0	1	1	0
Applying Deep Learning and Databases for Energy-efficient Architectural Design	1	0	0	0	0	0	0	0	0	0	1	0	0
Machine Learning Methods in Energy Simulations for Architects and Designers	1	0	0	0	0	0	0	0	0	1	1	0	0

A machine-learning model driven by geometry, material and structural performance data in architectural design process	1	0	0	0	0	0	0	0	1	1	1	0	0
Optimization and Prediction of Design Variables Driven by Building Energy Performance A Case Study of Office Building in Wuhan	1	0	0	0	0	0	0	0	0	0	1	0	0
Perceptive Machine Visuospatial Configurations Through Machine Intuition	1	0	0	0	0	0	0	0	1	0	0	1	0
Pedestrian Flow Monitoring and Prediction through smart material sensing surfaces	1	0	0	0	0	0	0	0	0	1	0	1	0
LOTI Using Machine Learning to simulate subjective opinions in design	1	0	0	0	0	0	0	0	0	1	0	0	0
Generating and Optimizing a Funicular Arch Floor Structure	1	0	0	0	0	0	0	0	0	1	0	0	0
Towards Hallucinating Machines Designing with Computational Vision	0	1	0	0	0	0	0	0	0	1	1	0	0
How Machines Learn to Plan A Critical Interrogation of Machine Vision Techniques in Architecture	0	1	0	0	1	0	0	0	0	0	1	0	0
Pipes of AI Machine Learning Assisted 3D Modeling Design	0	1	0	0	0	0	0	0	0	0	1	0	0
Automatic Generation of the Schematic Mechanical System Drawing by Generative Adversarial Network	0	1	0	0	1	0	0	0	0	0	1	0	0
An Academy of Spatial Agents Generating spatial configurations with deep reinforcement learning	0	1	0	0	0	0	0	1	0	0	1	0	0
Deep Learning surrogate models for spatial and visual connectivity	0	1	0	0	1	0	0	0	0	0	1	0	0

3D Graph Convolutional Neural Networks in Architecture Design	0	1	0	0	0	0	0	0	0	0	1	0	0
The Emoting City Designing feeling and artificial empathy in mediated environments	0	1	0	0	0	0	0	0	0	1	1	1	0
Occupancy-informed Introducing a method or flexible behavioural mapping in architecture using machine vision	0	1	0	0	0	0	0	0	0	0	1	0	0
A Large Scale Measurement and Quantitative Analysis Method of Facade Color in the Urban Street Using Deep Learning	0	1	0	0	0	0	0	0	0	0	1	1	0
HierarchyNet Hierarchical CNN Based Urban Building Classification	0	1	0	0	0	0	0	0	0	0	1	0	0
Drawing Recognition Integrating Machine Learning Systems into Architectural Design Workflows	0	1	0	0	0	0	0	0	0	0	1	0	0
Automatic Recognition and Segmentation of Architectural Elements	0	1	0	0	0	0	0	0	0	0	1	0	0
An AI Lens on Historic Cairo A Deep Learning Application for Minaret Classification	0	1	0	0	0	0	0	0	0	0	1	0	0
Automatic Generation of Horizontal Building Mask Images by Using a 3D Model with Aerial Photographs for Deep Learning	0	1	0	0	0	0	0	0	0	0	0	1	0
Machine Learning for Comparative Urban Planning at Scale An Aviation Case Study	0	1	0	0	0	0	0	0	0	0	0	1	0
Anxious Landscapes Correlating the Built Environment with Mental Health through Deep Learning	0	1	0	0	1	0	0	0	0	0	0	1	0

Deep Learning Methods for Urban Analysis and Health Estimation of Obesity	0	1	0	0	0	0	0	0	0	0	0	1	0
Clustering and Morphological Analysis of Campus Context	0	1	0	0	0	0	0	0	0	0	0	1	0
Urban Detection Kit A System for Collection and Analysis of Street Level Imagery	0	1	0	0	0	0	0	0	1	0	0	1	0
Detecting Urban Issues With the Object Detection Kit	0	1	0	0	0	0	0	0	1	0	0	1	0
A big data evaluation of urban street walkability using deep learning and environmental sensors	0	1	0	0	0	0	0	0	0	0	0	1	0
An investigation of the visual features of urban street vitality using a convolutional neural network	0	1	0	0	0	0	0	0	0	0	0	1	0
Monitoring Asphalt Pavement Aging and Damage Conditions from Low Altitude UAV Imagery Based on a CNN Approach	0	1	0	0	0	0	0	0	1	1	0	1	0
Making a New City Image	0	1	0	0	0	0	0	0	0	0	0	1	0
Simulating urban land use change by integrating a convolutional neural network with vector-based cellular automata	0	1	0	0	0	0	0	0	1	0	0	1	0
City Wide Traffic Congestion Prediction Based On CNN LSTM and Transpose CNN	0	1	0	1	0	0	0	0	0	0	0	1	0
GPS Based citywide traffic congestion forecasting using CNN-RNN and C3D hybrid model	0	1	1	1	0	0	0	0	0	0	0	1	0
Suggestive Site Planning with Conditional GAN and Urban GIS Data	0	1	0	0	1	0	0	0	0	0	0	1	0

Text To Form 3D Prediction by Linguistic Descriptions	0	1	0	0	0	1	0	0	0	0	1	0	0
Towards a Distributed Robotically Assisted Construction Framework Using Reinforcement Learning	0	1	0	0	0	0	0	1	0	1	0	0	0
Machine Learning for Fabrication of Graded Knitted Membranes	0	1	0	0	0	0	0	0	0	1	0	0	0
Exploration of Campus Layout Based on Generative Adversarial Network Discussing the Significance of Small Amount Sample Learning for Architecture	0	0	0	0	1	0	0	0	0	0	1	1	0
A Preliminary Study on the Formation of the General Layouts on the Northern Neighborhood Community Based on GauGAN Diversity Output Generator	0	0	0	0	1	0	0	0	0	0	1	1	0
Architectural Visualisation with Conditional Generative Adversarial Networks What machines read in architectural sketches	0	0	0	0	1	0	0	0	0	0	1	0	0
Drawn Together Machine Augmented Sketching in the Design Studio	0	0	0	0	1	0	0	0	0	0	1	0	0
DeepGreen Coupling Biological and Artificial Intelligence in Urban Design	0	0	0	0	1	0	0	0	0	0	1	1	0
Machine Learning Aided 2D 3D Architectural Form Finding at High Resolution	0	0	0	0	1	0	0	0	0	0	1	0	0
BIM Hyperreality Data Synthesis Using BIM and Hyperrealistic Rendering for Deep Learning	0	0	0	0	1	0	0	0	0	0	1	0	0
A Machine Learning Method of Predicting Behavior Vitality Using Open Source Data	0	0	0	0	1	0	0	0	0	0	0	1	0
Reprogramming Urban Block by Machine Creativity How to use neural networks as generative tools to design space	0	0	0	0	1	0	0	0	0	0	0	1	0
Encoded Images Representational Protocols for Integrating cGANs in Iterative Computational Design Processes	0	0	0	0	1	0	0	0	0	1	0	0	0

Data Driven Midsole Performance Oriented Midsole Design Using Computational Multi Objective Optimization	0	0	0	1	0	0	0	0	0	1	0	0	0
On AI Adoption Issues in Architectural Design Identifying the issues based on an extensive literature review	0	0	0	0	0	0	0	0	0	0	0	0	0
Steering into the Skid Arbitrating Human and Artificial Intelligences to Augment the Design Process	0	0	0	0	0	0	0	0	0	0	0	0	0
Urban Emotion The interrogation of social media and its implications within urban context	0	0	0	0	0	1	0	0	0	0	0	1	0
Reinforcement Learning for Sequential Assembly of SL Blocks	1	0	0	0	0	0	0	1	0	0	1	0	0
Training Spaces Fostering machine sensibility for spatial assemblages through wave function collapse and reinforcement learning	1	0	0	0	0	0	0	1	0	0	1	0	0
Towards Abductive Reasoning Based Computational Design Tools Using Machine Learning as a way to explore the combined design spaces of multiple parametric models	1	0	0	0	0	0	0	0	0	0	1	0	0
A Chained Machine Learning Approach to Motivate Retro Cladding of Residential Buildings	1	0	0	0	1	0	0	0	0	1	1	0	0
Approach to Auto Recognition of Human Trajectory in Squares using Machine Learning-Based Methods	1	1	0	0	0	0	0	0	1	0	0	1	0
A Research On Building Cluster Morphology Formation Based On Wind Environmental Performance And Deep Reinforcement Learning	1	0	0	0	0	0	0	1	0	0	0	1	0
Automatic Diminished Reality Based Virtual Demolition Method using Semantic Segmentation and Generative Adversarial Network for Landscape Assessment	0	1	0	0	1	0	0	0	0	0	1	1	0
Mixed Reality Landscape Visualization Method with Automatic Discrimination Process for Dynamic Occlusion Handling Using Instance Segmentation	0	1	0	0	0	0	0	0	0	0	1	1	0
Comparative Evaluation of Tensor based Data Representations for Deep Learning Methods in Architecture	0	1	0	0	0	0	0	0	0	0	1	0	0

Visualizing Deep Learning Models for Urban Health Analysis	0	1	0	0	0	0	0	0	0	0	0	1	0
From Streetscape to Data Semantic segmentation for the prediction of outdoor thermal comfort	0	1	0	0	0	0	0	0	0	0	0	1	0
A deep 2d 3d Feature Level fusion for classification of UAV multispectral imagery in urban areas	0	1	0	0	0	0	0	0	0	0	0	1	0
A novel CNN LSTM based approach to predict urban expansion	0	1	0	1	0	0	0	0	0	0	0	1	0
Fill In The Blanks Deep Convolutional Generative Adversarial Network to Investigate the Virtual Design Space of Historical Islamic Patterns	0	1	0	0	1	0	0	0	0	1	0	0	0
Limits to Applied ML in Planning and Architecture Understanding and defining extents and capabilities	0	0	0	0	0	0	0	0	0	1	0	0	0

Załącznik 3. Tabela źródeł kwerendy głównej wykorzystania sieci neuronowych w CAAD (do 2021 roku). Tematyka, źródła danych.

Kodowana techniką multi-hot:

	Tematyka		Źródła danych													
Tytuł	Morfolo- gia	Kom- pozycja (wspom- niana pośred- nio)	Wartości tabula- ryczne, liczbo- we, chmury punktów	Pro-jekt para- met- ryczny	Tekst	BIM	GIS i po- chodne	Dane mul-ti- mo-dalne	Ręcznie ozna- czane dane	Kla- syczne plany i prze- kroje	Szereg Cza- sowy	Zdjęcia sateli- tarne i lot- nicze	Symu- lacja	Ka- mera	Kamera panora- miczna / sfe- ryczna	Przeg- ładowe / Inne
PROCENT:	18,1%	8,7%	11,4%	6,7%	1,3%	2,7%	4,0%	3,4%	6,0%	8,1%	4,0%	9,4%	18,1%	9,4%	5,4%	10,1%
SUMA:	27	13	17	10	2	4	6	5	9	12	6	14	27	14	8	15
Modelling spatial interaction using a neural net	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Using Artificial Neural Nets to Predict Building Energy Parameters	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
Artificial Neural Networks A New Approach to Modelling Interregional Telecommunication Flows	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Artificial neural networks and naturally ventilated buildings	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
Urban change detection based on artificial neural network	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0

Capturing Housing Market Segmentation An Alternative Approach based on Neural Network Modelling	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Integration of neural networks and cellular automata for urban planning	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
An Artificial Neural Network and Entropy Model for Residential Property Price Forecasting in Hong Kong	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
An artificial neural network based approach for urban growth zonation in Dehradun city, India	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
Service life prediction models for exterior stone cladding	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
A neural network model to develop actions in urban complex systems represented by 2D meshes	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
Brains machines and buildings towards a neuromorphic architecture	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
Optimizing artificial neural network-based indoor positioning system using genetic algorithm	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Neural networks applied to service life prediction of exterior painted surfaces	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Approximation of simulation-derived visual com...a comparative study in machine learning	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
Analysis of human mobility patterns from GPS trajectories and contextual information	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
Artificial Imagination of Architecture with Deep Convolutional Neural Network	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
Dimensionality Reduction for Parametric Design Exploration	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
Energy Model Machine EMM Instant Building Energy Prediction using Machine Learning	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0

Modeling property values in Nigeria using artificial neural network	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Lace Wall Extending design intuition through machine learning	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
3D Spatial Analysis Method with First-Person Viewpoint by Deep Convolutional Neural Network with Omnidirectional RGB and Depth Images	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
DANIEL A Deep Architecture for Automatic Analysis and Retrieval of Building Floor Plans	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
Optical Integrity of Diminished Reality Using Deep Learning	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
Machines' Perception of Space Employing 3D Isovist Methods and a Convolutional Neural Network in Architectural Space Classification	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Computing brains learning algorithms and neurocomputation in the smart city	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
Machine learning for architectural design Practices and infrastructure	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
Learning Machine Learning as an Architect How to	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
The introspection of Deep Neural Networks - Towards Illuminating the black box	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
Learning indoor space perception	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Classification based symbolic indoor positioning over the Miskolc IIS data set	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
Applied Automatic Machine Learning Process for Material Computation	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Artificial Intelligence Aided Architectural Design	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Artificial intelligence in architecture Generating conceptual design via deep learning	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0

Street Frontage Net Urban image classification using deep convolutional neural networks	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
Multi Objective Qualitative Optimization MOQO in Architectural Design	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
CNN based generation of high accuracy urban distribution maps utilising SAR satellite imagery for short term change monitoring	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
Modelling A Complex Fabrication System New design tools for doubly curved metal surfaces fabricated using the English Wheel	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
Understanding and Visualizing Generative Adversarial Networks in Architectural Drawings	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
An artificial intelligence-based method to efficiently bring CFD to building simulation	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
Design in THE Age of Artificial Intelligence	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
Stochastic Hybrids From references to design options through Self Organizing Maps methodology	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Deep learning in design workflows The elusive design pixel	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
Deep Form Finding Using Variational Autoencoders for deep form finding of structural typologies	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0
A framework of developing machine learning models for facility life cycle cost analysis	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
Deep Reinforcement Learning for Autonomous Robotic Tensegrity ART	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
Alive A Multi Layered Flexible and Elastic Shape Aware Graphene Based Interface	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0

Stripe Segmentation for Branching Shell Structures A Data Set Development as a Learning Process for Fabrication Efficiency and Structural Performance	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
Re perceive 3D printing with Artificial Intelligence	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
Design Space Exploration of Initial Structural Design Alternatives via Artificial Neural Networks	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
Haptic Learning Towards Neural Network based adaptive Cobot Path Planning for unstructured spaces	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
Multimodal Classification of Urban Micro Events	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
Interdisciplinary AI A Machine Learning System for Streamlining External Aesthetic and Cultural Influences in Architecture	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
Imaginary Plans The potential of 2D to 2D Style transfer in planning processes	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
Generation of Floor Plan Variations with Convolutional Neural Networks and Case-based Reasoning	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0
HorizonNet Learning Room Layout with 1D Representation and Pano Stretch Data Augmentation	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
Architectural Drawing Recognition A case study for training the learning algorithm with architectural plan and section drawing images	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0

Optimising Image Classification Implementation of Convolutional Neural Network Algorithms to Distinguish Between Plans and Sections	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
Semantic Enrichment of Indoor Point Clouds An Overview of Progress towards Digital Twinning	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Deep Learning Model for Predicting Preference of Space by Estimating the Depth Information of Space using Omnidirectional Images	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
Designing deep CNN models based on sparse coding for aerial imagery a deep features reduction approach	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
Deep Generative Learning for the Generation and Analysis of Architectural Plans with Small Datasets	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0
Hybrid Elevations Using GAN Networks	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
An Anonymous Composition Design Optimization Through Machine Learning Algorithm	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0
GAN Loci Imaging Place using Generative Adversarial Networks	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
Predicting and steering performance in architectural materials	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
Automated Brick Pattern Generator for Robotic Assembly using Machine Learning and Images	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
Steps towards AI augmented parametric modeling systems for supporting design exploration	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
Robot Ex Machina A Framework for Real Time Robot Programming and Control	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
Pedestrian trajectory prediction using BiRNN encoder-decoder framework	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
Quo vadis AI in Architecture Survey of the current possibilities of AI in the architectural practice	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Trend Topics and Changing Concepts of Computational Design in the Last 16 Years	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
Application of AI in Urban Design	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
Do Robots Dream of Digital Sheep	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
Machine Learning from the Past	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
Smart spaces, information processing and the question of intelligence	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
Robotic Constraints Informed Design Process	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
Space Allocation Techniques SAT Computable Design Problems and Integrated Framework of Solvers	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
Spatial Assembly with Self Play Reinforcement Learning	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0
A Performance Based Urban Block Generative Design Using Deep Reinforcement Learning and Computer Vision	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
Applying Deep Learning and Databases for Energy-efficient Architectural Design	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
Machine Learning Methods in Energy Simulations for Architects and Designers	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
A machine-learning model driven by geometry, material and structural performance data in architectural design process	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
Optimization and Prediction of Design Variables Driven by Building Energy Performance A Case Study of Office Building in Wuhan	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
Perceptive Machine Visuospatial Configurations Through Machine Intuition	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
Pedestrian Flow Monitoring and Prediction through smart material sensing surfaces	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
LOTI Using Machine Learning to simulate subjective opinions in design	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
Generating and Optimizing a Funicular Arch Floor Structure	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0

Towards Hallucinating Machines Designing with Computational Vision	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
How Machines Learn to Plan A Critical Interrogation of Machine Vision Techniques in Architecture	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
Pipes of AI Machine Learning Assisted 3D Modeling Design	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
Automatic Generation of the Schematic Mechanical System Drawing by Generative Adversarial Network	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
An Academy of Spatial Agents Generating spatial configurations with deep reinforcement learning	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
Deep Learning surrogate models for spatial and visual connectivity	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0
3D Graph Convolutional Neural Networks in Architecture Design	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
The Emoting City Designing feeling and artificial empathy in mediated environments	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
Occupancy-informed Introducing a method or flexible behavioural mapping in architecture using machine vision	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
A Large Scale Measurement and Quantitative Analysis Method of Facade Color in the Urban Street	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

Using Deep Learning																
HierarchyNet Hierarchical CNN Based Urban Building Classification	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
Drawing Recognition Integrating Machine Learning Systems into Architectural Design Workflows	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
Automatic Recognition and Segmentation of Architectural Elements	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
An AI Lens on Historic Cairo A Deep Learning Application for Minaret Classification	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
Automatic Generation of Horizontal Building Mask Images by Using a 3D Model with Aerial Photographs for Deep Learning	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
Machine Learning for Comparative Urban Planning at Scale An Aviation Case Study	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
Anxious Landscapes Correlating the Built Environment with Mental Health through Deep Learning	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
Deep Learning Methods for Urban Analysis and Health Estimation of Obesity	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
Clustering and Morphological Analysis of Campus Context	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
Urban Detection Kit A System for Collection and Analysis of Street Level Imagery	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
Detecting Urban Issues With the Object Detection Kit	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0

A big data evaluation of urban street walkability using deep learning and environmental sensors	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
An investigation of the visual features of urban street vitality using a convolutional neural network	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
Monitoring Asphalt Pavement Aging and Damage Conditions from Low Altitude UAV Imagery Based on a CNN Approach	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
Making a New City Image	1	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0
Simulating urban land use change by integrating a convolutional neural network with vector-based cellular automata	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
City Wide Traffic Congestion Prediction Based On CNN LSTM and Transpose CNN	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
GPS Based citywide traffic congestion forecasting using CNN-RNN and C3D hybrid model	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
Suggestive Site Planning with Conditional GAN and Urban GIS Data	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
Text To Form 3D Prediction by Linguistic Descriptions	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
Towards a Distributed Robotically Assisted Construction Framework Using Reinforcement Learning	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
Machine Learning for Fabrication of Graded Knitted Membranes	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Exploration of Campus Layout Based on Generative Adversarial Network Discussing the Significance of	0	1	0	0	0	0	0	0	1	0	0	0	0	0	0	0

Small Amount Sample Learning for Architecture																
A Preliminary Study on the Formation of the General Layouts on the Northern Neighborhood Community Based on GauGAN Diversity Output Generator	0	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0
Architectural Visualisation with Conditional Generative Adversarial Networks What machines read in architectural sketches	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
Drawn Together Machine Augmented Sketching in the Design Studio	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
DeepGreen Coupling Biological and Artificial Intelligence in Urban Design	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
Machine Learning Aided 2D 3D Architectural Form Finding at High Resolution	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
BIM Hyperreality Data Synthesis Using BIM and Hyperrealistic Rendering for Deep Learning	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
A Machine Learning Method of Predicting Behavior Vitality Using Open Source Data	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
Reprogramming Urban Block by Machine Creativity How to use neural networks as generative tools to design space	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
Encoded Images Representational Protocols for Integrating cGANs in Iterative Computational Design Processes	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Data Driven Midsole Performance Oriented Midsole Design Using Computational Multi Objective Optimization	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0

On AI Adoption Issues in Architectural Design Identifying the issues based on an extensive literature review	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
Steering into the Skid Arbitrating Human and Artificial Intelligences to Augment the Design Process	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
Urban Emotion The interrogation of social media and its implications within urban context	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
Reinforcement Learning for Sequential Assembly of SL Blocks	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
Training Spaces Fostering machine sensibility for spatial assemblages through wave function collapse and reinforcement learning	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
Towards Abductive Reasoning Based Computational Design Tools Using Machine Learning as a way to explore the combined design spaces of multiple parametric models	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0
A Chained Machine Learning Approach to Motivate Retro Cladding of Residential Buildings	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
Approach to Auto Recognition of Human Trajectory in Squares using Machine Learning-Based Methods	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
A Research On Building Cluster Morphology Formation Based On Wind Environmental Performance And Deep Reinforcement Learning	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
Automatic Diminished Reality Based Virtual Demolition Method using Semantic Segmentation and Generative Adversarial Network for Landscape Assessment	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0

Mixed Reality Landscape Visualization Method with Automatic Discrimination Process for Dynamic Occlusion Handling Using Instance Segmentation	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
Comparative Evaluation of Tensor based Data Representations for Deep Learning Methods in Architecture	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
Visualizing Deep Learning Models for Urban Health Analysis	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
From Streetscape to Data Semantic segmentation for the prediction of outdoor thermal comfort	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
A deep 2d 3d Feature Level fusion for classification of UAV multispectral imagery in urban areas	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
A novel CNN LSTM based approach to predict urban expansion	1	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
Fill In The Blanks Deep Convolutional Generative Adversarial Network to Investigate the Virtual Design Space of Historical Islamic Patterns	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
Limits to Applied ML in Planning and Architecture Understanding and defining extents and capabilities	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

Załącznik 4: Lista punktów głównych wykorzystanych do treningu sieci neuronowej eksperymentu głównego oraz lista odrzuconych kandydatów.

Punkty główne ze względu na lokalizację (zbiór treningowy)				
Źródło	Strona	Punkt	Przesłanki	Lokalizacja
Kevin Lynch, The Image of the City (Lynch, 1960)	82	Kopuła Santa Maria del Fiore	Lokalizacja, skala, widoczność z wielu kątów	43.7731244688946, 11.256950187977283
Kazimierz Wejchert, Elementy Kompozycji	67	Ratusz, Sztokholm	Lokalizacja, rola znaczeniowa, widoczność, wysokość wieży	59.32727009950819, 18.056018797869122
	68	Cytadela w Irbilu, Irbil, Irak	Lokalizacja, skala	36.191210933629534, 44.00913794809795

Urbanistycznej (Wejchert, 1984)	68	Katedra Wniebowzięcia Najświętszej Maryi Panny w San Francisco	Lokalizacja, skala (na osi ul. O'Farrell St)	37.78423183174164, -122.42539032733146
	68	Zamek Królewski, Warszawa	Lokalizacja, skala, symbol	52.24774010305404, 21.014167214068163
	68	Wawel, Kraków	Lokalizacja, skala, symbol, na wzgórzu	50.05474086414226, 19.935143547684117
	68	Zamek Królewski, Chęciny	Lokalizacja, symbol, na wzgórzu	50.797412649263485, 20.460262023104846
	75	Wieża Eiffla, Paryż	Lokalizacja, skala, widoczność	48.85823672734345, 2.2945431835473373
	98	Ratusz, Sandomierz	Lokalizacja, piękno, organizacja przestrzeni	50.67934830667463, 21.74930596113819
	103	Bazylika Dominikanów, Grodzka, Kraków	Punkt charakterystyczny we wnętrzu urbanistycznym	50.05933397190924, 19.93906864905628
	103	Bazylika Franciszkanów, Grodzka, Kraków	Punkt charakterystyczny we wnętrzu urbanistycznym	50.05922783625676, 19.936508429930548
	103	Kościół św. Piotra i Pawła, Grodzka, Kraków	Punkt charakterystyczny we wnętrzu urbanistycznym	50.056938531502624, 19.939123449264546
	103	Kościół św. Andrzeja Grodzka, Kraków	Punkt charakterystyczny we wnętrzu urbanistycznym	50.05659879699593, 19.938399930555306
	103	Kościół św. Wojciecha, Grodzka Kraków	Punkt charakterystyczny we wnętrzu urbanistycznym	50.060909489198764, 19.937716083709404
	103	Zielona Brama, Długa, Gdańsk	Punkt charakterystyczny we wnętrzu urbanistycznym	54.3479487542666, 18.655738303579398
	103	Ratusz Głównego Miasta, Długa, Gdańsk	Punkt charakterystyczny we wnętrzu urbanistycznym	54.348795012676355, 18.652638529291472
	103	Złota Brama, Długa, Gdańsk	Punkt charakterystyczny we wnętrzu urbanistycznym	54.349682029318124, 18.648035798140057
	135	Gmach Admiralicji, Petersburg, Rosja	Lokalizacja, naprowadzenie przez linie naprowadzające, skala	59.937533472231046, 30.3086325034946
	135	Sobór św. Izaaka, Petersburg, Rosja	Lokalizacja, naprowadzenie przez linie naprowadzające	59.93401374108756, 30.306188910575546
	135	Pomnik Mikołaja I, Petersburg, Rosja	Lokalizacja, naprowadzenie przez linie naprowadzające	59.93209141704214, 30.30837462958869

	144	Rynek w Cieszynie, centrum zaakcentowane fontanną, Cieszyn	Lokalizacja, fontanna w centrum placu, w osi ul. Szersznika i Matejki	49.748754047391294, 18.633395404533033
	147	Muzeum Narodowe, Plac Wacława, Praga, Czechy	Punkt charakterystyczny we wnętrzu urbanistycznym	50.0788960556454, 14.430894625736745
	148	Kościół św. Kazimierza Królewicza, Rynek Nowego Miasta, Warszawa	Lokalizacja, skala, kontrast	52.253195278330104, 21.008799004338233
	153	Kolumna Vendome, Plac Vendome, Paryż	Lokalizacja, centrum placu, wysokość, naprowadzenie przez otwarcie wąskie, węzeł założenia sieciowego	48.867469943263195, 2.329442923887159
	155	Łuk Triumfalny, Plac De Gaulle'a, Paryż	Lokalizacja, centrum placu, naprowadzenie przez otwarcia kierunkowe, węzeł założenia sieciowego	48.873768393508314, 2.2950556398040525
	155	Palace Albanija, Plac Terazije, Belgrad, Serbia	Lokalizacja, otwarcie agresywne, zamknięcie wnętrza urbanistycznego	44.815087864814686, 20.460130818534253
	165	Bazylika Kolegiacka, Pułtusk	Lokalizacja, początek wnętrza podłużnego	52.70750597693507, 21.088530872765254
	165	Zamek, Pułtusk	Lokalizacja, początek wnętrza podłużnego	52.70380812487929, 21.094723871177564
	165	Ratusz, Pułtusk	Lokalizacja, centrum wnętrza podłużnego	52.705844098292765, 21.09126635843592
	168	Pałac Kultury i Nauki w Warszawie	Lokalizacja, skala, naprowadzenia, otwarcia, sprzęganie szeregu wnętrz	52.231759984653166, 21.006210337564905
Juliusz Żórawski, O Budowie Formy Architektonicznej (Żórawski, 1962)	50	Kopuła Bazyliki św. Piotra, Plac św. Piotra, Watykan	Lokalizacja w zespole formalnym Bazyliki Watykańskiej	41.902166506491284, 12.453380378958762
	50, 130	Dziedziniec Królewski, Pałac Wersalski, Wersal, Francja	Lokalizacja w zespole formalnym Pałacu Wersalskiego, węzeł założenia sieciowego	48.804324297071865, 2.1221411080918013
	50, 130	Skrzyżowanie osi, Pałac Wersalski, Wersal, Francja	Lokalizacja w zespole formalnym Pałacu	48.80796942185697, 2.1083895183337797

			Wersalskiego, węzeł założenia sieciowego	
50, 130	Centrum Parku Wersalskiego, Pałac Wersalski, Wersal, Francja	Lokalizacja w zespole formalnym Pałacu Wersalskiego, węzeł założenia sieciowego	48.810016208806275, 2.1000599070438986	
50, 130	Centrum polany w Parku Wersalskim, Pałac Wersalski, Wersal, Francja	Lokalizacja w zespole formalnym Pałacu Wersalskiego, węzeł założenia sieciowego	48.81442471663893, 2.0839451265006317	
52	Kościół Farny, Kazimierz Dolny	Lokalizacja w zespole formalnym Rynku w Kazimierzu Dolnym, Topografia	51.322711426689, 21.948628889726415	
52	Klasztor Franciszkański, Kazimierz Dolny	Lokalizacja w zespole formalnym Rynku w Kazimierzu Dolnym, Topografia	51.32087405325941, 21.945140773563352	
52	Campanila, Plac św. Marka, Wenecja	Lokalizacja w zespole formalnym Placu św. Marka	45.434025912165076, 12.339092995552168	
52	Bazylika św. Marka, Plac św. Marka, Wenecja	Lokalizacja w zespole formalnym Placu św. Marka	45.43451609824373, 12.339733475527506	
95	Sukiennice, Rynek, Kraków	Lokalizacja, centrum, naprowadzenia	50.061647834367996, 19.937315107395793	
95	Bazylika Mariacka, Rynek, Kraków	Lokalizacja	50.061675708795484, 19.938982492824767	
95	Wieża Ratuszowa, Rynek, Kraków	Lokalizacja	50.06147146955907, 19.936420718989517	
118	Rondo Pól Elizejskich, Paryż	Lokalizacja w zespole formalnym Luwru	48.86898261130083, 2.3101378906524426	
118	Front Kościoła de la Madeleine, Paryż	Lokalizacja w zespole formalnym Luwru	48.8693998261807, 2.3240308641214726	
118	Luwr, Paryż	Lokalizacja w zespole formalnym Luwru	48.86058053577298, 2.3376416605527313	
118	Łuk Carrousel, Plac Carrousel, Luwr, Paryż	Lokalizacja w zespole formalnym Luwru	48.861716817964265, 2.332933597576387	

	VI	Centrum placu, Palmanova, Włochy	Dominanta ze względu na umiejscowienie, punkt centralny założenia radialnego	45.90537119979507, 13.309922227650372
Francis D. K. Ching, Architecture Form, Space, and Order (Ching, 2007)	5	Wieża klasztorna, Mont-Saint-Michel, Normandia, Francja	Lokalizacja, widoczność, topografia, wysokość, strzelistość	48.63604104667767, -1.5114044222987941
	7	Lincoln Memorial, The Mall, Waszyngton, USA	Lokalizacja, widoczność	38.88930740144755, -77.05015872612371
	7	Washington Monument, The Mall, Waszyngton, USA	Lokalizacja, widoczność	38.88946469983048, -77.03525028423452
	7	Kapitol, The Mall, Waszyngton, USA	Lokalizacja, widoczność	38.889809176942585, -77.00912382948968
	128	Wieża ratusza, Piazza del Campo, Siena, Włochy	Lokalizacja, naprowadzenia ulic, strzelistość	43.31831653018856, 11.332146530598532
	129	Taj Mahal, Agra, Indie	Lokalizacja, skala, punkt centralny	27.17497404617232, 78.04201862461521
	270	Pałac Karlsruhe, Karlsruhe, Niemcy	Lokalizacja, centrum założenia radialnego	49.01396719257856, 8.40441705573585
	276	Plac przed Operą Garnier, Paryż, Francja	Lokalizacja, punkt formalnie podkreślony	48.87068797788575, 2.3323230481763524
	276	Plac Trocadero, Paryż, Francja	Lokalizacja, punkt formalnie podkreślony	48.86287321044156, 2.2873146954558727
	276	Bassins du Chaps de Mars, Pola Marsowe, Paryż, Francja	Lokalizacja, punkt formalnie podkreślony	48.856102414304054, 2.29790349028674
	276	Esplanade Jacques Chaban-Delmas, Paryż, Francja	Lokalizacja, punkt formalnie podkreślony	48.84728273681396, 2.31163527992454
	276	Fontanna w Ogrodzie Luksemburskim, Paryż, Francja	Lokalizacja, punkt formalnie podkreślony	48.84692429920298, 2.3372009493797767
	277	Biały Dom, Waszyngton, USA	Lokalizacja, punkt formalnie podkreślony	38.89753731966178, -77.03651592253495
	277	Lincoln Park, Waszyngton, USA	Lokalizacja, punkt formalnie podkreślony	38.88978118235831, -76.98984532402181
	277	Więzienie DC, Waszyngton, USA	Lokalizacja, punkt formalnie podkreślony	38.883949370306745, -76.97670938952504

	Zał. Nr 5, Krajobraz	Pomnik gen. Tadeusza Kościuszki, Plac Wolności	Obiekt o znaczeniu kompozycyjnym, wyróżniający się ze względu na kubaturę (formę), wysokość lub walory architektoniczne widoczny w panoramach z wielu punktów widokowych; Skrzyżowanie osi widokowych	51.776789126483266, 19.454710476677143
Studium Uwarunkowań i Kierunków Zagospodarowania Przestrzennego Miasta Łodzi, 2018 (Prezydent miasta Łodzi, 2018)	Zał. Nr 5, Krajobraz	Kościół Wniebowzięcia Najświętszej Maryi Panny. Plac Kościelny	Obiekt o znaczeniu kompozycyjnym, wyróżniający się ze względu na kubaturę (formę), wysokość lub walory architektoniczne widoczny w panoramach z wielu punktów widokowych; Skrzyżowanie osi widokowych	51.78317720219291, 19.45396901345193
	Zał. Nr 5, Krajobraz	Pomnik Pamięci Ofiar Pomordowanych w Więzieniu na Radogoszczu, Plac Pamięci Narodowej	Miejsce o widoku harmonijnym i pozytywnym odbiorze, obiekt o znaczeniu kompozycyjnym, wyróżniający się ze względu na kubaturę (formę), wysokość lub walory architektoniczne widoczny w panoramach z wielu punktów widokowych; Skrzyżowanie osi widokowych	51.80892197389786, 19.438688189912302
	Zał. Nr 5, Krajobraz	Wieża kościoła karmelitów bosych, Czereśniowa	Obiekt o znaczeniu kompozycyjnym, wyróżniający się ze względu na kubaturę (formę), wysokość lub walory architektoniczne widoczny z osi widokowej	51.8081468942271, 19.435090138135216
	Zał. Nr 5, Krajobraz	Skrzyżowanie Strykowskiej, Wojska Polskiego i Palki	Miejsce o widoku harmonijnym z pojedynczymi elementami zaburzającymi przestrzeń, węzeł komunikacyjny, skrzyżowanie osi, otwarcie widokowe	51.79236495888682, 19.4886882414949

	Załącznik Nr 5, Krajobraz	Skrzyżowanie w Nowosolnej	Skrzyżowanie osi na ciągu widokowym, węzeł założenia radialnego	51.793859651737456, 19.589673898312135
	Załącznik Nr 5, Krajobraz	Teatr Wielki, Plac Dąbrowskiego	Obiekt o znaczeniu kompozycyjnym, wyróżniający się ze względu na kubaturę (formę), wysokość lub walory architektoniczne widoczny w panoramach z wielu punktów widokowych; Obiekt tworzący główny motyw sylwety miasta w panoramach	51.77345590237322, 19.469707668705905
	Załącznik Nr 5, Krajobraz	Kościół pw. św. Teresy i św. Jana Bosko, Rondo Solidarności	Obiekt o znaczeniu kompozycyjnym, wyróżniający się ze względu na kubaturę (formę), wysokość lub walory architektoniczne widoczny w panoramach z wielu punktów widokowych; Obiekt tworzący główny motyw sylwety miasta w panoramach. Węzeł założenia radialnego, otwarcie widokowe	51.778226642584436, 19.48090472253959
	Załącznik Nr 5, Krajobraz	Brama Miasta, Nowe Centrum Łodzi	Obiekt o znaczeniu kompozycyjnym, wyróżniający się ze względu na kubaturę (formę), wysokość lub walory architektoniczne widoczny w panoramach z wielu punktów widokowych; Obiekt tworzący główny motyw sylwety miasta w panoramach	51.769007953435796, 19.46544822746343
	Załącznik Nr 5, Krajobraz	Kościół Ewangelicki, róg Piotrkowskiej i Czerwonej	Obiekt o znaczeniu kompozycyjnym, wyróżniający się ze względu na kubaturę (formę), wysokość lub walory architektoniczne widoczny z osi widokowej	51.748947985165664, 19.460288994047186

	Zał. Nr 5, Krajobraz	Archikatedra Łódzka, Plac Katedralny	Obiekt o znaczeniu kompozycyjnym, wyróżniający się ze względu na kubaturę (formę), wysokość lub walory architektoniczne widoczny z osi widokowej	51.7465481573585, 19.460811155352847
	Zał. Nr 5, Krajobraz	Wieża Kościoła Św. Anny	Obiekt o znaczeniu kompozycyjnym, wyróżniający się ze względu na kubaturę (formę), wysokość lub walory architektoniczne widoczny z osi widokowej	51.748760787608354, 19.48999854320184
	Zał. Nr 5, Krajobraz	Kościół św. Wojciecha, Stawy Jana	Obiekt o znaczeniu kompozycyjnym, wyróżniający się ze względu na kubaturę (formę), wysokość lub walory architektoniczne widoczny z osi widokowej	51.71313383560449, 19.488822820042742
	Zał. Nr 5, Krajobraz	Kościół św. Faustyny Kowalskiej	Obiekt o znaczeniu kompozycyjnym, wyróżniający się ze względu na kubaturę (formę), wysokość lub walory architektoniczne widoczny z osi widokowej	51.74092418911683, 19.463921520268148
	Zał. Nr 5, Krajobraz	Blok mieszkalny, Łódzki Manhattan	Obiekt o znaczeniu kompozycyjnym, wyróżniający się ze względu na wysokość; szczyt sylwetki Łódzkiego Manhattanu	51.75841861236623, 19.460056342266615
	Zał. Nr 13, Kryst. Układu Przestrz.	Atlas Arena	Dominanta w zamknięciu kompozycyjnym wewnętrznej osi współczesnej wschód- zachód	51.75720301768292, 19.42501006859015
Studium Uwarunkowań i Kierunków Zagospodarowania Przestrzennego Miasta Łodzi, 2010	Zał. Nr 13, Krystal. Układu Przestrz.	Obecna „Stajnia Jednorożców”, Łódzki Manhattan	Węzeł komunikacyjny na przecięciu głównych osi krystalizujących plan Łodzi: osi historycznej północ- południe i osi współczesnej wschód-zachód	51.7592302084747, 19.457067986333477

(Prezydent miasta Łodzi, 2010)	n/d	Skrzyżowanie Alei Unii Lubelskiej i Konstantynowskiej	Węzeł założenia radialnego, otwarcie widokowe	51.767347105376324, 19.421319217825985
Opracowanie własne	n/d	Kościół św. Katarzyny, Zgierz	Dominanta, zamknięcie osi widokowych, topografia terenu	51.85606809474244, 19.403719915059764
	n/d	Skwer Orzeszkowej	Węzeł założenia radialnego, w osi Parku Adama Mickiewicza i Liściastej, styk parku z zabudową jednorodzinną	51.80364490021924, 19.448517570556696
	n/d	Most Arturówek	Oś stawów na Bzurze, węzeł widokowy i komunikacyjny w kompleksie Arturówka	51.822739317054115, 19.474116941045487
	n/d	Plac Słoneczny, Radogoszcz	Punkt centralny krystalizujący zabudowę osiedla na Radogoszczu	51.82057670889436, 19.443397750387696
	n/d	Kościół Najświętszego Sakramentu	Dominanta w osi Balladyny	51.81902796232642, 19.440944165924964
	n/d	Kopiec w Parku Ocalałych	Dominanta wysokościowa, węzeł założenia parkowego	51.785467417188364, 19.474582897679383
	n/d	Pomnik Sprawiedliwych Wśród Narodów Świata, Park Ocalałych	Węzeł założenia parkowego	51.78534687726929, 19.470659781812284

Odrzuceni kandydaci do zbioru treningowego. Przykłady nieaktualne, nieistniejące, dominanty i punkty formalnie podkreślone jednak niebędące punktami głównymi oraz punkty główne w nieodpowiedniej skali kompozycji.

Źródło	Strona	Punkt	Przesłanki	Lokalizacja
Kevin Lynch, The Image of the City (Lynch, 1960)	79	Wiatrowskaz z konikiem polnym, Faneuil Hall, Boston	Symbol	42.36004681293719, -71.05608434517856
	79	Złota kopuła State House, Boston	Lokalizacja na wzgórzu	42.358174439996965, -71.06365005993443
	79	Los Angeles City hall, LA	Wysokość, forma	34.05373945159497, -118.24270006057436
	79	Christian Science Building, Boston	Kontrast z otoczeniem	42.344905314532014, -71.0844238818036

	79	Jersey City Medical Center, przez skalę, Boston	Skala, obecnie nieaktualne	40.71631288597379, -74.05050889797708
	79	Old Hall of Records, Los Angeles Civic Center, LA	Skala okien, orientacja, detal, wyburzony	34.05415760852688, -118.24420217242267
	80	Old Hancock Building, Boston	Widoczność z wielu kątów, wysokość, obecnie nieaktualne ze względu na budowę Hancock Tower w latach 70.	42.34980365499649, -71.07325103052949
			Następca Old Hancock, też widoczność, warto dodać, wybudowany w latach 70. po wydaniu Image Of The City	42.34922887724831, -71.07507774050572
	80	Richfield Oil Building, Los Angeles	Widoczność z wielu kątów, wysokość, wyburzony	34.05109536320206, -118.25719648167308
	80	Little Gray Lady, 7th/Flower st. Corner, Los Angeles	Mała skala, cofnięcie od ulicy, wiek, materiał (drewno), wyburzony	34.0490326485702, -118.25878798165265
	81	Telephone Building, Bowdoin Square, Cambridge Street, Boston	Lokalizacja, pomaga podjąć decyzję o kontynuowaniu podróży wzdłuż ulicy	42.36139626824704, -71.06158677052751
	81	Baptist Temple Symphony Hall, Los Angeles	Funkcja, negatywny kontrast między wzniosłą funkcją a słabą formą, wyburzone	34.04943500856028, -118.25234030903788
Kazimierz Wejchert, Elementy Kompozycji Urbanistycznej (Wejchert, 1984)	81	Custom House Tower, Boston	Wysokość, widoczność z wielu kątów, nieaktualne, obrosnięte wyższymi budynkami	42.359058602722364, -71.05356664523302
	60	Wieża WTC, Nowy York	Skala, widoczność z wielu kątów, zniszczone	40.7115161000248, -74.01318311777727
	66	Gmach ONZ, Nowy York	Lokalizacja i rola znaczeniowa	40.750107455187965, -73.96772474728935
	68	Alcazar de Toledo, Toledo, Hiszpania	Skala, wyróżniająca się forma	39.85791251490771, -4.0205081766324655
	68	Katedra NSMP, Toledo, Hiszpania	Skala, wyróżniająca się forma	39.85709799709746, -4.023574920356383
	68	Seminario Conciliar de San Ildefonso de Toledo	Skala, wyróżniająca się forma	39.85390849764476, -4.023989478439541

	68	Skarpa staromiejska, Lublin	Lokalizacja, skala, symbol, na wzgórzu	51.247891054987896, 22.567945795202885
	68	Hradczany, Praga, Czechy	Lokalizacja, skala, symbol, na wzgórzu	50.09060879569342, 14.400508825432444
	73	Rondo gen. J. Ziętki, Katowice	Lokalizacja, sąsiedztwo ważnych obiektów	50.2643570826509, 19.023559714586202
	75	Centrum Pompidou, Paryż	Lokalizacja, skala, kontrast	48.86047151124023, 2.352460724671782
	79	Pomnik Deportowanych, Paryż	Układ przestrzenny, punkt podkreślony	48.85163525581611, 2.3526449696610374
	97	City Hall, Toronto, Kanada	Lokalizacja, forma, piękno	43.65349088264935, -79.38397248015865
	103	Posąg Neptuna, Długa, Gdańsk	Punkt charakterystyczny we wnętrzu urbanistycznym	54.348545404820186, 18.653241216625045
	103	Dwór Artusa, Długa, Gdańsk	Punkt charakterystyczny we wnętrzu urbanistycznym	54.3486921629679, 18.65334985881403
	136	Teatr Wybrzeże, Targ Węglowy, Gdańsk	Lokalizacja, płaszczyzna zamykająca, naprowadzenie przez linie naprowadzające	54.35118601294387, 18.64862343364292
	137	Wieżyczka Kościoła Paulinów, Stare Miasto, Warszawa	Lokalizacja, zamknięcie zaułka, linie naprowadzające	52.24828499096968, 21.01269745708259
	138	Małostrażńska Wieża Mostowa, Praga, Czechy	Lokalizacja, linie naprowadzające tworzą cokół dla dominanty	50.08731418478959, 14.406866444643768
	149	Kościół Mariacki, Stare Miasto, Gdańsk	Skala	54.34980556485089, 18.653253868933913
	149	Biała Wieża, Rynek, Hradec Kralove, Czechy	Wysokość, strzelistość	50.20900783271558, 15.83081466108146
	149	Katedra Świętego Ducha, Rynek, Hradec Kralove, Czechy	Wysokość, strzelistość	50.20876401060737, 15.831181091543646
	149	Ratusz, Rynek, Hradec Kralowe, Czechy	Wysokość, strzelistość	50.2092911301241, 15.831425996987806
	155	Chilehaus, Hamburg, Niemcy (+liczne otwarcia	Lokalizacja, otwarcia agresywne, zamknięcie wewnątrz urbanistycznych	53.54830776019474, 10.003000335414914

		agresywne tego typu w całym Hamburgu)		
	165	Katedra w Sienie, Siena, Włochy	Skala, wyróżniająca się forma, sprzęganie szeregu wnętrz urbanistycznych	43.31780149601075, 11.329034108204878
Juliusz Żórawski, O Budowie Formy Architektonicznej (Żórawski, 1962)	50	Obelisk w centrum, Plac św. Piotra, Watykan	Lokalizacja w zespole formalnym Bazyliki Watykańskiej	41.90223811543213, 12.457262932471398
	50	Fontanna, Ognisko północne elipsy, Plac św. Piotra, Watykan	Lokalizacja w zespole formalnym Bazyliki Watykańskiej	41.90277489658461, 12.45723569424325
	50	Fontanna, Ognisko południowe elipsy, Plac św. Piotra, Watykan	Lokalizacja w zespole formalnym Bazyliki Watykańskiej	41.90170130750023, 12.457278008723337
	50	Posąg Marka Aureliusza, Plac Kapitoński, Rzym	Lokalizacja w zespole formalnym Placu Kapitońskiego	41.893323183044274, 12.482934735894471
	50	Wieża zegarowa Pałacu Senatorów, Plac Kapitoński, Rzym	Lokalizacja w zespole formalnym Placu Kapitońskiego	41.89301876417793, 12.483561305130888
	52	Pałac Dożów, Plac św. Marka, Wenecja	Lokalizacja w zespole formalnym Placu św. Marka	45.43384799274172, 12.339997088644589
	52	Kolumna św. Marka, Plac św. Marka, Wenecja	Lokalizacja w zespole formalnym Placu św. Marka	45.4333540584212, 12.33991578944312
	52	Kolumna San Todaro, Plac św. Marka, Wenecja	Lokalizacja w zespole formalnym Placu św. Marka	45.433285220072136, 12.33965995328723
	133	Pomnik Gattamelaty Donatella, Piazza del Santo, Padwa, Włochy	Lokalizacja na Piazza del Santo	45.40157301647186, 11.87998528794386
	133	Fasada Bazyliki św. Antoniego, Piazza del Santo, Padwa, Włochy	Lokalizacja na Piazza del Santo	45.40133877761612, 11.88033432873618
Francis D. K. Ching, Architecture Form, Space, and Order (Ching, 2007)	5	Posąg Marka Aureliusza, Plac Kapitoński, Rzym	Lokalizacja, rysunek posadzki	41.893323183044274, 12.482934735894471
	20	Obelisk, Piazza della Trinità dei Monti, Rzym, Włochy	Lokalizacja, widoczność, topografia, naprowadzenie przez schody hiszpańskie	41.90616294978277, 12.483210466966078

	20	Kościół Trinita dei Monti, Rzym, Włochy	Lokalizacja, widoczność, topografia, naprowadzenie przez schody hiszpańskie	41.90623092251156, 12.48341910603524
	31	Obelisk, Piazza Ducale, Sabbioneta, Włochy	Lokalizacja	44.99931830761596, 10.489976172243857
	31	Wieża ratusza, Piazza Ducale, Sabbioneta, Włochy	Lokalizacja, wysokość, strzelistość	44.999354711748424, 10.490290654131071
	128	Punkt wystąpienia, Piazza del Campo, Siena, Włochy	Naprowadzenia rysunku na posadze	43.318267770612586, 11.331817478341481
	128	Obelisk w centrum, Plac św. Piotra, Watykan	Lokalizacja w zespole formalnym bazyliki Watykańskiej	41.90223811543213, 12.457262932471398
	128	Fontanna, ognisko północne elipsy, Plac św. Piotra, Watykan	Lokalizacja w zespole formalnym bazyliki Watykańskiej	41.90277489658461, 12.45723569424325
	128	Fontanna, ognisko południowe elipsy, Plac św. Piotra, Watykan	Lokalizacja w zespole formalnym bazyliki Watykańskiej	41.90170130750023, 12.457278008723337
	276	Obserwatorium Paryskie, Paryż, Francja	Lokalizacja, punkt formalnie podkreślony	48.83641607356397, 2.3365973292578155

Załącznik 5. Tabela ewaluacji wytrenowanego algorytmu na zbiorze testowym

Kodowana techniką multi-hot:

Wartości procentowe:			74,4%	50,0%	13,8%	10,6%	3,8%	75,6%	56,9%	2,5%	Skuteczność całkowita: 59,4%
Nazwy robocze	Koord.	L.p.	Czy rozpoznany jakiegokolwiek punkt główny?	Czy dominanta lub punkt orientacyjny o znaczącej skali o lokalizacji podkreślonej przez kompozycję miejską i/lub topografię terenu?	Czy zakończ. założenia osiowego?	Czy główny punkt centralny w kompozycji radialnej?	Czy jeden z głównych węzłów założenia sieciowego?	Czy linie naprowadz. są zaznaczone sensownie?	Czy przynajmniej jedno z kryteriów dot. punktów głównych spełnione?	Czy pominięcie punktu miało sens?	Oznaczone punkty główne
Plac Unii Lubelskiej	52.21357, 21.02163	1	1	1		1	1	1	1		Plac Unii, Biurowiec Plac Unii, Riviera
		2	1	1					1		Biurowiec Plac Unii
		3	1	1		1	1	1	1		Plac Unii, Biurowiec Plac Unii, Riviera
		4	1			1	1	1	1		Plac Unii, Estakada przy Placu Na Rozdrożu (pominięcie)

											samego Placu na Rozdrożu)
		5	1	1		1	1	1	1		Plac Unii, Biurowiec Plac Unii
Muzeum Fryderyka Chopina	52.23659, 21.02291	6	1	1				1	1		Prudential
		7									
		8	1	1	1				1		Wieżowiec domykający oś ul. Kopernika
		9									
		10									
Plac Zbawiciela (+ place okoliczne)	52.21985, 21.01800	11	1				1	1	1		Estakada przy Placu Na Rozdrożu (pominięcie samego Placu na Rozdrożu)
		12	1	1	1			1	1		Otwarcie na Plac Konstytucji
		13	1	1	1			1	1		Dwa główne otwarcia na Plac Konstytucji
		14	1	1	1			1	1		Dwa główne otwarcia na Plac Konstytucji
		15	1				1	1	1		Estakada przy Placu Na Rozdrożu (pominięcie samego Placu na Rozdrożu)
Osiedle Zacisze	52.28421, 21.06992	16	1	1				1	1		Placyk przed Domem Kultury Zacisze
		17	1	1				1	1		Placyk przed Domem Kultury Zacisze
		18	1	1				1	1		Placyk przed Domem Kultury Zacisze, Skwer z placem zabaw przy mostku pieszym nad Kanałem Bródnowskim
		19	1			1		1	1		Duże skrzyżowanie Młodzieżowej z Radzywińskiej
		20	1	1				1	1		Placyk przed Domem Kultury Zacisze
Ursynów Północny	52.15936, 21.02794	21	1	1	1			1	1		Wyjście z ciągu pieszego ul. Dzwonniczej na Aleję Komisji Edukacji Narodowej, duże Skrzyżowanie Romera z Puławską
		22	1	1	1			1	1		Duże Skrzyżowanie Romera z Puławską
		23	1	1	1			1	1		Wieżowiec mieszkalny w osi Parku Kozłowskiego
		24	1	1	1			1	1		Wyjście z ciągu pieszego ul. Dzwonniczej na Aleję Komisji Edukacji Narodowej
		25						1			
		26	1	1		1		1	1		Duże skrzyżowanie Płaskowickiej z

Ursynów Płd., Natolin	52.14015, 21.05629									Aleja Komisji Edukacji Narodowej
		27	1	1		1		1	1	Duże skrzyżowanie Płaskowickiej z Dereniową/Stryeń skich
		28	1	1				1	1	Wieżowiec mieszkalno- usługowy Galeria Ursynów
		29	1	1				1	1	Wieżowiec mieszkalno- usługowy Galeria Ursynów
		30	1	1				1	1	Wieżowiec mieszkalno- usługowy Galeria Ursynów
Saska Kępa, ul. Francuska	52.23326, 21.05492	31	1	1		1		1	1	Dwa wieżowce mieszkalne w osiach ulic radialnych na Saskiej Kępie
		32	1	1				1	1	Wejście do Stadionu Narodowego od strony Ronda Waszyngtona
		33	1	1		1		1	1	Rondo Waszyngtona
		34								
		35	1	1					1	Rondo Waszyngtona
Cerkiew Praska	52.25481, 21.03322	36	1	1				1	1	Skrzyżowanie Alei Solidarności z Targową przy Dworcu Wileńskim
		37	1	1				1	1	Skrzyżowanie Alei Solidarności z Targową przy Dworcu Wileńskim
		38	1	1				1	1	Skrzyżowanie Alei Solidarności z Targową przy Dworcu Wileńskim
		39	1	1	1			1	1	Skrzyżowanie Alei Solidarności z Targową przy Dworcu Wileńskim, Przebieg piesze między budynkami w osi długiego budynku Jagiellońska 44
		40	1	1				1	1	Skrzyżowanie Alei Solidarności z Targową przy Dworcu Wileńskim
Stara Praga	52.25214, 21.03764	41	1	1				1	1	Dworzec Wileński
		42	1							Skrzyżowanie Zamoyskiego z Sokoła
		43	1	1		1		1	1	Centrum Placu Weteranów 1863 roku w osi Katedry św. Michała
		44	1	1				1	1	Skrzyżowanie Alei Solidarności z Targową przy Dworcu Wileńskim
		45	1	1				1	1	Skrzyżowanie Alei Solidarności z

										Targową przy Dworcu Wileńskim
Zakłady FSO	52.28464, 21.01087	46						1		
		47					1			
		48						1		
		49	1				1			Punkt na Jagiellońskiej naprzeciwko przychodni Luxmed
		50					1			
Focus	52.21757, 21.00811	51	1	1			1	1		Wejście do Focusa
		52					1			
		53	1	1			1	1		Wejście do Focusa
		54	1	1			1	1		Riviera
		55	1	1			1	1		Riviera, Zebra Tower
Nowodwory	52.33233, 20.94701	56	1	1	1		1	1		Blok mieszkalny Premium Point zamykający oś ul. Ćwiklińskiej
		57	1				1			Stadion Białoleckiego Ośrodka Sportu
		58	1		1		1	1		Narożny blok mieszkalny przy Książkowej 9F w osi wewnętrznej drogi osiedlowej
		59	1		1		1	1		Narożny blok mieszkalny przy Książkowej 9F w osi wewnętrznej drogi osiedlowej
		60								
Domaniewska	52.18348, 21.00218	61	1	1			1	1		Skrzyżowanie Domaniewskiej z Wołoską: węzeł komunikacyjny „Mordor”
		62	1	1			1	1		Skrzyżowanie Wołoskiej z Marynarską/Wilanowską
		63	1	1			1	1		Skrzyżowanie Domaniewskiej z Wołoską: węzeł komunikacyjny „Mordor”
		64	1	1			1	1		Skrzyżowanie Domaniewskiej z Wołoską: węzeł komunikacyjny „Mordor”, skrzyżowanie Woronicza z Wołoską
		65	1	1			1	1		Skrzyżowanie Domaniewskiej z Wołoską: węzeł komunikacyjny „Mordor”
Sady Żoliborskie	52.26723, 20.97172	66	1	1			1	1		Skrzyżowanie Krasińskiego z Popiełuszki
		67	1	1		1	1	1		Skrzyżowanie Słowackiego z Popiełuszki
		68	1				1			Kładka pieszo-rowerowa nad aleją Armii Krajowej

		69	1	1				1	1		Skrzyżowanie Krasieńskiego z Popieluski
		70	1	1		1		1	1		Skrzyżowanie Słowackiego z Popieluski
Plac Narutowicza	52.21910, 20.98368	71	1	1	1			1	1		Wieżowiec Ochota Residence
		72	1	1				1	1		Wieżowiec Atlas Tower
		73	1	1				1	1		Skrzyżowanie Grójeckiej z Wawelską/Kopińską
		74	1	1				1	1		Wieżowiec Ochota Residence
		75	1	1				1	1		Skrzyżowanie Grójeckiej z Wawelską/Kopińską
Las Bielański	52.29324, 20.95605	76						1			
		77						1		1	
		78						1			
		79						1			
		80	1	1				1	1		Skrzyżowanie Marymonckiej/De wajtis i Marymonckiej: wjazd do Lasu Bielańskiego
Pałac w Wilanowie	52.16509, 21.08919	81	1								Pawilon Potockiego 20/22
		82	1					1			Skrzyżowanie Alei Wilanowskiej z Przyczółkową
		83	1					1			Skrzyżowanie Alei Wilanowskiej z Przyczółkową
		84	1								Skrzyżowanie Alei Wilanowskiej z Przyczółkową, pętla autobusowa Wilanów
		85									
Centrum Nauki Kopernik, Bulwary Wiślane	52.24220, 21.02883	86						1			
		87	1	1					1		Motel One Warszawa Chopin
		88	1	1				1	1		Elektrownia Powiśle, blok mieszkalny przy Leszczyńskiej
		89									
		90									
Muzeum Powstania Warsz.	52.23224, 20.98138	91	1	1					1		Warsaw Spire, Warsaw Trade Tower
		92	1	1				1	1		Warsaw Spire, Warsaw Trade Tower
		93									
		94									
		95	1	1					1		Warsaw Spire
Stary Mokotów	52.20354, 21.01006	96	1	1				1	1		Skrzyżowanie Małalińskiego z Aleją Niepodległości
		97	1	1				1	1		Skrzyżowanie Małalińskiego z

										Aleja Niepodległości
		98	1	1				1	1	Skrzyżowanie Madalińskiego z Aleją Niepodległości
		99	1	1	1			1	1	Wieżowiec mieszkalny Giordana Bruna 34
		100								
Pałac na Wyspie (Łazienki)	52.21476, 21.03587	101	1					1		Centrum Stawu Łazienkowskiego w osi Podchorążówki (przed Pałacem na Wyspie)
		102	1					1		Centrum Stawu Łazienkowskiego w osi Podchorążówki (przed Pałacem na Wyspie)
		103	1							Blok mieszkalny Podchorążych 39a
		104						1		
		105						1		Centrum Stawu Łazienkowskiego w osi Podchorążówki (przed Pałacem na Wyspie)
Białoleka	52.30455, 20.97790	106	1		1			1	1	Blok mieszkalny na wzniesieniu zamykający placik osiedlowy w osi ul. Leszczykowej
		107	1		1			1	1	Blok mieszkalny na wzniesieniu zamykający placik osiedlowy na w osi ul. Leszczykowej
		108								
		109	1		1			1	1	Blok mieszkalny na wzniesieniu zamykający placik osiedlowy w osi ul. Leszczykowej, blok mieszkalny naprzeciwko Szkoły Podstawowej nr 118
		110	1		1			1	1	Blok mieszkalny na wzniesieniu zamykający placik osiedlowy w osi ul. Leszczykowej, blok mieszkalny naprzeciwko Szkoły Podstawowej nr 118
Stadion Narodowy	52.23942, 21.04580	111								
		112	1	1		1		1	1	Centrum Stadionu Narodowego
		113								
		114	1	1		1			1	Centrum Stadionu Narodowego
		115	1	1					1	Wejście do Stadionu Narodowego od strony Ronda Waszyngtona
Boernerowo	52.26295, 20.90216	116						1		
		117								
		118	1							Mały parking samochodów

										osobowych między Boernerowem a Lotniskiem Bemowo
		119						1		
		120						1		
Bielany w okolicy stacji metra Młociny	52.29032, 20.93509	121	1	1				1	1	Skrzyżowanie Kasprowicz z Sokratesa/Przy Agorze
		122	1	1					1	Skrzyżowanie Kasprowicz z Al. gen. Marii Witek: Węzeł Młociny
		123	1					1		Blok mieszkalny Wrzeciono 17
		124	1	1				1	1	Skrzyżowanie Kasprowicz z Sokratesa/Przy Agorze, blok mieszkalny Wrzeciono 42
		125	1	1				1	1	Skrzyżowanie Kasprowicz z Sokratesa/Przy Agorze
Fort VIIA Służewiec	52.17242, 21.01565	126	1			1		1	1	Skrzyżowanie Al. Lotników z Modzelewskiego, drobne założenie radialne Niegocińska-Al. Lotników
		127	1	1				1	1	Centrum stawu na Potoku Służewieckim
		128						1		
		129	1	1				1	1	Skrzyżowanie Al. Lotników z Modzelewskiego, wieżowiec mieszkalny Irysowa 29 w osi ul. Niedźwiedziej
		130	1					1		Skrzyżowanie Al. Lotników z Modzelewskiego, Gmach Wydziału Zarządzania UW
Fort Szczęśliwice	52.20671, 20.94989	131	1					1		Skrzyżowanie Al. Jerozolimskich z ul. Śmigłowca
		132						1		
		133						1		
		134						1		Narożny blok mieszkalny Włodarzewska 83
		135	1					1		
Fort Chrzanów	52.22021, 20.89245	136	1	1	1	1		1	1	Skrzyżowanie Lazurowej, Sterniczej, Kopalnianej i Steligowskiej w osi Fortu Chrzanów, węzeł komunikacyjny
		137	1					1		Dojście do małego stawu na parkingu za halą handlową Selgros
		138	1					1		
		139						1	1	

		140	1	1	1	1		1	1		Skrzyżowanie Lazurowej, Sterniczej, Kopalnianej i Steligowskiej w osi Fortu Chrzanów, węzeł komunikacyjny
Fort Blizne	52.24688, 20.89487	141	1					1			Estakada lazurowej nad Aleją Obronców Grodna
		142	1					1			Estakada lazurowej nad Aleją Obronców Grodna
		143	1								Estakada lazurowej nad Aleją Obronców Grodna
		144	1								Estakada lazurowej nad Aleją Obronców Grodna
		145	1					1			Akademik Wojskowy nr 3
Fort Wawrzyszew	52.27749, 20.90736	146									
		147	1					1			Punkt na fosie Fortu Wawrzyszew w osi fortu
		148									
		149									
		150	1	1	1				1		Wieżowiec biurowy Wólczyńska 133 domykający oś Wólczyńskiej
Fort Bema	52.25661, 20.93882	151	1					1			Blok mieszkalny Obronców Tobruku 23
		152	1					1			Narożnik Cmentarza Wojskowego
		153	1					1			Blok mieszkalny Obronców Tobruku 23
		154						1			
		155						1			
Fort Służew	52.16655, 21.04087	156	1	1					1		Dom studencki IKAR SGGW
		157	1	1				1	1		Dom studencki IKAR SGGW
		158	1								Dom studencki Feniks SGGW
		159	1	1	1			1	1		Blok mieszkalny Noskowskiego 8 domykający oś Noskowskiego, widoczny z otwarcia widokowego wzdłuż Nowoursynowskiej
		160	1	1				1	1		Dom studencki IKAR SGGW

Załącznik 6: Lista punktów głównych wykrytych przez sieć neuronową we fragmencie Warszawy

W celu demonstracji możliwości seryjnego przetwarzania większych obszarów sieć neuronowa wytrenowana w eksperymencie głównym została aplikowana do fragmentu obszaru zabudowy śródmiejskiej Warszawy o wymiarach 5x5km. Obszar został przetworzony przez sieć w 100 zachodzących na siebie na siebie kafelkach (każdy pokrywający około 1km kwadratowy). Na potrzeby wizualizacji nałożono także 7 kafelków ze zbioru testowego (Rys. 72).

Na badanym obszarze sieć rozpoznała 64 punkty główne. *Kursywą* oznaczono 2 przykłady znajdujące się także w zbiorze treningowym. **Pogrubieniem** oznaczono 43 przykłady, w których decyzja sieci neuronowej wydaje się trafna. Zwykłym tekstem oznaczono 19 przykładów niejednoznacznych. Od północy w kierunku południowym:

- Skrzyżowanie ul. Tatarskiej z Powązkowską. Zamknięcie osi ul. Tatarskiej, zakład kamieniarski bez dominanty przestrzennej.
- **Babka Tower przy rondzie Radosława.** Dominanta wysokościowa przy podkreślonym kompozycyjnie przecięciu osi komunikacyjnych. Brama wyznaczona wraz z wieżą kompleksu Forest (symetria środkowa względem ronda).
- **Wieża kompleksu biurowego Forest przy Rondzie Radosława.** Dominanta wysokościowa przy podkreślonym kompozycyjnie przecięciu osi komunikacyjnych. Brama wyznaczona wraz z Babka Tower (symetria środkowa względem ronda).
- **Wieżowiec Intraco.** Dominanta wysokościowa na zamknięciu kilku osi widokowych. Brama Północna wyznaczona wraz z North Gate.
- Zamknięcie ul. Koźlej przy ul. Franciszkańskiej. Naprzeciwko Koźlej 5. Otwarcie widokowe na zieloną ul. Fondamińskiego i tył Kościoła św. Franciszka Serafickiego.
- **Kościół Franciszka Serafickiego.** Podkreślona kompozycyjnie elewacja przy skośnie wycofanej pierzei. Zaakcentowany narożnik.
- **Ostroga regulacyjna na przedłużeniu ul. Ratuszowej.** Zamknięcie osi nad Wisłą bez dominanty przestrzennej. Ścieżka pieszka. Ekspozycja na tle Wisły i naprowadzenie przez oś wyspy piaskowej.
- *Zamek Królewski.* Naprowadzenia widokowe, podkreślona elewacja, lokalna dominanta wysokościowa wieży, lokalizacja na wzgórzu.
- **Skrzyżowanie Al. Jana Pawła i ul. Anielewicza.** Przecięcie dużych osi widokowych.

- Parking naprzeciwko centrum handlowego Klif. Na podkreślonym widokowo załamaniu osi Okopowej pomiędzy Klif Tower a kompleksem Okopowa 56.
- **Skrzyżowanie ul. Radziwie z Obozową.** Otwarcie widokowe, skrzyżowanie większej osi widokowej z kilkoma mniejszymi. Zakończenie rytmicznego układu Wola Tower, brama zapraszająca na ul. Radziwie. Narożnik placu wyznaczonego przez MPO Obozowa 43.
- Anielewicza 30. Skrzyżowanie osi skośnego rytmu czterech punktowców mieszkaniowych z osią ul. Anielewicza.
- **Błękitny wieżowiec.** Dominanta wysokościowa, zamknięcie kilku osi widokowych. Sprzężenie Placu Bankowego w osi jego północno-wschodniej pierzei.
- **Teatr Wielki.** Dominanta przestrzenna w osi Placu Teatralnego. Zamknięcie kilku osi widokowych. Na przedłużeniu zielonej pierzei Ogrodu Saskiego.
- **Skrzyżowanie Al. Jana Pawła II i Al. Solidarności.** Przecięcie dwóch dużych osi widokowych oznaczone podwyższeniem zabudowy w narożnikach.
- Al. Solidarności 127. Naprzeciwko Sądu Okręgowego w Warszawie.
- **Elektrownia Powiśle.** Lokalizacja na szczycie niskiego wzgórza (o wiele czytelniejsza w przestrzeni cech niż w rzeczywistości). Na zamknięciu kilku drobnych osi widokowych (m.in. osiedla Nowe Powiśle).
- Część ul. Leszczyńskiej między numerami 1a i 4. Na przecięciu przedłużenia osi ul. Elektrycznej z osią ul. Leszczyńskiej.
- **Hala Gwardii.** Zakończenie osi Saskiej, przecięcie z osią ul. Zimnej i osią ścieżki pieszej w Parku Mirowskim (od ul. Grzybowskiej).
- **Skrzyżowanie Górczewskiej i Płockiej.** Przecięcie dwóch dużych osi widokowych. Podkreślone dodatkowo przez skośnie usytuowany budynek Płocka 41. Skośne elewacje wzdłuż skrzyżowania wyznaczają pierzeje pięciokątnego placu.
- **Warsaw Trade Tower.** Dominanta wysokościowa na zamknięciu wielu dużych osi widokowych, dodatkowo podkreślona przez łuk ul. Okopowej. Wraz z Warsaw Spire i Warsaw Unit wyznaczają warszawski „Isengard”.
- **Wieżowiec Cosmopolitan.** Dominanta wysokościowa w narożniku trójkątnego placu Grzybowskiego. Zamknięcie osi ul. Emilii Plater i Próżnej.
- **Prudential.** Dominanta wysokościowa, sprzężenie placu Powstańców Warszawy.

- Chopin Motel One. Narożnik kwartału na szczycie skarpy przy kładce pieszej Muzeum Fryderyka Chopina.
- **Tamka 49.** Dominanta wysokościowa na zamknięciu osi ul. Karasia-Kopernika.
- Centrum placu wyznaczonego przez ul. Kopernika, Kopczyńskiego, Tamkę i wieżowiec Tamka 40. Obecnie rondo.
- **Skrzyżowanie Wolskiej i Płockiej.** Przecięcie dużych osi widokowych. Brama ul. Płockiej wyznaczona przez szczyt wysokościowej zabudowy mieszkaniowej Skierniewickiej 21 i Płockiej 17. Stacja metra Płocka.
- **Warsaw Spire.** Dominanta wysokościowa widoczna z wielu otwarć widokowych. Wraz z Warsaw Trade Tower i Warsaw Unit wyznaczają warszawski „Isengard”.
- **Młotek przy Smolnej 9.** Dominanta wysokościowa na skarpie. Zamknięcie osi ul. Smolnej i ul. Czerwonego Krzyża.
- Dworzec Warszawa Powiśle. Przecięcie osi kolejowej z Alejami Jerozolimskimi. Zakończenie mostu Poniatowskiego.
- **Skrzyżowanie Żelaznej i Prostej.** Przecięcie dużych osi widokowej. Środek osi poprzecznej skweru Nelsona Mandeli. Zaakcentowany wieżowcem Mennica Legacy Tower.
- *Pałac Kultury i Nauki.* Dominanta przestrzenna, przecięcie osi widokowych. Centrum placu wyznaczonego przez ul. Marszałkowską, Świętokrzyską, Emilii Plater i Aleje Jerozolimskie.
- **Złota 44.** Dominanta wysokościowa na zamknięciu kilku otwarć widokowych podkreślona przez geometrię kompleksu handlowo-biurowego Złote Tarasy.
- **Varso Tower.** Dominanta wysokościowa, najwyższy obecnie budynek w Warszawie. Widoczność z wielu otwarć widokowych.
- **Rondo Charlesa de Gaulle’a.** Przecięcie ważnych osi widokowych. Oś trapezoidalnego placu wyznaczonego przez poszerzenie ul. Nowy Świat. Co ciekawe, punkt wskazany przez sieć dokładnie pokrywa się z lokalizacją słynnej Palmy Joanny Rajkowskiej (Rajkowska, 2002); sama palma nie znajdowała się w mapie cech widzianych przez sieć neuronową.
- **Warsaw Unit.** Dominanta wysokościowa przy Rondzie Daszyńskiego. Widoczna z wielu otwarć widokowych. Wraz z Warsaw Spire i Warsaw Trade Tower wyznacza warszawski „Isengard”.

- Adgar Renaissance Tower. Dominanta wysokościowa. Zamknięcie kilku mniejszych osi widokowych. Naprowadzenie przez bramę ul. Łomnickiego.
- **Novotel Warszawa Centrum.** Dominanta wysokościowa podkreślona przez rondo Romana Dmowskiego będące przecięciem ważnych osi widokowych. Pierwszy, narożny wieżowiec od strony południowo-wschodniej wyróżniający się na tle otaczających go niższych kwartałów.
- **Pomnik Wincentego Witosa na Placu Trzech Krzyży.** Zakończenie osi ul. Nowy Świat i ul. Bolesława Prusa. Podkreślony kompozycyjnie uskok w Placu Trzech Krzyży utworzony przez wycofaną elewację Ośrodka Rozwoju Edukacji w Warszawie.
- **Hotel Marriott.** Dominanta wysokościowa. Swoisty układ podwójny z Oxford Tower podkreślony przez identyczne proporcje.
- Roma Office Center. Dominanta wysokościowa ukryta za zabudową ul. Nowogrodzkiej i Poznańskiej.
- Biurowiec Warsaw Vibe. Biurowiec w narożniku Towarowej i Kolejowej.
- **Oxford Tower.** Dominanta wysokościowa. Swoisty układ podwójny z Hotelem Marriott podkreślony przez identyczne proporcje.
- **Atlas Tower.** Dominanta wysokościowa podkreślona przez rondo na placu Zawiszy. Pierwszy wieżowiec od strony południowo-zachodniej. Wyeksponowany widokowo przez płaszczyznę Warszawskich Filtrów.
- Skrzyżowanie ul. Dalekiej i Grójeckiej, pierzeja zachodnia.
- Skrzyżowanie ul. Dalekiej i Grójeckiej, pierzeja wschodnia.
- **Północne otwarcie Placu Konstytucji.** Skrzyżowanie osi widokowych ul. Marszałkowskiej, Koszykowej i Pięknej. Brama utworzona przez pierzeje placu Konstytucji.
- **Centrum południowej części Placu Konstytucji.** Punkt centralny pomiędzy południowymi osiami ul. Marszałkowskiej i Waryńskiego. Naprowadzenie przez oś północnej części ul. Marszałkowskiej i Śniadeckich.
- **Ochota Residence.** Dominanta wysokościowa podkreślona przez ostry narożnik i naprowadzenie widokowe ul. Kaliskiej.
- **Eurocentrum Alfa.** Najwyższy z wieżowców bramy zachodniej Warszawy. Obecnie pierwszy od strony południowej w rytmicznym układzie 10 brył biurowców o zbliżonych do siebie proporcjach.

- Grójecka 19/25. Budynek mieszkalny o charakterystycznej bryle na planie litery X. Lokalna dominanta wysokościowa na skrzyżowaniu ul. Niemcewicza i Grójeckiej.
- **Skrzyżowanie Al. Jerozolimskich i ul. Sokołowskiego „Grzymały”.** Charakterystyczne załamanie i jednocześnie poszerzenie Al. Jerozolimskich kierujące uwagę na ścianę wieżowców bramy zachodniej. Przecięcie ważnych osi widokowych.
- **Biurowiec West Station II.** Zaakcentowany przez otwarcie widokowe Sokołowskiego „Grzymały” i obecną przerwę w zabudowie bramy zachodniej między West Station II a Erocentrum Alfa.
- Skrzyżowanie ul. Sokołowskiego „Grzymały”, Barskiej i Szczęśliwickiej. Przecięcie osi widokowych. Podkreślenie przez skośną elewację budynku mieszkalnego Szczęśliwicka 8 stanowiącą płaszczyznę zginającą oś widokową ul. Barskiej.
- **Wejście do przejścia podziemnego między Parkiem Zachodnim a Dworcem Zachodnim.** Naprowadzenie przez osie ścieżek Parku Zachodniego. Główne dojście piesze na Dworzec Zachodni. Skrzyżowanie z osią Al. Jerozolimskich.
- Skrzyżowanie ul. Nowowiejskiej i Waryńskiego. Przecięcie osi widokowych. Skrzyżowanie między Placem Politechniki i Placem Zbawiciela.
- **Skrzyżowanie Al. Ujazdowskich, Al. Armii Ludowej, ul. Koszykowej i Al. Szucha.** Skrzyżowanie wielu ważnych osi widokowych przy Placu Na Rozdrożu. Podkreślone przez zetknięcie zabudowy z zieloną pierzeją Łazienek, Ogrodu Botanicznego i Parku Ujazdowskiego.
- Focus. Lokalna dominanta przestrzenna podkreślona przez ścieżki Pola Mokotowskiego.
- Przejścia podziemne skrzyżowania Al. Niepodległości i Al. Armii Ludowej. Wielopoziomowe skrzyżowanie ważnych osi widokowych z podziemnymi przejściami łączącymi poszczególne poziomy i narożniki skrzyżowania.
- **Pomnik Lotnika.** Skrzyżowanie ul. Wawelskiej, Uniwersyteckiej i Raszyńskiej. Charakterystyczne załamanie ul. Raszyńskiej. Plac z rondem zaakcentowany przez ostre narożniki ul. Wawelskiej i Uniwersyteckiej oraz ul. Uniwersyteckiej i Raszyńskiej.
- Skrzyżowanie ul. Grójeckiej, Kopińskiej i Wawelskiej. Przecięcie osi widokowych.
- **Akademik Riviera.** Dominanta przestrzenna na skrzyżowaniu Al. Armii Ludowej i ul. Waryńskiego. Ekspozycja widokowa z Pola Mokotowskiego. Naprowadzenie przez oś ul. Oleandrów i ścieżki Pola Mokotowskiego.

- **Wieżowiec Plac Unii.** Dominanta wysokościowa sprzęgająca Plac Unii Lubelskiej. Naprowadzenie widokowe ul. Puławskiej. Otwarcie widokowe od strony skrzyżowania Batorego i Waryńskiego.
- **Plac Unii Lubelskiej.** Jeden z placów gwiazdzistych założenia ujazdowskiego. Przecięcie kilku osi widokowych. Silny punkt centralny.

Załącznik 7: Tabela rozmiarów warstw sieci eksperymentu głównego i liczba trenowalnych parametrów

Całkowita liczba parametrów:	52 851 908			
Generator:	50 215 299			
Dyskryminator:	2 636 609			
Generator, lista warstw				
Liczba filtrów i progów	1d	2d	3d	Parametry
64	4	4	3	3136
128	4	4	64	131200
256	4	4	128	524544
512	4	4	256	2097664
512	4	4	512	4194816
512	4	4	512	4194816
512	4	4	512	4194816
512	4	4	512	4194816
512	4	4	512	4194816
512	4	4	1024	8389120
512	4	4	1024	8389120
512	4	4	512	4194816
256	4	4	1024	4194560
128	4	4	512	1048704
64	4	4	256	262208
3	4	4	128	6147

Dyskryminator, lista warstw				
Liczba filtrów i progów	1d	2d	3d	Parametry
64	4	4	6	6208
256	4	4	128	524544
512	4	4	256	2097664
1	4	4	512	8193

Załącznik 8. Wybór ciekawszych baz danych, które można wykorzystać w trenowaniu architektonicznych sieci neuronowych.

Wybór na podstawie przeprowadzonej kwerendy stanu wiedzy.

Nazwa	Źródło	Opis
Cyfrowa Kolekcja Muzeum Architektury we Wrocławiu	https://ma.wroc.pl/pl/kolekcja/	Zdigitalizowane przykłady tysięcy dzieł architektonicznych podzielonych na architekturę międzywojenną i współczesną
SpaceNet Challenge Datasets	https://spacenet.ai/datasets/	Publiczna baza danych zawierająca 67 tysięcy kilometrów kwadratowych oznaczonych zdjęć satelitarnych (m.in. mapy rozwoju urbanistycznego w czasie)
Sentinel 2	https://dataspace.copernicus.eu/analyse/apis	Otwarte dane satelitarne Europejskiej Agencji Kosmicznej systemu Copernicus z dostępnymi serwisami API w języku Python.

Numeryczny model pokrycia terenu Geoportal	https://www.geoportal.gov.pl/en/dane/numeryczny-model-pokrycia-terenu	Dane dostępne przez geoportal, reprezentacja powierzchni terenu wraz z obiektami wystającymi ponad powierzchnię (budynki, drzewa, infrastruktura...)
MexCulture142	https://github.com/montoyaobeso/mexculture142	Zbiór 284 budynków meksykańskiego dziedzictwa kulturowego podzielonych na 142 klasy
Architectural Style Classification Dataset	https://sites.google.com/site/zhexuutssjtu/projects/arch https://www.kaggle.com/datasets/wwymak/architecture-dataset	Zbiór 5000 zdjęć obiektów architektonicznych podzielonych na 25 klas względem stylu architektonicznego
UAVid Semantic Segmentation Dataset	https://uavid.nl/	Zbiór filmów lotniczych nakręconych w mieście przez autonomiczne drony. Filmy posegmentowane semantycznie na 8 klas.
Mapster	http://igrek.amzp.pl/	Zbiór archiwalnych materiałów kartograficznych w różnych formatach
Arcbazar	arcbazar.com/	Zbiór cyfrowych, konkursowych projektów architektonicznych w formie BIM/CAD. Większość projektów płatna.
Laion-400	https://laion.ai/laion-400-open-dataset/	400 milionów par obraz-opis (wiele przykładów architektonicznych i urbanistycznych)

Laion-5B	https://laion.ai/laion-5b-a-new-era-of-open-large-scale-multi-modal-datasets/	5 miliardów par obraz-opis (wiele przykładów architektonicznych i urbanistycznych)
ROBIN Dataset	https://github.com/gesstalt/ROBIN	Zbiór 510 rzutów architektonicznych mieszkań podzielonych względem liczby pomieszczeń
Harmonized Landsat and Sentinel-2	https://hls.gsfc.nasa.gov/hls-data/	Zbiór danych satelitarnych NASA i USGS
Open Buildings Dataset	https://www.kaggle.com/datasets/paultimothymooney/open-buildings	Wielkoskalowy zbiór obrysów 512 milionów budynków na zdjęciach satelitarnych Afryki
Modern Architecture	https://www.kaggle.com/datasets/tompaulat/modernarchitecture	Zbiór 100 tysięcy zdjęć obiektów architektonicznych oznaczonych względem typu budynku i przedstawionego fragmentu
UrbanSound8K	https://www.kaggle.com/datasets/chrisfilo/urbansound8k	Zbiór 8732 oznaczonych nagrań odgłosów miejskich
Urban Segmentation - ISPRS	https://www.kaggle.com/datasets/aletbm/urban-segmentation-isprs	Zbiór zdjęć satelitarnych poddanych semantycznej segmentacji dla 3 miast: Poczdam, Toronto i Vaihingen

Dodatkowe materiały cyfrowe związane z niniejszą rozprawą doktorską znajdują się w repozytorium: <https://github.com/TomaszDzieduszynski>