



Imię i nazwisko studenta: Bartosz Bieliński  
Nr albumu: 165430  
Studia pierwszego stopnia  
Forma studiów: stacjonarne  
Kierunek studiów: Automatyka i Robotyka  
Profil: Systemy decyzyjne i robotyka

Imię i nazwisko studenta: Piotr Winkler  
Nr albumu: 165504  
Studia pierwszego stopnia  
Forma studiów: stacjonarne  
Kierunek studiów: Automatyka i Robotyka  
Profil: Systemy decyzyjne i robotyka

## **PRACA DYPLOMOWA INŻYNIERSKA**

Tytuł pracy w języku polskim: Zastosowanie sieci neuronowych do edycji obrazów

Tytuł pracy w języku angielskim: Application of neural networks for image editing

Potwierdzenie przyjęcia pracy	
Opiekun pracy	Kierownik Katedry/Zakładu (pozostawić właściwe)
<i>podpis</i>	<i>podpis</i>
dr inż. Mariusz Domżalski	

Data oddania pracy do dziekanatu:

**POLITECHNIKA GDAŃSKA**

Katedra Systemów Decyzyjnych i Robotyki

**PRACA INŻYNIERSKA**

**Zastosowanie sieci neuronowych do edycji  
obrazów**

**Autorzy**

Piotr Winkler

Bartosz Bieliński

Gdańsk 2019

## **STRESZCZENIE**

Tematem pracy jest zbadanie możliwości zastosowania sieci neuronowych do edycji obrazów. Głównym celem było stworzenie narzędzi opartych na wyuczonych sieciach neuronowych, służących do odpowiedniego przetwarzania i modyfikowania obrazu. <Zakres pracy >

<Zastosowane metody badań ><wyniki ><najważniejsze wnioski >

**Słowa kluczowe:** sieć neuronowa, przetwarzanie obrazu, konwolucyjna sieć neuronowa, splotowa sieć neuronowa, model generatywny, głęboka sieć neuronowa,

**Dziedzina nauki i techniki zgodna z OECD:** Nauki inżynierijne i techniczne, Elektrotechnika, elektronika, inżynieria informatyczna, Sprzęt komputerowy i architektura komputerów

## **ABSTRACT**

The subject of the work is research on the possibilities of applying neural networks for image editing. The main goal was to create tools based on trained neural networks used for appropriate processing and modifying of images. <Zakres pracy >

<Zastosowane metody badań ><wyniki ><najważniejsze wnioski >

**Keywords:** neural network, image processing, convolutional neural network, generative adversarial network, deep neural network

**Field of science and technology in accordance with the requirements of the OECD:** Engineering and technology, Electrical engineering, Electronic engineering, Information engineering, Computer hardware and architecture

## **WYKAZ WAŻNIEJSZYCH OZNACZEŃ I SKRÓTÓW**

ANN (ang. Artificial Neural Network) - Sztuczna sieć neuronowa

DNN (ang. Deep Neural Network) - Głęboka sieć neuronowa

CNN (ang. Convolutional Neural Network) - Splotowa sieć neuronowa

FCN (ang. Fully Convolutional Network) - Warstwa gęsta

GAN (ang. Generative Adversarial Network) - Sieci o modelu generatywnym

VAE (ang. Variational Autoencoder) - Autoenkodery wariacyjne

ReLU (ang. Rectified Linear Unit) - Jednostronnie obcięta funkcja liniowa

BatchNorm (ang. Batch Normalization) - Normalizacja zbioru danych pogrupowanych w pakiety

YUV - Model barw, w którym Y odpowiada za luminancję obrazu, a UV są to dwa kanały chrominacji i kodują barwy

IcGAN (ang. Invertible conditional Generative Adversarial Network) - Odwracalny warunkowy model generatywny

cGAN (ang. conditional Generative Adversarial Network) - warunkowy model generatywny

# **Spis treści**

<b>WYKAZ WAŻNIEJSZYCH OZNACZEŃ I SKRÓTÓW</b>	<b>4</b>
<b>1 Wstęp i cel pracy</b>	<b>6</b>
1.1 Cel pracy . . . . .	6
1.2 Dotychczasowe dokonania . . . . .	6
1.3 Założenia projektowe . . . . .	7
1.4 Układ pracy . . . . .	7
<b>2 Podstawy teoretyczne</b>	<b>8</b>
2.1 Sieci splotowe . . . . .	8
2.2 FCN . . . . .	10
2.3 Modele generatywne . . . . .	10
2.4 Autoenkodery . . . . .	11
<b>3 Przegląd rozwiązań</b>	<b>12</b>
3.1 Colorful image colorization . . . . .	12
3.2 Image Style Transfer Using Convolutional Neural Networks . . . . .	13
3.3 Invertible Conditional GANs for image editing . . . . .	14
3.4 Neural photo editing . . . . .	15
<b>4 &lt;Opis, implementacja oraz testy naszego rozwiązania &gt;</b>	<b>17</b>
<b>5 Podsumowanie</b>	<b>17</b>
<b>Bibliografia</b>	<b>18</b>
<b>Spis rysunków</b>	<b>19</b>
<b>Spis tabel</b>	<b>19</b>

# 1 WSTĘP I CEL PRACY

Sztuczne sieci neuronowe sięgają swym początkiem lat 40. XX wieku. Historia ich rozwoju odnotowała trzy okresy, w których rozwiązania te odbijały się szerokim echem w środowisku naukowym.

Pierwszy model neuronu, a potem perceptron zapoczątkowały rozwój tej dziedziny nauki, jednak pierwsze sieci jednowarstwowe nie były w stanie rozwiązywać złożonych problemów. Przeszkodę nie do pokonania stanowiła dla nich nawet prosta funkcja logiczna XOR. Z tego powodu badania sieci neuronowych zostały na długi czas porzucone.

Pojawienie się algorytmu wstępnej propagacji błędów pozwalającego skutecznie uczyć wielowarstwowe sieci neuronowe ponownie wzmożło zainteresowanie tematem, jednak tym razem na drodze postępowi stanęły ograniczenia technologiczne ówczesnych czasów.

Wreszcie wraz z nadaniem XXI wieku postępujący rozwój komputerów oraz internetu umożliwił sztucznym sieciom neuronowym rozwinięcie skrzydeł. Wejście w erę "big data" otworzyło dostęp do olbrzymich zbiorów danych niezbędnych do treningu sieci, a pojawienie się wysokowydajnych jednostek obliczeniowych pozwoliło znacznie ten proces przyspieszyć.

Zapoczątkowany w ten sposób rozwój trwa do dnia dzisiejszego. Sztuczne sieci neuronowe odnajdują zastosowanie w wielu dziedzinach życia i nauki. Grają w gry, przeprowadzają symulacje, przewidują i prognozują zachowania rynku, czy pogody, analizują i przetwarzają obrazy cyfrowe.

Z punktu widzenia niniejszej pracy największe znaczenie ma oczywiście ostatni z wymienionych punktów. Zdefiniowanie sieci neuronowych, jako matematycznych modeli obliczeniowych ujawnia ich naturalne predyspozycje do pracy na obrazach cyfrowych. W praktyce stanowią one bowiem zbiór liczb, wartości poszczególnych pikseli, który sieć neuronowa jest w stanie analizować, przetwarzać i modyfikować.

## 1.1 Cel pracy

Celem pracy jest stworzenie szeregu narzędzi programistycznych oferujących szeroki wachlarz możliwości edytowania obrazu. Narzędzia te oparte mają być na technologii sieci neuronowych. W szczególności przetestowana będzie skuteczność rozwiązań dedykowanych do przetwarzania obrazów, takich jak sieci konwolucyjne albo modele generatywne.

Po opracowaniu tychże narzędzi, opisane zostaną efekty pracy oraz zbadana zostanie skuteczność sieci neuronowych jako rozwiązania nakreślonej problematyki. Omówione zostaną także wykorzystane architektury zaimplementowanych modeli, wykorzystane funkcje kosztu, metody aktualizowania wag sieci oraz przebiegi treningu modeli.

## 1.2 Dotychczasowe dokonania

<Syntetyczny opis dotychczasowych dokonań w danej tematyce? >

### ***1.3 Założenia projektowe***

Główym założeniem pracy było zaprojektowanie i zaimplementowanie narzędzi programistycznych służących do edycji obrazu. Narzędzia te muszą wykorzystywać do swoich celów nauuczone sieci neuronowe. Na podstawie jakości działania tychże narzędzi, oceniona zostanie ich rzetelność oraz skuteczność.

Następnie przeprowadzona zostanie analiza słuszności zastosowania sieci neuronowych jako rozwiązania przedstawionej problematyki. Uzyskane rozwiązanie zostanie także zestawione ze znanyymi algorytmami do edycji obrazu nie opierającymi się na technologii sieci neuronowych.

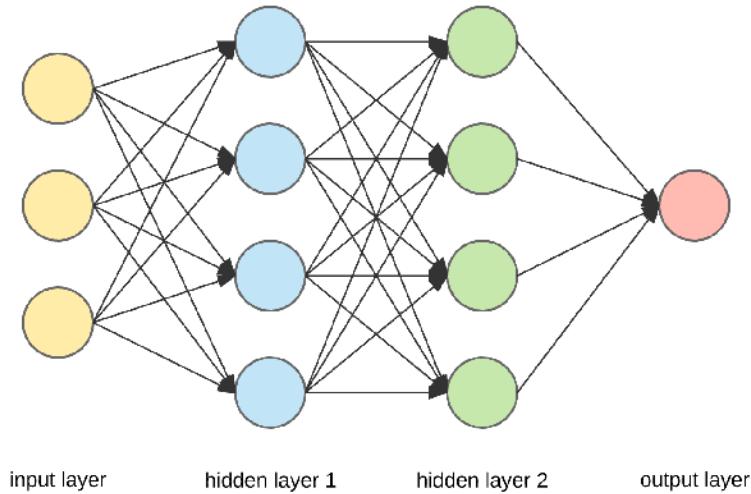
### ***1.4 Układ pracy***

W pierwszym rozdziale przedstawiono zarys rozwoju Sieci Neuronowych na przestrzeni lat oraz niezbędne podstawy teoretyczne związane z przedstawioną problematyką i wybranym dla niej rozwiązaniem.

W kolejnym rozdziale dokonano przeglądu już istniejących rozwiązań, opisano ich przeznaczenie, sposób działania oraz uzyskane rezultaty. Oceniono także wpływ danego rozwiązania na rozwój sieci neuronowych w dziedzinie przetwarzania i edytowania obrazów.

## 2 PODSTAWY TEORETYCZNE

W dzisiejszych czasach sieci neuronowe zajmują ważną pozycję na rynku narzędzi do edycji obrazu. Jest to głównie spowodowane ich umiejętnością do reprodukowania i modelowania niesłiniowych procesów, a także nowoczesnymi technikami przetwarzania plików graficznych. Jednak pierwsze architektury ANN (ang. artificial neural network) nie nadawały się do przetwarzania grafik. Było to częściowo spowodowane faktem, że obrazy, będące w rzeczywistości macierzami wartości pikseli, ciężko było skutecznie podać na wejście typowych architektur DNN (ang. deep neural network) zbudowanych pierwotnie z wielu warstw ukrytych, pomiędzy którymi połączenia są na zasadzie każdy z każdym oraz mają swoje wagi podlegające modyfikacji w trakcie procesu uczenia. Taka struktura pokazana została na Rysunku 2.1. Obrazy o niskiej rozdzielcości można było przekształcić w wektory wartości poszczególnych pikseli i w takiej postaci podawać na wejście sieci, jednak w przypadku obrazów o wyższej rozdzielcości to rozwiązanie, ze względu na znaczącą długość powstałych wektorów, nie oferowało dobrych rezultatów. Dopiero nowe architektury sieci spowodowały przełom w tej dziedzinie. Wprowadzenie do najistotniejszych i najciekawszych z nich zostanie przedstawione w poniższym rozdziale, a także rozwinięte w dalszej części tej pracy.



Rysunek 2.1: Struktura DNN

### 2.1 Sieci splotowe

Neuronowe sieci splotowe (CNN ang. convolutional neural network) stanowią podstawową strukturę w zakresie przetwarzania i analizowania obrazów cyfrowych. Są to sieci o hierarchicznej strukturze stanowiące podwaliny większości klasyfikatorów, detektorów, czy sieci segmentujących.

Autorzy jednego z artykułów traktujących o sieciach splotowych [3] opisują je następująco:

*'CNN to skuteczny algorytm poznawczy, stosowany powszechnie przy rozpoznawaniu wzorców i przetwarzaniu obrazów. Posiada wiele cech, takich jak prosta struktura, mniej parametrów treningowych, czy zdolność do adaptacji. CNN stały się gorącym tematem w zakresie analizy głosu i rozpoznawania obrazu. Ich struktura oparta na podziale wag czyni je bardziej podobnymi do biologicznych sieci neuronowych. Redukuje to złożoność modelu sieci oraz liczbę wag'.*

Na CNN składają się zazwyczaj trzy rodzaje warstw, z których każda posiada inne cechy.

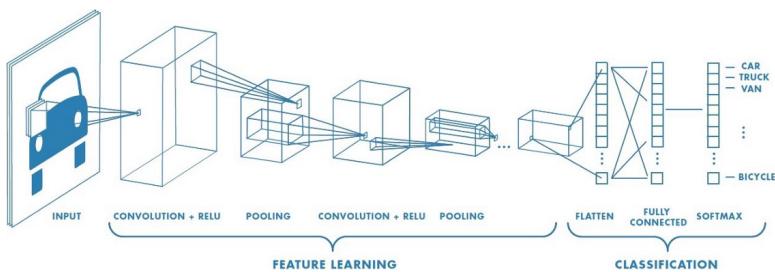
Podstawową warstwę stanowi warstwa splotowa. Składa się ona ze zbioru filtrów (neuronów) odpowiedzialnych za ekstrakcję cech z analizowanych obrazów poprzez dokonanie operacji konwolucji na obrazie poprzez przesuwanie zestawu filtrów wzduż niego. Na wyjściu filtrów otrzymuje się macierze o mniejszej rozdzielcości reprezentujące wyniki operacji konwolucji w danym punkcie. Każda kolejna warstwa splotowa wydobywa z obrazu cechy o wyższych poziomach abstrakcji bazując na wynikach obliczeń poprzednich warstw tego rodzaju. Dzięki temu procesowi kolejne warstwy filtrów uczą się rozpoznawać kluczowe cechy na obrazie, od drobnych elementów takich jak krawędzie albo kształty po bardziej złożone takie jak części ciała albo całe obiekty. Filtry te są zazwyczaj inicjowane losowymi wartościami i w miarę trenowania, dopasowują swoje parametry do wybranej problematyki.

Drugim istotnym elementem sieci splotowych jest warstwa poolingu. Może zostać opisana następująco [4]:

*'We wszystkich przypadkach pooling pomaga uczynić reprezentację w przybliżeniu niezmiennej w stosunku do małych tłumaczeń danych wejściowych. Niezmienność wobec tłumaczenia oznacza, że jeśli poddamy dane wejściowe nieznacznej translacji, to wartość większości wyników poddanych poolingu nie ulegnie zmianie'.*

Końcowy element CNN w większości przypadków stanowią warstwy gęste (FCN ang. Fully Convolutional Network). Odpowiadają one za dokonanie odpowiedniej klasyfikacji obrazu na podstawie danych dostarczonych przez warstwy poprzedzające. Są przez to nieodzowne w przypadku zadań związanych z wszelkiego rodzaju klasyfikacją obrazów.

Wymienione tutaj elementy składowe sieci splotowych mogą przyjmować różne rozmiary i występować w różnych konfiguracjach, co przedstawiono na poniższym Rysunku 2.2.

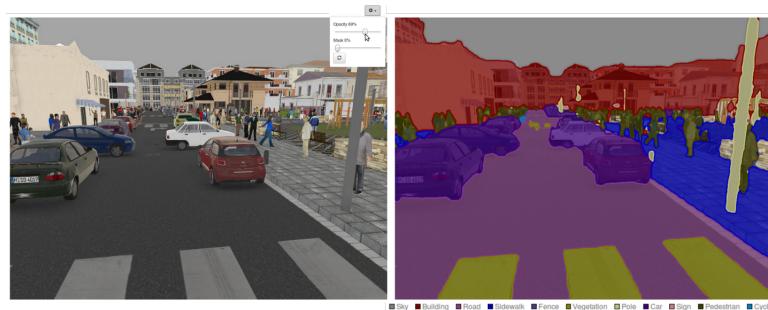


Rysunek 2.2: Przykładowa struktura CNN

Zapewnia to szerokie pole do eksperymentów i sprawia, że sieci te zdolne są rozwiązywać złożone, różnorodne problemy z wielu dziedzin codziennego życia.

## 2.2 FCN

Jednym z kluczowych problemów, jakie stawia przed badaczami edycja obrazów jest zagadnienie segmentacji semantycznej. Klasyczna klasyfikacja, polegająca na przypisywaniu obrazów do odpowiednich grup tematycznych, jest w tym przypadku sprowadzana do poziomu pojedynczych pikseli. Oznacza to, że sieci neuronowe przeznaczone do tego zadania są w stanie dokonać klasyfikacji dla każdego pojedynczego piksela analizowanego obrazu. Na tej podstawie uzyskiwany jest podział na segmenty, z których każdy reprezentuje inną klasę obiektów, jak na poniższym Rysunku 2.3.



Rysunek 2.3: Segmentacja semantyczna

W większości modele odpowiadające za przeprowadzanie segmentacji składają się z szeregowego połączenia enkodera oraz dekodera. Enkoder jest zazwyczaj pre-trenowaną siecią neuronową przeznaczoną do klasyfikowania obrazów. Dekoder odpowiada za semantyczne rzutowanie cech w niskiej rozdzielcości, wyuczonych przez enkoder, na wysoką rozdzielcość samych pikseli tworząc wspomniany wcześniej podział segmentowy.

FCN (ang. Fully Convolutional Networks) stanowią szczególny rodzaj sieci neuronowych przeznaczonych do segmentacji obrazów. Składają się one wyłącznie z kombinacji warstw splotowych oraz poolingu. Są w stanie przetwarzać obrazy o dowolnej, zmiennej wielkości, w odróżnieniu od innych typów modeli, w których zastosowanie warstw gęstych (FCL) wymusza z góry ustalone rozmiary danych wejściowych.

Naprzemienne przepuszczanie obrazów przez wspomniane warstwy splotowe oraz pooling może powodować niską rozdzielcość wyjściowych rezultatów pracy tych sieci oraz rozmycie granic poszczególnych obiektów. Z tego powodu w nowoczesnych rozwiązaniach stosuje się dodatkowe mechanizmy zapobiegające tego typu trendom.

## 2.3 Modele generatywne

Koncepcja modeli generatywnych, w skrócie GANów, przedstawiona została w 2014 roku przez Iana Goodfellow oraz jego współpracowników na uniwersytecie w Montrealu [1]. Modele te stanowią połączenie dwóch głębokich sieci neuronowych działających przeciwstawnie do siebie nawzajem.

Pierwsza sieć to tak zwany generator. W odniesieniu do tematu pracy, jego działanie polega na generowaniu nowych obrazów, lub ich fragmentów na podstawie wektora szumów.

Obrazy te przekazywane są, równolegle z zestawem obrazów prawdziwych, do dyskryminatora stanowiącego drugą część modelu GAN. Działanie tej sieci neuronowej polega na określeniu (w skali 0 do 1), w jakim stopniu produkty wyjściowe generatora odpowiadają obrazom rzeczywistym.

W opisany modelu występuje zatem podwójna pętla sprzężenia zwrotnego. Dyskryminator określa autentyczność obrazów porównując je ze zdefiniowaną odgórnie bazą danych. Z kolei generator otrzymuje informację o skuteczności swojego działania ze strony dyskryminatora.

Model generatywny znajduje się w stanie ciągłego konfliktu. Generator dąży do jak najdokładniejszego fałszowania obrazów w celu oszukania dyskryminatora, którego celem jest z kolei jak najdokładniejsze wykrywanie podróbek. Obie sieci neuronowe nieustannie dążą do osiągnięcia przewagi nad rywalem w procesie treningu. Ciągła rywalizacja sprawia, że zarówno generator, jak i dyskryminator zyskują coraz wyższą skuteczność działania.

W praktyce modele generatywne są w stanie naśladować dowolną dystrybucję danych. Są w stanie kreować światy podobne do naszego w zakresie obrazu, dźwięku czy mowy. Można powiedzieć, że są to prawdziwi syntetyczni artyści.

#### **2.4 Autoenkodery**

### 3 PRZEGŁĄD ROZWIĄZAŃ

Na przestrzeni ostatnich paru lat pojawiło się wiele rozwiązań zastosowania sieci neuronowych do edycji obrazu, duża część z nich była przełomowa w swojej dziedzinie. Powstawały rewolucyjne architektury sieci oraz technologie z nimi związane. Takie cechy sieci jak niezwykła zdolność do generalizacji zdobytej wiedzy na nowe przypadki oraz olbrzymia elastyczność sprawiły, że znalazły one wiele rzeczywistych zastosowań.

W tym rozdziale skupiono się na przedstawieniu kilku interesujących rozwiązań dla omawianej problematyki.

#### 3.1 *Colorful image colorization*

Wraz z rozwojem sieci neuronowych, rosło zainteresowanie możliwościami zastosowania ich do kolorowania czarno-białych obrazów. Jedno z dostępnych rozwiązań tego zagadnienia zostało przedstawione przez grupę pracowników Uniwersytetu w Berkeley [5]. Zamiarem ich pracy było stworzenie modelu, który niekoniecznie odtwarza oryginalne barwy obrazu, ale generuje barwy prawdopodobne, zdolne przekonać ludzkiego obserwatora o autentyczności obrazu. Uzyskane rezultaty zostały przedstawione na Rysunku 3.1.



Rysunek 3.1: Efekt kolorowanie czarno-białych zdjęć przez wytrenowany model.

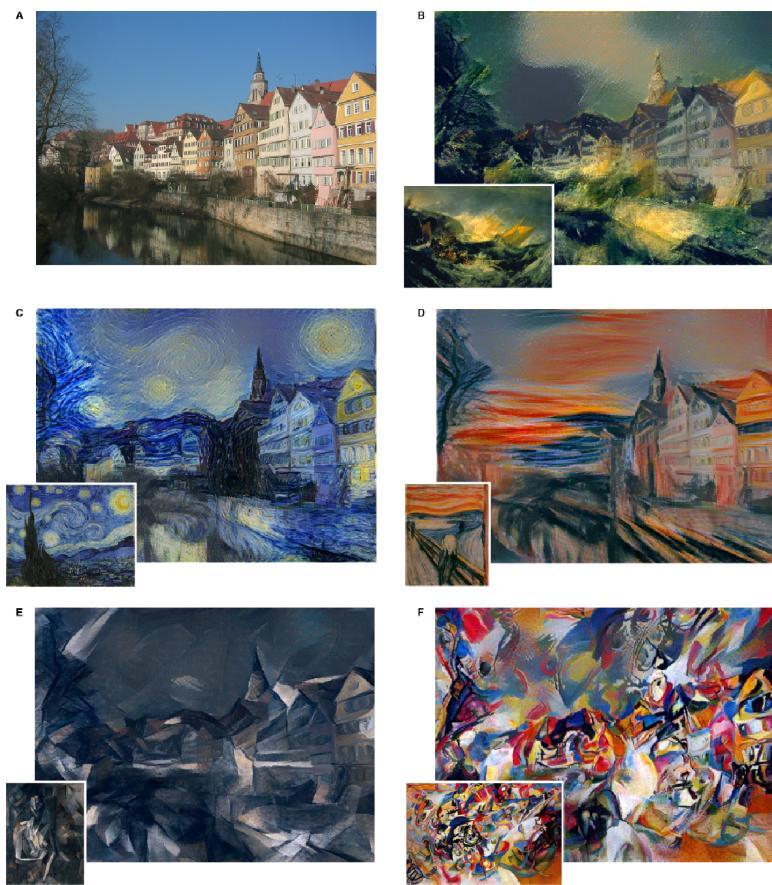
Wykorzystany model składa się z wielu warstw CNN, w których skład wchodzą warstwa filtrów konwolucyjnych, warstwa ReLU (ang. Rectified Linear Unit) oraz warstwa BatchNorm (ang. Batch normalization). Aby zapobiec utracie informacji przestrzennych, sieć nie posiada warstw łączących. Istotny był także sposób przygotowania zbioru danych do trenowania modelu. Obrazy ze zbioru uczącego były wpierw konwertowane do modelu YUV, a następnie kanał Y był podawany na wejście modelu, warstwy UV pełniły funkcję pożąданiej odpowiedzi w uczeniu nadzorowanym.

Ważnym aspektem zbadanym w artykule było także dobranie odpowiedniej funkcji kosztu. Nieuodpowiedni wybór skutkował desaturacją kolorowanych obrazów, jedną z potencjalnych przyczyn tego zjawiska może być tendencja sieci do tworzenia bardziej konserwatywnych odpowiedzi. Aby zniwelować ten efekt w modelu została zastosowana specjalna technika modyfikacji funkcji kosztu. Polega ona na przewidywaniu dystrybucji możliwych kolorów dla każdego piksela i zmienianiu kosztu dla modelu, w celu wyróżnienia rzadko spotykanych kolorów.

Powstałe rozwiązańe dowodzi olbrzymiego potencjału zastosowanie sieci neuronowych w dziedzinie pracy nad obrazami.

### 3.2 Image Style Transfer Using Convolutional Neural Networks

W roku 2016 został przedstawiony światu A Neural Algorithm of Artistic Style [6]. Wprowadził on przełom w dziedzinie przenoszenia stylu jednego obrazu na inny, a jego sukces opierał się na właściwym wykorzystaniu konwolucyjnych sieci neuronowych. Podstawą tego sukcesu było odkrycie przez Leona A. Gatys oraz jego współpracowników, że w CNN reprezentacja treści obrazu oraz jego stylu jest rozłączna. Umożliwia to wydobycie stylu przetwarzanego obrazu oraz połączenie go z treścią innego obrazu, czego dokonuje właśnie A Neural Algorithm of Artistic Style. Rezultaty takich operacji można zaobserwować na Rysunku 3.2



Rysunek 3.2: Obrazy będące kombinacją treści zdjęcia ze stylami kilku znanych dzieł sztuki.

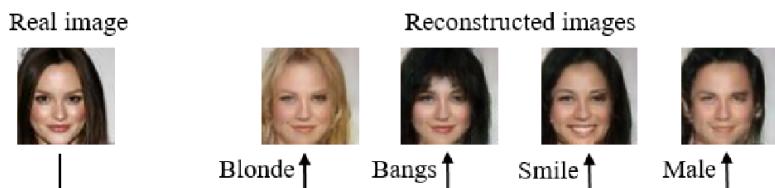
Do zbudowania modelu zostały użyte warstwy konwolucyjne oraz łączące z architektury VGG-Network [7], która została wytrenowana pod kątem rozpoznawanie obiektów i określania ich położenia. Dzięki temu sieć przetwarzając obraz tworzy jego reprezentację, która wraz z kolejnymi warstwami, przedstawia coraz wyraźniejszą informację o obiektach, a niekoniecznie o dokładnym wyglądzie obrazu. W modelu nie została użyta ani jedna warstwa gęsta, dzięki czemu na wyjściu możliwe jest otrzymanie dwuwymiarowego obrazu. Dla lepszej syntezy obrazów, w warstwach łączących zastosowano próbkowanie wartością średnią zamiast maksymalną. Takie zabiegi umożliwiają wyliczenie reprezentacji stylu z korelacji pomiędzy różnymi cechami w różnych warstwach konwolucyjnych.

Cały proces renderowania polega na odpowiednim zapisaniu w modelu treści oraz stylu obrazów otrzymanych przez wcześniejsze podanie na model tychże obrazów. Wpierw obraz, z którego pobierany jest styl, jest podawany na wejście sieci oraz przeliczany, reprezentacja stylu wyselekcjonowana z właściwych warstw jest przechowywana, obraz z treścią jest poddawany temu samemu procesowi, ale reprezentacja treści jest wyciągana z ostatnich warstw konwolucyjnych. W celu uzyskania fuzji obrazów, zapisane reprezentacje treści i stylu są zapisywane w tych warstwach modelu skąd zostały odczytane, a następnie na wejście podawany jest obraz składający się z losowego szumu białego. Następnie poprzez iteracyjną minimalizację funkcji kosztu, obraz wejściowy jest modyfikowany, co w rezultacie końcowym doprowadza do nałożenia zapisanego stylu na wczytaną treść.

A Neural Algorithm of Artistic Style jest świetnym przykładem, jak elastyczne mogą być interfejsy do modyfikacji obrazu oparte na technologii sieci neuronowych.

### 3.3 Invertible Conditional GANs for image editing

Edycja obrazów może być dokonywana na wielu różnych poziomach zaawansowania i abstrakcji, operacje takie jak nakładanie filtrów mogą być wykonywane przez proste algorytmy. Jednak w przypadku próby modyfikacji elementów na obrazie, algorytmy te nie będą w stanie dokonać semantycznych zmian ze względu na brak możliwości zrozumienia treści obrazu. Rozwiązanie tego problemu zostało przedstawione w postaci modelu IcGAN (ang. Invertible Conditional Generative Adversarial Network) w roku 2016 [8]. Zaprezentowany model był to enkoder z możliwością generowania wektora informacji o atrybutach obrazu połączony z warunkowym GAN zdolnym do kontrolowania cech generowanych obrazów na podstawie dodatkowej informacji warunkowej. Takie działanie umożliwia wprowadzania zmian w atrybutach generowanego obrazu uzyskiwanego na wyjście cGAN (ang. conditional Generative Adversarial Network). Rezultaty działania modelu można zaobserwować na Rysunku 3.3.



Rysunek 3.3: Obrazy generowane przez IcGAN.

Wykorzystany w IcGAN Ekonder w rzeczywistości składa się z dwóch podrzędnych Enkoderów, Enkoder  $E_z$  koduje wejściowy obraz do utajonego wektora  $z$  reprezentacji obrazu, natomiast Enkoder  $E_y$  generuje wektor informacji  $y$  oddających pewne kluczowe atrybuty obrazu. Enkody są trenowane z użyciem już wytrenowanego cGAN oraz obrazów rzeczywistych z etykietami ze zbioru uczącego. Zbadane zostały także różne podejścia interakcji między dwoma Enkoderami, wyróżnić można podejście, w którym Enkdodery są w pełni niezależne, podejście gdzie wyjście  $E_z$  jest zależne od wyjścia  $E_y$ , a także podejście gdzie  $E_z$  oraz  $E_y$  są połączone w jednej Enkoder o współdzielonych warstwach i dwóch wyjściach.

W przypadku cGAN możemy wyróżnić dwa najważniejsze czynniki, które trzeba mieć na uwadze. Pierwszym jest źródło wektora  $y$  podawanego na Generator. W przypadku Dyskryminatora  $y$  jest pobierany ze zbioru treningowego, jednakże w przypadku podawania tego samego wektora na Generator wystąpiła możliwość, że może dojść do niepożdanego przeuczenia modelu. Autorzy artykułu dokonali analizy tego rozwiązania, a także zbadali wydajność metody Bezpośredniej Interpolacji oraz Jądrowego estymatora gęstości. Wynikiem tych badań było stwierdzenie, że dla danej problematyki najlepiej sprawdza się podawanie wektora  $y$  ze zbioru uczącego, możliwość przeuczenia modelu została skomentowana następująco:

*'Jest to możliwe tylko, gdy informacje warunkowe są do pewnego stopnia unikatowa dla każdego obrazu. W tym przypadku, gdzie atrybuty obrazów są binarne, jeden wektor  $y$  może opisać wystarczająco duży i zróżnicowany podzbiór obrazów, zapobiegając nadmiernemu dopasowaniu się modelu do danego  $y$ .'*

Drugim czynnikiem jest warstwa Generatora i Dyskryminatora cGAN na którą podany jest wektor  $y$ . Guim Perarnau oraz jego współpracownicy ustalili, że najlepsze rezultaty otrzymuje się po podaniu wektora  $y$  na warstwę wejściową Generatora oraz pierwszą warstwę konwolucyjną Dyskryminatora.

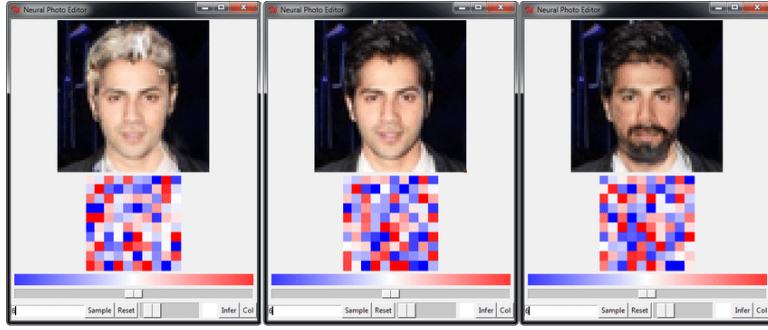
Ważnym spostrzeżeniem z analizy rozwiązania IcGAN jest obecność olbrzymiej ilości różnorodnych rozwiązań opartych na sieciach neuronowych, coraz to nowe architektury zostają wynalezione, aby udoskonalić zastosowania sieci neuronowych do przetwarzania i modyfikowania obrazów.

### 3.4 Neural photo editing

W 2017 roku Andrew Brock, Theodore Lim, J.M. Ritchie and Nick Weston zaprezentowali Neural Photo Editor [2], narzędzie do edytowania obrazu wyposażone w mechanizmy wykrywania kontekstu zmiany. Twórcy opisują swój twór następująco:

*'Interfejs wykorzystujący moc generatywnych sieci neuronowych do wprowadzania dużych, semantycznie spójnych zmian w istniejących obrazach.'*

Użytkowanie wygląda następująco: użytkownik pędzlem o określonym kolorze i rozmiarze maluje na wybranym obrazie, jednak zamiast zmieniać wartości pojedynczych pikseli, interfejs odczytuje kontekst edycji i wprowadza zmiany semantyczne w kontekście żądanej zmiany koloru. Efekt działania interfejsu został przedstawiony na Rysunku 3.4.



Rysunek 3.4: Efekt działania Neural Photo Editor.

Skuteczność NPE (Neural Photo Editor) polega na zastosowaniu IAN (ang. Introspective Adversarial Network), czyli sieci złożonej z połączonych VAE (ang. Variational Autoencoder) [9] oraz GAN, w taki sposób, że dekodująca sieć autoenkodera jest używana jako sieć generująca w GAN. Poprzez przechwytywanie przez model dalekosieżnych zależności, wykorzystanie bloku obliczeniowego bazującego na rozszerzonych splotach o współdzielonych wagach oraz dzięki zastosowaniu ulepszonej generalizacji, udało się osiągnąć dokładną rekonstrukcję obrazu bez strat na jakości detali.

Powstanie NPE utwierdza w przekonaniu, że aktualne możliwości sieci neuronowych do edycji obrazu znacznie przewyższają zwykłe algorytmy pod względem możliwości oraz uzależnienia od wkładu ludzkiego.

**4 <OPIS, IMPLEMENTACJA ORAZ TESTY NASZEGO ROZWIAZANIA >**

**5 PODSUMOWANIE**

## BIBLIOGRAFIA

- [1] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, Yoshua Bengio: *Generative Adversarial Networks*, ('2014)
- [2] Andrew Brock, Theodore Lim, J.M. Ritchie, Nick Weston: *NEURAL PHOTO EDITING WITH INTROSPECTIVE ADVERSARIAL NETWORKS*, arXiv ('2017)
- [3] Tianyi Liu, Shuang Sang Fang, Yuehui Zhao, Peng Wang, Jun Zhang: *Implementation of Training Convolutional Neural Networks*, arXiv ('2015), s.2.
- [4] Ian Goodfellow , Yoshua Bengio, Aaron Courville: *Deep Learning*, ('2016), s.342.
- [5] Richard Zhang, Phillip Isola, Alexei A. Efros: *Colorful Image Colorization*, arXiv ('2016)
- [6] Leon A. Gatys, Alexander S. Ecker, Matthias Bethge: *Image Style Transfer Using Convolutional Neural Networks*, ('2016)
- [7] Karen Simonyan, Andrew Zisserman: *Very Deep Convolutional Networks For Large-Scale Image Recognition*, ('2016)
- [8] Guim Perarnau, Joost van de Weijer, Bogdan Raducanu, Jose M. Álvarez: *Invertible Conditional GANs for image editing*, arXiv ('2016)
- [9] Diederik P. Kingma, Max Welling: *Auto-Encoding Variational Bayes*, arXiv ('2014)

## **Spis rysunków**

2.1	Struktura DNN . . . . .	8
2.2	Przykładowa struktura CNN . . . . .	9
2.3	Segmentacja semantyczna . . . . .	10
3.1	Efekt kolorowanie czarno-białych zdjęć przez wytrenowany model. . . . .	12
3.2	Obrazy będące kombinacją treści zdjęcia ze stylami kilku znanych dzieł sztuki. . . . .	13
3.3	Obrazy generowane przez IcGAN. . . . .	14
3.4	Efekt działania Neural Photo Editor. . . . .	16

## **Spis tabel**