

Politechnika Warszawska

WYDZIAŁ ELEKTRONIKI
I TECHNIK INFORMACYJNYCH



Instytut Automatyki i Informatyki Stosowanej

Praca dyplomowa inżynierska

na kierunku Automatyka i Robotyka

Semantyczna analiza środowiska przez robota usługowego

Piotr Hondra

Numer albumu 303752

promotor

mgr inż. Maciej Stefańczyk

WARSZAWA 2023

Semantyczna analiza środowiska przez robota usługowego

Streszczenie. Przetwarzanie obrazu ma wiele zastosowań. Jednym z nich jest semantyczna analiza środowiska. W tej pracy inżynierskiej zostanie przeprowadzona semantyczna analiza środowiska we wnętrzach. Do jej wykonania pozyskane zostaną informacje o przynależności pikseli do znaczeniowych grup (segmentacja semantyczna) oraz każdemu obrazowi zostanie przyporządkowana nazwa miejsca, które przedstawia (klasyfikacja sceny). Środowiska wewnętrzne stanowią unikalny zestaw wyzwań. Jednocześnie rozwiązywanie ich oddziennie wydaje się mało efektywne. Trudno wyobrazić sobie szeregowego odtwarzanie algorytmów w dziedzinie robotyki czy urządzeń Internetu Rzeczy, gdzie zasoby są bardzo ograniczone, a wymagania na szybkość odpowiedzi wysoko postawione. Implikuje to wybór zaawansowanych algorytmów, które najlepiej działają równolegle. Aktualnym trendem rozwiązań przetwarzania obrazu są głębokie sieci neuronowe, dzięki swojej wysokiej skuteczności. Co więcej, te sieci pozwalają się łączyć w architektury, umożliwiając jednoczesne wykonywanie obu wspomnianych wcześniej zadań. Takie podejście w literaturze jest określone jako uczenie wielozadaniowe. Ma ono szereg zalet. W tej pracy pokażę, że wspólna segmentacja semantyczna oraz klasyfikacja scen w środowiskach wewnętrznych jest korzystna. Dodatkowo zostaną przeprowadzone badania nad aktualnymi technikami głębokiego uczenia w kontekście uczenia wielu zadań. Ostatecznie wyniki wszystkich metod zostaną porównane i ocenione na dużym zbiorze danych scen wnętrz.

Słowa kluczowe: głębokie sieci neuronowe, uczenie wielozadaniowe, segmentacja semantyczna, klasyfikacja sceny

Semantic analysis of the environment by a mobile service robot

Abstract. Image processing has many applications. One of them is a semantic analysis of the environment. In this thesis, a semantic analysis of the environment in the interior will be carried out. In order to perform it, information about the belonging of pixels to meaningful groups will be extracted (semantic segmentation), and each image will be assigned the name of the place it represents (scene classification). Indoor environments present a unique set of challenges. At the same time, solving them separately seems inefficient. It is difficult to imagine serial algorithmic reproduction in robotics or Internet of Things devices, where resources are minimal, and demands on response speed are high. This implies the selection of advanced algorithms that work best in parallel. The current trends in image processing solutions are deep neural networks, thanks to their high efficiency. Moreover, these networks can be combined into architectures, enabling the previously mentioned tasks to be performed simultaneously. This approach is referred to in the literature as multi-task learning. It has several advantages. In this thesis, joint semantic segmentation and scene classification in indoor environments will be shown as beneficial. In addition, research will be conducted on current deep learning techniques in the context of multi-task learning. Finally, the results of all methods will be compared and evaluated on a large dataset of indoor scenes.

Keywords: deep neural networks, multi-task learning, semantic segmentation, scene classification

Spis treści

1. Wprowadzenie	7
1.1. Cel pracy	7
1.2. Motywacje	7
2. Wstęp teoretyczny	9
2.1. Definicje zadań	9
2.1.1. Klasyfikacja sceny	9
2.1.2. Segmentacja obrazu	10
2.2. Nadzorowane uczenie maszynowe	10
2.3. Głębokie uczenie i konwolucje	11
2.4. Rozwój klasyfikacji obrazów	11
2.5. Rozwój segmentacji semantycznej	12
2.6. DeepLabV3	13
2.7. Finetuning	14
2.8. Uczenie wielozadaniowe	14
3. Rozwiązanie	16
3.1. Przegląd rozwiązań	16
3.2. Rozwiązywanie problemu	19
3.2.1. Uczenie wielozadaniowe	21
3.2.2. Wyłącznie klasyfikacja	22
3.2.3. Wyłącznie segmentacja	22
3.2.4. Finetuning	22
3.2.5. Pośrednia klasyfikacja z segmentacji	22
3.2.6. Bezpośrednia klasyfikacja z segmentacji	23
3.3. Zbiór danych	23
3.3.1. Wybór zbioru danych	24
3.3.2. Analiza zbioru danych	24
4. Wyniki	26
4.1. Analiza miar jakości	26
4.2. Analiza czasowa	30
4.3. Analiza konkretnych przykładów	31
4.3.1. Segmentacja semantyczna	31
4.3.2. Klasyfikacja sceny	39
5. Podsumowanie	42
5.1. Wnioski	42
5.2. Podsumowanie	43
Bibliografia	45
Spis rysunków	48

0. Spis treści

Spis tabel	48
-------------------	-------	----

1. Wprowadzenie

1.1. Cel pracy

Celem pracy jest zbadanie problemu wspólnej segmentacji semantycznej i klasyfikacji sceny we wnętrzach. Segmentacja semantyczna polega na przypisaniu etykiety do każdego piksela obrazu, natomiast klasyfikacja sceny polega na rozpoznaniu typu sceny przedstawionej na obrazie. Oba zadania mają szerokie spektrum zastosowań, takich jak autonomiczna nawigacja czy robotyka manipulacyjna.

Środowiska wewnętrzne, takie jak domy i biura, stanowią unikalny zestaw wyzwań dla segmentacji semantycznej i klasyfikacji scen. Środowiska te są często nieuporządkowane i zawierają wiele różnych obiektów, co utrudnia dokładną segmentację i klasyfikację. Dodatkowo wnętrza mogą się znacznie różnić pod względem układu i wyglądu, co czyni trudnym opracowanie modelu, który może być uogólniony na różne typy scen wewnętrznych.

Głównym celem tej pracy jest opracowanie modelu opartego na głębokim uczeniu przy jednoczesnej semantycznej segmentacji i klasyfikacji sceny w różnych rodzajach pomieszczeń. Proponowany model zostanie wytrenowany i oceniony na dużym zbiorze danych scen wewnętrznych i zostanie porównany z aktualnymi metodami segmentacji semantycznej i klasyfikacji scen.

Aby osiągnąć ten cel, zostaną podjęte następujące pytania badawcze

- Jak można zaprojektować model oparty na głębokim uczeniu do wspólnej segmentacji semantycznej i klasyfikacji scen w środowiskach wewnętrznych?
- Czy przestrzeń reprezentacji po wytrenowaniu na zadaniu segmentacji semantycznej może być użyta do zadania klasyfikacji sceny?
- Jak dobrze proponowany model radzi sobie na dużym zbiorze danych scen wewnętrznych i jak wypada w porównaniu z aktualnymi metodami segmentacji semantycznej i klasyfikacji scen osobno?
- Jak proponowany model może być wykorzystany do poprawy wydajności w robotyce mobilnej?

Podsumowując, celem tej pracy jest opracowanie i ocena modelu opartego o głębokie uczenie dla wspólnej segmentacji semantycznej i klasyfikacji scen w środowiskach wewnętrznych oraz zbadanie potencjału modelu do poprawy jakości i wydajności na tychże zadaniach.

1.2. Motywacje

Wspólna segmentacja oraz klasyfikacja polega na oznaczaniu i kategoryzowaniu różnych regionów w obrębie wnętrz, oraz ich charakteryzowanie. Techniki te mogą być stosowane w różnych dziedzinach, w tym w robotyce, zarządzaniu budynkami i rozszerzonej rzeczywistości.

1. Wprowadzenie

W robotyce wspólna segmentacja semantyczna i klasyfikacja scen może być wykorzystana do umożliwienia robotom zrozumienia i nawigacji w środowiskach wewnętrznych. Może to obejmować identyfikację różnych obiektów i regionów w scenie, takich jak ściany, meble i ludzie, a także określenie ogólnego układu i funkcjonalności przestrzeni, np. czy jest to kuchnia, czy salon. Dzięki zrozumieniu środowiska w ten sposób roboty mogą poprawić swoją zdolność do wykonywania zadań, takich jak manipulacja obiektyami, nawigacja i interakcja człowiek-robot.

W zarządzaniu budynkiem wspólna segmentacja semantyczna i klasyfikacja sceny może być wykorzystana do poprawy funkcjonalności i wydajności budynków poprzez automatyczną identyfikację i etykietowanie różnych obiektów i regionów w budynku. Może to obejmować identyfikację różnych pomieszczeń, klatek schodowych i wind, jak również określenie ogólnego układu i funkcjonalności przestrzeni, np. czy jest to biuro, czy fabryka. Dzięki zrozumieniu środowiska w ten sposób, systemy zarządzania budynkiem mogą poprawić swoją zdolność do wykonywania zadań, takich jak zarządzanie energią, bezpieczeństwo i wykrywanie zajętości.

W dziedzinie rozszerzonej rzeczywistości wspólna segmentacja semantyczna i klasyfikacja sceny mogą być wykorzystane do poprawy realizmu doświadczeń AR poprzez zrozumienie środowiska rzeczywistego i rozszerzenie go o dodatkowe informacje lub obiekty wirtualne. Dzięki zrozumieniu środowiska w ten sposób, doświadczenia AR mogą być bardziej świadome kontekstowo, zapewniając w ten sposób bardziej realistyczne i angażujące doświadczenia.

Wspólna segmentacja semantyczna i klasyfikacja sceny w środowiskach wewnętrznych jest wymagającym, ale ważnym obszarem badawczym o wielu potencjalnych zastosowaniach. Wiąże się to z wykorzystaniem zaawansowanych technik widzenia komputerowego, solidnych i wydajnych algorytmów oraz starannej oceny w rzeczywistych środowiskach wewnętrznych. W miarę rozwoju technologii prawdopodobnie zostaną zidentyfikowane nowe przypadki użycia i zastosowania, i nadal będzie to aktywny obszar badań.

2. Wstęp teoretyczny

W tym rozdziale przedstawione zostaną najważniejsze koncepcje niezbędne do dalszej analizy pracy. Celem tego rozdziału jest przede wszystkim jednoznaczne sprecyzowanie, czym jest segmentacja semantyczna oraz klasyfikacja sceny w dziedzinie pomieszczeń. Zostanie udzielona odpowiedź na fundamentalne pytania, między innymi, czym jest uczenie maszynowe oraz dlaczego warto korzystać z głębokich sieci neuronowych. W dalszej części zostaną przedstawione aktualne sposoby realizacji celów pracy z przedstawieniem ich rozwoju na przestrzeni lat. Rozdział wieńczą opisy bardziej zaawansowanych technik realizacji wspomnianych algorytów.

2.1. Definicje zadań

2.1.1. Klasyfikacja sceny



Rysunek 2.1. Problem różnorodności wewnętrzklasowej oraz wieloznaczności semantycznej [1].

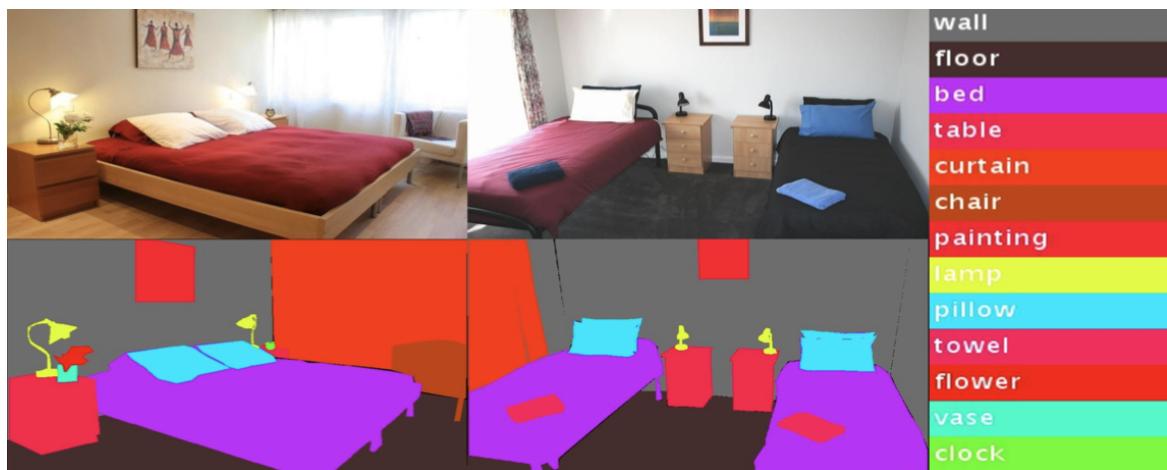
Zadanie klasyfikacji sceny polega na przyporządkowaniu kategorii miejsca, które przedstawia obraz. Istnieje duża różnica między klasyfikacją obrazu a klasyfikacją sceny w kontekście trundości. Klasyfikacja obrazu jako taka zajmuje się przyporządkowaniem klasy obiektu pierwszoplanowego, np. czy na obrazie znajduje się pies, czy kot. Klasyfikacja sceny natomiast musi wziąć pod uwagę wszystkie cechy obrazu, zarówno tła, jak i pierwszego planu, by określić odpowiednie miejsce.

W kontekście środowisk wewnętrznych klasyfikacja scen stanowi wyzwanie ze względu na zmienność scen wewnętrznych, obecność okluzji oraz fakt, że ten sam typ sceny może wyglądać inaczej na różnych obrazach. Wyróżniamy między innymi problem różnorodności wewnętrz klasowej oraz wieloznaczności semantycznej, co zostało przedstawione na

2. Wstęp teoretyczny

rys. 2.1. Pierwszy z nich polega na fakcie, iż jedno miejsce może zostać przedstawione w bardzo różnej konfiguracji m.in. oświetlenia, ekspozycji, obiektów znajdujących się na obrazie. Drugi jest związany z występowaniem tych samych obiektów dla różnych klas scen.

2.1.2. Segmentacja obrazu



Rysunek 2.2. Segmentacja wewnętrz pomieszczeń [2].

Zadanie segmentacji obrazu to przyporządkowanie każdemu pikselowi etykiety takiej jak „łóżko”, „kanapa” lub „umywalka”, do każdego piksela w obrazie (rys. 2.2). W rezultacie obraz zostaje podzielony na spójne regiony pod względem pewnych własności. Segmentacja może być reprezentowana jako tablica 2D, gdzie każdy element odpowiada pikselowi w obrazie wejściowym i ma wartość wskazującą jego etykietę klasy.

Zadanie segmentacji można rozszerzyć do zadania segmentacji instancji (ang. instance segmentation), czyli segmentacji klasycznej rozszerzonej o rozróżnienie poszczególnych obiektów w ramach tej samej klasy. W przypadku klasycznej wersji nie jesteśmy w stanie rozróżnić dwóch stojących obok siebie łóżek, gdyż mapa segmentacji jest dla nich jednakoła. Segmentacja instancji pozwala natomiast takie rozróżnienie uczynić. Segmentacja semantyczna w dalszej części pracy będzie odnosić się do klasycznej wersji. Segmentacja instancji nie jest tematem pracy.

2.2. Nadzorowane uczenie maszynowe

Uczenie maszynowe jest częścią sztucznej inteligencji, które umożliwia przeprowadzanie wnioskowania z danych. Algorytmy te rozpoznają wzorce i dokonują przewidywań.

Uczenie nadzorowane to rodzaj uczenia maszynowego, w którym algorytm jest szkoleny na etykietowanym zestawie danych, gdzie pożądane wyjście dla danego wejścia jest już znane. W kontekście głębokiego uczenia się, algorytmy uczenia nadzorowanego wykorzystują sieci neuronowe do uczenia się z danych i dokonywania przewidywań.

Jedną z głównych zalet wykorzystania głębokiego uczenia do uczenia nadzorowanego jest możliwość uczenia się złożonych i nieliniowych zależności wynikających z danych. Głębokie sieci neuronowe, z ich wieloma warstwami, mogą uczyć się i reprezentować wielowymiarowe i abstrakcyjne cechy danych, co pozwala im osiągnąć satysfakcjonujące rezultaty w wielu zadaniach. Co więcej, algorytmy głębokiego uczenia mogą obsługiwać duże ilości danych i mogą być łatwo zrównoleglone, co pozwala na skrócenie czasu treningu.

Istnieją jednak również ograniczenia w stosowaniu głębokiego uczenia do uczenia nadzorowanego. Jednym z ograniczeń jest konieczność posiadania dużej ilości oznaczonych danych. Aby wytrenować głęboką sieć neuronową, wymagana jest ich znaczna ilość. Dane nie zawsze mogą być łatwo dostępne lub łatwe do uzyskania. Co więcej, algorytmy głębokiego uczenia są często podatne na niskie obciążenie lub wysoką wariancję, zwłaszcza gdy ilość i jakość danych jest ograniczona. Może to prowadzić do słabej generalizacji.

2.3. Głębokie uczenie i konwolucje

Uczenie głębokie odnosi się do uczenia maszynowego, które charakteryzuje się wykorzystaniem głębokich sieci neuronowych. Składają się one z wielu warstw sztucznych neuronów. W kontekście wizji komputerowej głębokie uczenie jest wykorzystywane do skutecznego rozwiązywania wielu zadań, w tym klasyfikacji obrazów, wykrywania obiektów czy segmentacji semantycznej.

Jedną z kluczowych zalet głębokiego uczenia w wizji komputerowej jest zdolność do automatycznego uczenia się hierarchicznych reprezentacji obrazów. Wykorzystuje je się do wyodrębnienia wysokopoziomowych cech, które są wysoce zróżnicowane dla danego zadania. Stanowi to kontrast do tradycyjnych metod widzenia komputerowego, które zazwyczaj opierają się na ręcznie opracowanych cechach.

2.4. Rozwój klasyfikacji obrazów

Jedną z najwcześniejniejszych i najbardziej wpływowych prac w dziedzinie głębokich spłotowych sieci neuronowych (CNN) jest „ImageNet Classification with Deep Convolutional Neural Networks” autorstwa Alexa Krizhevsky’ego et al. (2012)[3]. W pracy tej przedstawiono zastosowanie głębokich sieci neuronowych do klasyfikacji obrazów i osiągnięto najwyższe wyniki na zbiorze danych ImageNet. Praca ta wyznaczyła nowy punkt odniesienia dla klasyfikacji obrazów i zapoczątkowała szerokie zastosowanie CNN w zadaniach widzenia komputerowego.

W kolejnych latach wielu badaczy zaproponowało różne modyfikacje i ulepszenia podstawowej architektury CNN. Jednym z ważnych składów jest architektura Inception, wprowadzona przez Szegedy et al. w „Going Deeper with Convolutions” (2015)[4]. Architektura Inception wykorzystuje kombinację różnych rozmiarów filtrów konwolucyjnych

2. Wstęp teoretyczny

do ekstrakcji cech w wielu skalach, co pozwala sieci uczyć się bardziej złożonych i abstrakcyjnych cech niż wcześniejsze architektury.

Kolejną ważną innowacją było wykorzystanie połączeń rezydualnych, które zostało zaproponowane przez He et al. w „Deep Residual Learning for Image Recognition” (2016)[5]. Połączenia rezydualne pozwalają na trenowanie znacznie głębszych sieci, zapobiegając problemowi zanikających gradientów. Tak jak przedtem ImageNet posłużył do wykazania zalet tego rozwiązania.

Podsumowując, głębokie CNN są wysoce efektywne w zadaniach widzenia komputerowego, takich jak klasyfikacja obrazów. Rozwój głębokich CNN zaznaczył się kilkoma ważnymi kamieniami milowymi, takimi jak stosowanie różnych filtrów splotowych oraz wykorzystaniem połączeń rezydualnych. Te innowacje doprowadziły do znacznej poprawy wydajności na zbiorze danych ImageNet i zainspirowały dalsze badania w innych zadaniach widzenia komputerowego.

2.5. Rozwój segmentacji semantycznej

Jednym z najwcześniejzych i najbardziej wpływowych artykułów w dziedzinie głębokich CNN do segmentacji semantycznej jest „Fully Convolutional Networks for Semantic Segmentation” autorstwa Longa, Shelhamera i Darrella (2015)[6]. W pracy tej, zaprezentowanej na konferencji Computer Vision and Pattern Recognition (CVPR), przedstawiono architekturę sieci w pełni splotową (FCN) do segmentacji semantycznej. Architektura FCN wykorzystuje serię warstw splotowych i upsamplingu do produkcji gęstych predykcji per-piksel. Praca ta pokazała, że CNN mogą być wykorzystane do predykcji na poziomie pikseli i stworzyła podstawy dla wielu późniejszych podejść do segmentacji semantycznej.

Innym kluczowym wkładem w dziedzinie segmentacji semantycznej jest „U-Net: Convolutional Networks for Biomedical Image Segmentation” autorstwa Ronneberger, Fischer i Brox (2015)[7]. W pracy tej, zaprezentowanej na międzynarodowej konferencji Medical Image Computing and Computer-Assisted Intervention (MICCAI), przedstawiono architekturę U-Net do segmentacji obrazów biomedycznych. Architektura U-Net wykorzystuje kombinację warstw splotowych i poolingowych do ekstrakcji cech w wielu skalach oraz serię warstw upsamplingu do produkcji gęstych predykcji per-piksel. Praca ta pokazała, że architektura U-Net dzięki zastosowaniu połączeń pomijających (skipping connections) jest w stanie znacznie lepiej rekonstruować obraz. Szczególnie dotyczy to elementów małej skali, które wcześniej były pomijane przez FCN. Praca ta została szeroko wykorzystana w obrazowaniu medycznym i nie tylko.

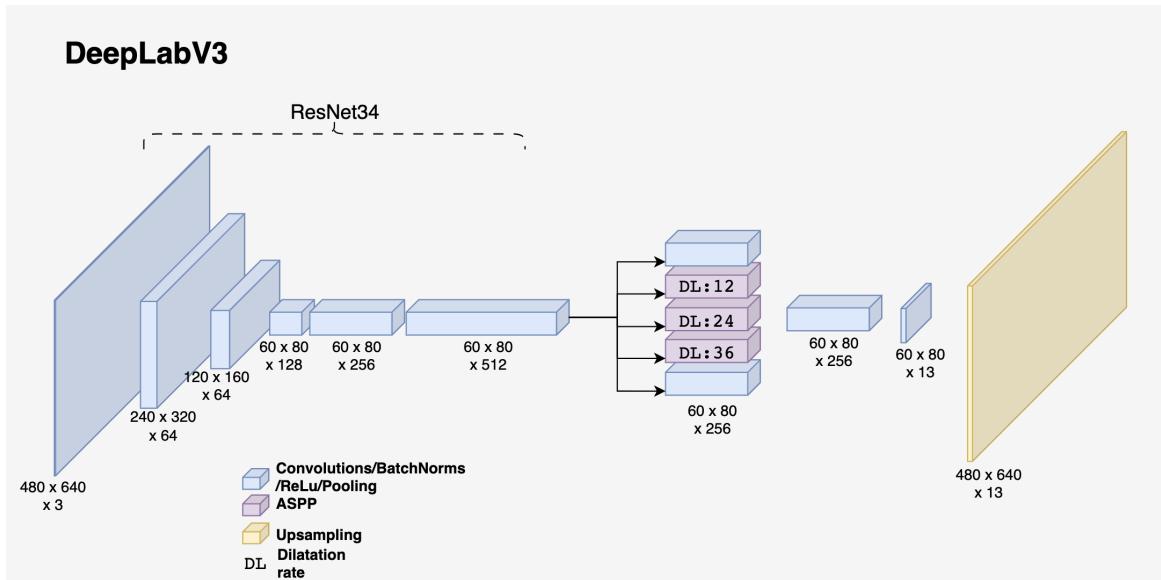
Kolejną ważną pracą w dziedzinie segmentacji semantycznej jest „DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs” autorstwa Chen, Papandreou, Kokkinos, Murphy i Yuille (2016)[8]. W pracy tej, zaprezentowanej na International Conference on Computer Vision (ICCV), przedstawiono architekturę DeepLab do segmentacji semantycznej. Architektura De-

epLab wykorzystuje rozszerzony splot (atrous convolution) do zwiększenia pola widzenia warstw splotowych oraz warunkowe pola losowe (CRF) do dopracowania predykcji. Praca ta pokazała, że użycie rozszerzonego splotu i CRF może poprawić efekty segmentacji semantycznej.

Podsumowując, segmentacja semantyczna jest zadaniem o dużym znaczeniu w wizji komputerowej, a głębokie CNN okazały się wysoce skuteczne w rozwiązywaniu tego zadania. Rozwój głębokich CNN do segmentacji semantycznej został oznaczony przez kilka ważnych kamieni milowych, w tym wprowadzenie FCN przez Long et al., U-Net przez Ronneberger et al. i DeepLab przez Chen et al. Te architektury wyznaczyły nowe standardy w segmentacji semantycznej i zostały szeroko przyjęte w różnych dziedzinach zastosowań.

2.6. DeepLabV3

Literatura uważa go za model lepszy od sieci U-Net czy FCN. Model DeepLabV3 (rys. 2.3) nie korzysta z połączeń pomijających. Informacje o kontekście w wielu skalach uzyskuje przez moduł Spatial Pyramid Pooling (SPP). Wykorzystuje on bloki Atrous Spatial Pyramid Pooling (ASPP) oraz klasyczny pooling. Bloki ASPP składają się ze splotu, normalizacji pakietowej oraz funkcji aktywacji ReLU. Sploty przyjmują różną postać. Pierwszy blok to splot o jądrze 1x1. Następne bloki korzystają z rozszerzonego splotu o dylatacji oraz wypełnieniu (padding) równemu współczynnikowi rozszerzenia (dilatation rate). Dla kolejnych 3 bloków wynosi on 12, 24, 36. Ostatni blok SPP to zwykły pooling. Bloki składające się na moduł SPP są następnie dodawane wzdużnie i poddawane splotowi. Następnie egzekwuje się splot o wyjściowej liczbie kanałów równej ilości klas. Końcowy etap obejmuje upsampling do pożądanego wymiaru.



Rysunek 2.3. Klasyczna architektura DeepLabV3 z backbonem ResNet34.

2.7. Finetuning

Finetuning jest metodą uczenia głębokich sieci neuronowych. Polega on na odtworzeniu wag modelu, wcześniej wytrenowanego na dużym zbiorze danych jak ImageNet, a następnie próbie dostosowania go do obecnie rozważanego problemu. W kontekście wizji komputerowej pierwsze warstwy modelu są najczęściej ogólne i odnoszą się do ogólnych cech obrazu. Ta wiedza pozwala wnioskować, że pierwsza część modelu nie zależy w głównej mierze od zbioru danych oraz rozważanego zadania, tylko jest czymś ogólnym dla wielu problemów przetwarzania obrazu. Zatem pojawia się możliwość ponownego użycia części gotowego modelu. W takim przypadku mowa o transferze wiedzy. Technicznie finetuning najczęściej rozpoczyna się od uczenia modelu, wykorzystując jedynie ostatnie warstwy. W miarę kolejnych epizodów uczenia wykorzystuje się coraz więcej warstw sieci. Taki zabieg nazwy się odmrażaniem kolejnych warstw sieci.

2.8. Uczenie wielozadaniowe

Uczenie wielozadaniowe jest techniką uczenia maszynowego, w której model jest trenowany do wykonywania wielu zadań jednocześnie. Zabieg ten stosuje się w celu nauczenia się wspólnych reprezentacji, które mogą poprawić skuteczność we wszystkich zadaniach. To podejście zyskało uwagę w ostatnich latach ze względu na rosnące zapotrzebowanie na modele, które mogą wykonywać wiele zadań z wysoką dokładnością i wydajnością. Uczenie wielozadaniowe ma szereg zastosowań, takich jak widzenie komputerowe, przetwarzanie języka naturalnego i rozpoznawanie mowy.

Sebastian Ruder w swoim przeglądzie literatury „An Overview of Multi-Task Learning in Deep Neural Networks” (2017) [9] dość zwięźle definiuje uczenie wielozadaniowe jako optymalizację co najmniej dwóch funkcji straty. Co więcej, pokazuje, że takie podejście ma swoje silne biologiczne analogie. Autor dopatruje się tutaj odpowiedzi na pytanie, czym jest uczenie się uczenia (learning to learn), a więc główna przesłanka bardzo silnego nurtu meta-learningu. Podkreśla, że uczenie wielozadaniowe pomaga osiągać lepsze rezultaty niż klasyczne uczenie jednego zadania. Zachęca nawet do stosowania uczenia wielozadaniowego w przypadku, gdy potrzebujemy zaledwie jednego zadania poprzez znalezienie zadania lub zadań komplementarnych. Autor wielokrotnie odwołuje się do dzieła „Multitask learning: A knowledge-based source of inductive bias” (1993) [10] przypominając, że uczenie wielozadaniowe przyczynia się do lepszej generalizacji modelu, a więc uniezależnienie się od domeny uczącej na rzecz szeroko pojętej wiedzy.

Ruder opisuje dwa główne podejścia do uczenia wielozadaniowego — twardy oraz miękkie dzielenie wag sieci (soft/hard parameter sharing). Twardy dzielenie wag jest najczęściej stosowane. Polega na uwspólnieniu pierwszej części sieci, odpowiedzialnej za zdefiniowanie przestrzeni reprezentacji (ang. backbone) oraz rozdzieleniu kolejnych warstw związanych z konkretnym zadaniem. Miękkie dzielenie wag polega na zbudowaniu

wielu sieci, odpowiednich dla danego zadania. Co więcej, sieci te podczas uczenia są regularyzowane w ten sposób, aby zachęcić je do posiadania jak najbardziej skoncentrowanych wag.

Takie podejście może się powieść jedynie w przypadku, kiedy dwa zadania są powiązane ze sobą. Powstało wiele prac poświęconych odpowiedzi na pytanie, które zadania warto wybrać, a które należy rozpatrywać osobno. Jednym z takich dzieł jest praca zespołu ze Stanfordu „Which Tasks Should Be Learned Together in Multi-task Learning?” Standley et al. (2020) [11]. Przedstawia ona pojęcie negatywnego wpływu (ang. negative transfer), który najprościej rzecz ujmując sprawia, że sieć uczy się gorzej niż pojedyncze sieci. Autorzy zbadali, że największy wpływ na jakość uczenia wielozadaniowego ma właśnie odpowiedni dobór zadań, a niekoniecznie rozmiar zbioru danych czy wielkość modelu. Oczywiście należy zwrócić uwagę, że przytoczone czynniki nie są bez znaczenia, jedynie w porównaniu z doborem zadań mają pomijalne znaczenie. Co ciekawe zadania afinczne względem siebie mogą mieć dodatni wpływ w przypadku transferu wiedzy, a nie muszą być afinczne w kontekście uczenia wielozadaniowego.

Gdy jednak zadania są pokrewne względem siebie, jesteśmy w stanie zaobserwować konkretne korzyści związane ze wspólnym uczeniem. Ruder wymienia kilka najważniejszych. Po pierwsze zyskujemy tak zwaną niejawną augmentację danych (ang. implicit data augmentation). Każde z zadań posiada pewien szum związany z konkretnym zadaniem. Uczenie wielu zadań pozwala w pewnym stopniu wyeliminować szum związany z konkretnym zadaniem na rzecz lepszej generalizacji. Kolejną zaletą jest lepsze skupienie uwagi na ważnych informacjach. Ma to szczególne znaczenie w przypadku gdy dane są ograniczone lub wielowymiarowe. Uczenie wielozadaniowe może pomóc w wyborze tych najbardziej znaczących cech. Co więcej, wspólna wiedza zdobyta podczas uczenia może okazać się znacząca. Niektóre cechy są łatwiejsze do wykrycia dla jednego zadania, inne dla drugiego. Łącząc te informacje przez tak zwane „podsłuchiwanie” (ang. eavesdropping) model jest w stanie zbudować lepszą przestrzeń reprezentacji. Oprócz zyskania na jakości modelu przypadek twardego dzielenia wag pozwala znaczco ograniczyć wielkość modelu. Nie trzeba bowiem stosować wielu backbone’ów, które stanowią największą część modelu w kontekście liczby parametrów. Implikuje to znacznie zmniejszenie czasu uczenia oraz wnioskowania [11].

3. Rozwiązanie

W tym rozdziale przedstawione zostaną wybrane metody, które zostały sprawdzone w ramach analizy problemu. Rozważania zostaną przedstawione w ścisłym związku z pytaniami badawczymi przedstawionymi w celu pracy, a więc:

- Jak można zaprojektować model oparty na głębokim uczeniu do wspólnej segmentacji semantycznej i klasyfikacji scen w środowiskach wewnętrznych?
- Czy przestrzeń reprezentacji po wytrenowaniu na zadaniu segmentacji semantycznej może być użyta do zadania klasyfikacji sceny?
- Jak dobrze proponowany model radzi sobie na dużym zbiorze danych scen wewnętrznych i jak wypada w porównaniu z aktualnymi metodami segmentacji semantycznej i klasyfikacji scen osobno?
- Jak proponowany model może być wykorzystany do poprawy wydajności w robotyce mobilnej?

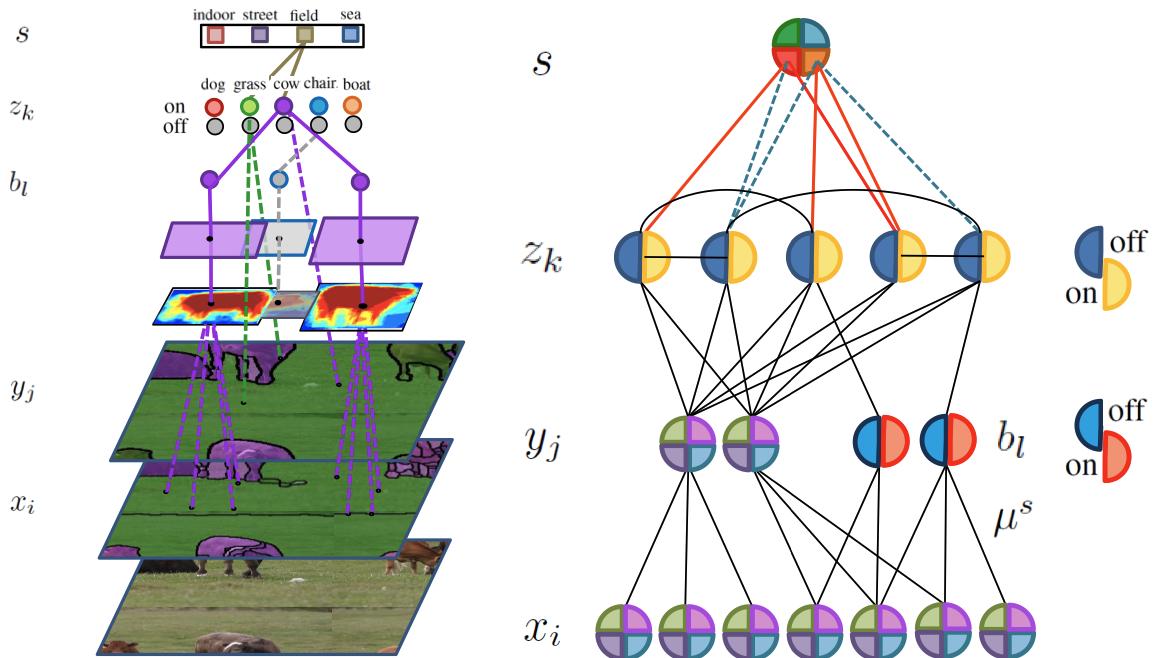
Opis rozwiązań problemu zostanie poprzedzony przeglądem rozwiązań. Analiza dotychczasowego stanu wiedzy pozwoli lepiej ukierunkować badania. Intuicja oraz wskazówki zdobyte podczas przeglądu zostaną uwzględnione w doborze metod i eksperymentów.

3.1. Przegląd rozwiązań

Przegląd literatury jest kluczowym aspektem każdej pracy naukowej. W tym rozdziale zostaną przedstawione wyłącznie rozwiązania obejmujące łączną segmentację semantyczną oraz klasyfikację sceny. Szczególny nacisk położony zostanie na architektury głębokich, wielozadaniowych sieci neuronowych. Niestety przyjęte założenia w pracy nie zostały opisane przez nikogo wcześniej, zgodnie z najlepszą wiedzą autora. Niektóre prace naukowe przedstawiają ten sam problem, to jest klasyfikacji i segmentacji łącznie, ale obejmują go w innej domenie danych. Z drugiej strony artykuły obejmujące środowiska wewnętrz są dobrze zdefiniowane, jednak często w swoich rozwiązańach autorzy korzystają z obrazu głębi, który nie zawiera się w zakresie badań tej pracy. Nie mniej wszystkie poniższe artykuły stanowią cenne źródło informacji oraz wskazówek.

Pierwszym z prezentowanych artykułów jest „Describing the Scene as a Whole: Joint Object Detection, Scene Classification and Semantic Segmentation” autorstwa Yao J. et al. (2012)[12]. Prezentuje on algorytm, który ówcześnie wyznaczył najlepsze podejście (ang. state-of-the-art (SOTA)). Rozwiązanie opiera się na warunkowych polach losowych, które ówcześnie były szeroko stosowane. Mimo że nie jest to rozwiązanie oparte o głębokie sieci neuronowe, to autorzy wskazują tutaj ważne zagadnienia. Po pierwsze udowadniają, że połączenie segmentacji i klasyfikacji okazało się owocne nie tylko pod względem jakości, ale również wydajności w kontekście czasowym. W swojej pracy wykorzystują podejście równolegle zgodne z rysunkiem 3.1. Podsumowując, „Describing the Scene as a Whole:

Joint Object Detection, Scene Classification and Semantic Segmentation” nie jest propozycją architektury głębokiej sieci. Wskazuje on na problemy z łączeniem zadań szeregowo, jednocześnie udowadniając, że taka praktyka był ówcześnie stosowana, więc nie można uznawać stosowania połączenia szeregowego jako niedopuszczalnego. Poza tym autorzy doceniają wspólne realizowanie zadań, oceniacąc je jako bardziej efektywne czasowo i obliczeniowo.

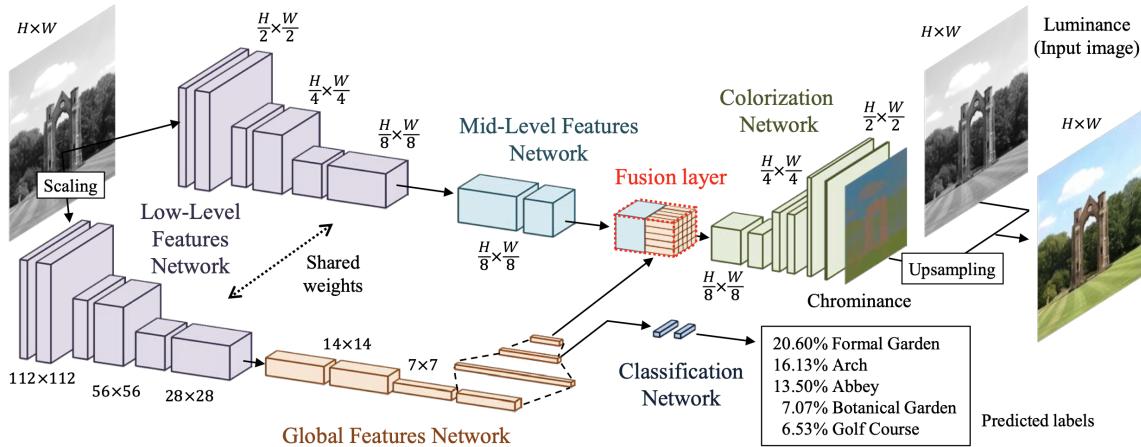


Rysunek 3.1. Describing the Scene as a Whole: Joint Object Detection, Scene Classification and Semantic Segmentation (2012) [12].

Artykuł „Let there be Color!: Joint End-to-end Learning of Global and Local Image Priors for Automatic Image Colorization with Simultaneous Classification” (2016) [13] przedstawia rozwiązanie problemu jednoczesnego klasyfikowania sceny oraz kolorowania zdjęć. Do realizacji zadania kolorowania potrzebna jest semantyczna maska. Wynika z tego, że kolorowanie jest rozszerzeniem segmentacji semantycznej. Rozumiejąc towarzyszące analogie, można przejść do analizy rozwiązania. Przedstawiona architektura (rys.3.2) jest przykładem sieci wielozadaniowej, używającej miękkiego dzielenia parametrów, ale tylko i wyłącznie w obrębie pierwszej części sieci. Szczególnie ciekawa jest konkatenacja cech wysokiego poziomu (Global Features Network) z cechami średniopoziomowymi (Mid-Level Features Network), która ma miejsce w warstwie fuzji (Fusion layer). Iizuka et al. formułują wniosek oznajmiający o kluczowym znaczeniu tej warstwy w kontekście całego zadania. Wiedza o scenie zdjęcia może dostarczyć informacji wpływających na decyzję, czy na obrazie znajduje się niebo, czy trawa. Rozważając sceny wewnętrz, oczywiste jest, że nie będzie tam takich grup semantycznych, jednak bezpośrednia informacja o

3. Rozwiązanie

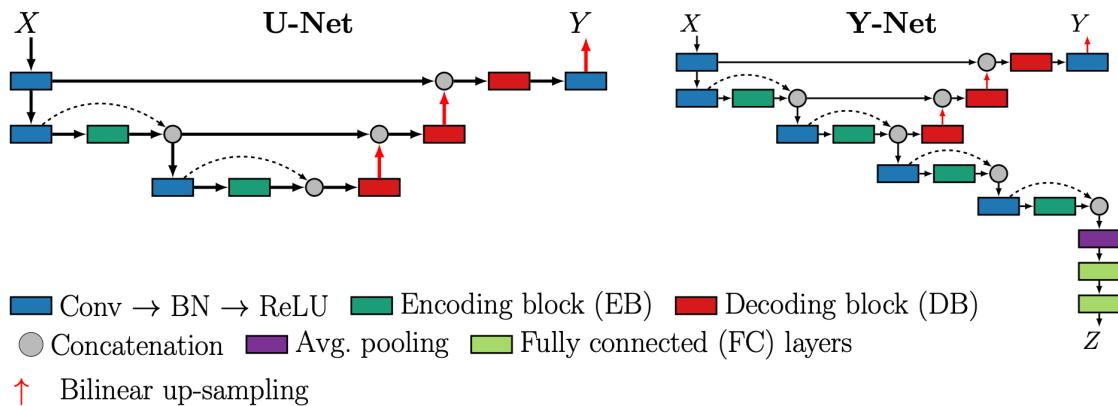
miejscu sceny, np. łazienka, może pomóc w ustaleniu etykiet segmentacji. Podsumowując, cechy nauczone na zadaniach klasyfikacji i segmentacji, mogą wzajemnie pozytywnie na siebie wpływać, realizując pozytywny transfer.



Rysunek 3.2. Let there be Color!: Joint End-to-end Learning of Global and Local Image Priors for Automatic Image Colorization with Simultaneous Classification 2016 [13].

Zastosowanie łącznej segmentacji oraz klasyfikacji tym razem w domenie obrazowania medycznego przedstawia „Y-Net: Joint Segmentation and Classification for Diagnosis of Breast Biopsy Images” (2018) [14]. Zadania te są realizowane przez twarde dzielenie parametrów w kontekście uczenia wielozadaniowego (rys. 3.3). Architektura jest prostym rozszerzeniem klasycznego U-Netu. Autorzy wskazują, że taki zabieg powoduje dużą modularność, ponieważ do dowolnego modelu segmentacji można podłączyć sieć klasycyjną. Przeprowadzone eksperymenty dla segmentacji udowodniły, że dokładność pozostała na tym samym poziomie. W przypadku klasyfikacji wyniki były wyższe niż dotychczasowe SOTA na tym zbiorze. Jako funkcję straty autorzy użyli sumy entropii skrośnej każdego z zadań. Podsumowując, zadanie segmentacji osiągnęło ten sam wysoki wynik co SOTA, a zadanie klasyfikacji ustanowiło nowe SOTA na tym zbiorze, wykorzystując znacznie mniej parametrów.

Najbliższym artykułem tej pracy inżynierskiej jest „Efficient Multi-Task RGB-D Scene Analysis for Indoor Environments” (2022) [15], który został opublikowany w czasie tworzenia tej pracy. Przedstawia on jedną głęboką sieć neuronową rozwiązującą następujące zadania: segmentacja semantyczna oraz segmentacja instancji (łącznie ang. pantopic segmentation), estymację orientacji instancji oraz klasyfikację sceny. Rozważaną przez autorów domeną są podobnie jak w przypadku tej pracy sceny wnętrz. Znaczną różnicą poza dodatkowymi zadaniami jest użycie przez Seichter et al. informacji o głębi. Zgodnie z wnioskami z niniejszego artykułu przetwarzanie łączne obrazów RGB i głębi jest kluczowe z punktu widzenia jakości predykcji, więc nie można bezpośrednio po-

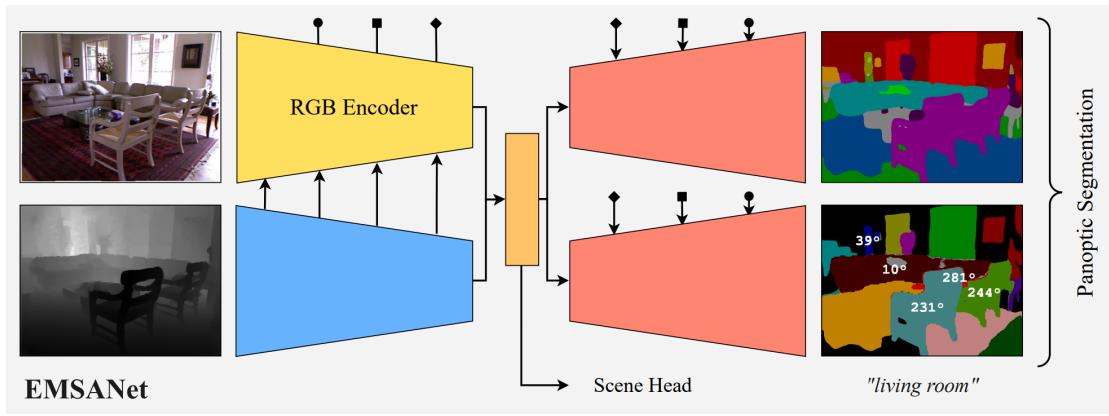


Rysunek 3.3. Y-Net: Joint Segmentation and Classification for Diagnosis of Breast Biopsy Images 2018 [14].

równać go z niniejszą pracą. Autorzy wykonali wiele eksperymentów, badając różne metodologie. Architektura jest przedstawiona na rysunku 3.4. Autorzy zdecydowali się na twardy dzielenie parametrów, argumentując całkowitą niezależnością w przypadku chęci wyłączenia jednego zadań z wnioskowania. Pierwszym krokiem, który wykonali, było ustalenie punktu odniesienia poprzez trenowanie osobno każdego z zadań. Trening każdej sieci z osobna był rozważany pod względem wielu backbone'ów ze zróżnicowaniem na uczenie wyłącznie obrazu głębi, obrazu RGB lub RGB-D. Z reguły w przypadku segmentacji oraz klasyfikacji większy backbone wpływał na polepszenie wyników. Trenując zadania łącznie, zdecydowano się na ważoną sumę entropii skrośnej dla zadania segmentacji i klasyfikacji w proporcjach odpowiednio 3:1. Przyjęty krok uczenia, będąc sprawdzonym przez przeszukiwanie liniowe (ang. grid search), jest wyjątkowo duży, bo wynosi 0.02. Autorzy zastosowali zaawansowane techniki dostosowywania kroku uczenia w trakcie treningu poprzez użycie planisty polityki jednego cyklu (ang. one cycle policy scheduler). Jako optymalizator użyto SGD z momentem oraz drobną regularyzacją. Podsumowując, zgodnie z prezentowanymi wynikami na wspólnej segmentacji oraz klasyfikacji autorom nie udało się polepszyć działania modelu na segmentacji semantycznej. Z powodzeniem jednak wzrosła dokładność klasyfikacji na zbiorze NYUv2.

3.2. Rozwiązanie problemu

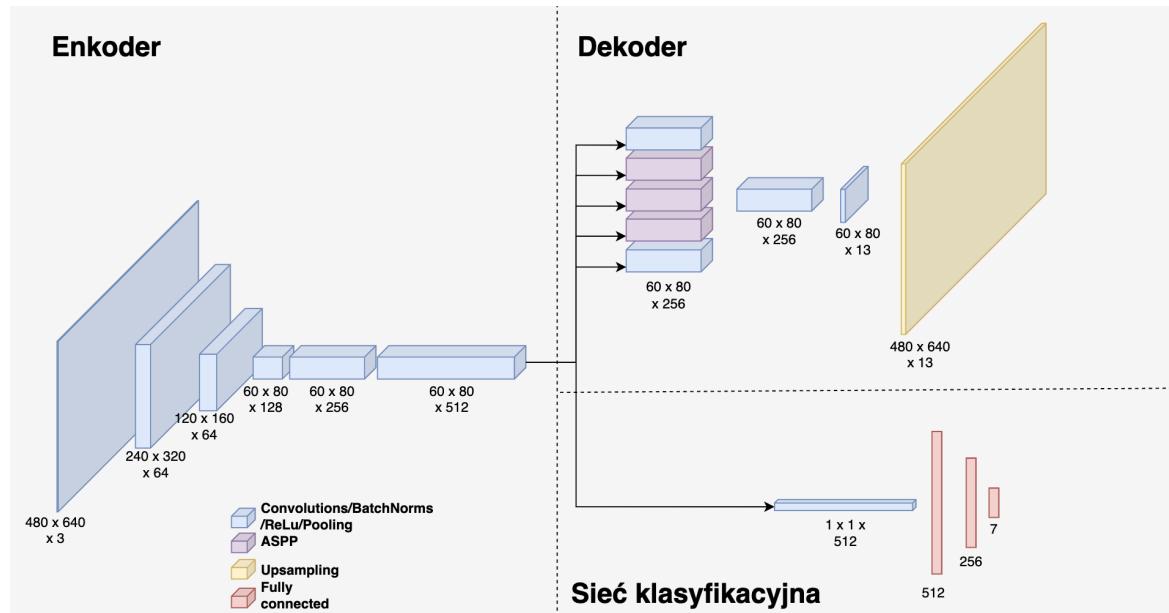
W tym podrozdziale zostaną przedstawione eksperymenty, które wykonano w celu zbadania uczenia wielozadaniowego segmentacji semantycznej oraz klasyfikacji sceny w domenie pomieszczeń. Pierwszym etapem, jakiego dokonano, było wyznaczenie punktu odniesienia. Z punktu widzenia pracy najłatwiej byłoby znaleźć gotowe wyniki segmentacji oraz klasyfikacji sceny na wybranym zbiorze danych. Niestety żadne z przytaczanych rozwiązań nie odpowiada w pełni zakresowi pracy. Postanowiono stworzyć taki punkt od-



Rysunek 3.4. Efficient Multi-Task RGB-D Scene Analysis for Indoor Environments [15]

niesienia samemu przez analogiczne trenowanie sieci segmentacyjnej oraz klasyfikacyjnej osobno.

Mając taką wiedzę, eksperymentowano dalej z różnymi architekturami uczenia wielozadaniowego. Wybrano uczenie łączne o twardym dzieleniu parametrów. Podejście to ma wiele zalet. [14] podkreśla łatwość i wszechstronność implementacji. Wystarczy dołączyć do modelu część klasyfikacyjną. Co więcej, wszyscy autorzy ([14], [15]) chwalą znacznie mniejszą ilość parametrów sieci, co bezpośrednio wpływa na czas treningu oraz wnioskowania. Architektura sieci przedstawia rys. 3.5. Jest to DeepLabv3 rozszerzony za enkoderem o sieć klasyfikacyjną podobnie jak w artykule [14], gdzie rozszerzono sieć U-Net.



Rysunek 3.5. Architektura wielozadaniowej sieci.

Normalizacja jest ważnym krokiem przetwarzania wstępniego w problemach widzenia komputerowego. W pracy „Normalization Techniques in Training DNNs: Methodology,

Analysis and Application" Lei et. al. [16], autorzy udowadniają, że normalizacja stabilizuje i przyśpiesza trening oraz prawdopodobnie prowadzi do poprawy generalizacji. Jako przetwarzanie wstępne obrazu zastosowano normalizację gaussowską. Obrazy RGB zostały poddane normalizacji ze średnią (0.485, 0.456, 0.406) oraz odchyleniem standardowym (0.229, 0.224, 0.225), które odpowiadają parametrom rozkładu normalnego na zbiorze ImageNet. Gotowe wagi uzyskane poprzez uczenie na bazie ImageNet służyły jako wagi początkowe enkodera.

Znalezienie optymalnego zestawu hiperparametrów nie jest proste. Niewłaściwy dobór grozi brakiem osiągnięcia pożądanych rezultatów. W celu pozyskania optymalnego zestawu skorzystano z narzędzia Optuna [17]. Wykorzystano do tego algorytm TPE (Tree-structured Parzen Estimator), który jest znacznie korzystniejszy niż klasyczne przeszukiwanie siatką (Grid Search). Optymalizacja hiperparametrów nie tylko poprawia łatwość doboru hiperparametrów, ale przede wszystkim podwyższa wiarygodność rezultatów. Hiperparametry były optymalizowane względem straty na zbiorze walidacyjnym. Do optymalizowanych parametrów zalicza się tylko krok uczenia, chyba że stwierdzono w dalszej części rozdziału inaczej.

Do klasycznych funkcji straty dla segmentacji semantycznej zaliczamy entropię skrośną, ale również coraz popularniejsze Lovász Softmax [18] czy Focal Loss [19]. Entropia i Focal jest stratą związaną z dystrybucją pikseli, Lovász Softmax skupia się bardziej na konkretnych regionach [19]. W przypadku zadania klasyfikacji najczęściej spotykana jest entropia skrośna. W pracy wykorzystano entropię skrośną zarówno dla klasyfikacji, jak i dla segmentacji semantycznej podobnie jak [14] oraz [15]. Entropia była ważona poprzez odwrotność sumy odpowiednio pikseli dla danej etykiety semantycznej oraz etykiet związanej ze scenami.

Samo uczenie nie było długie, bo trwało od 5 do 15 epok. Zastosowano wczesne przerwanie treningu (Early Stopping), monitorując strategię na zbiorze walidacyjnym, by uniknąć przeuczenia. Poza tym zastosowano zmiennej krok uczenia poprzez planistę typu eksponentjalnego (exponential learning rate policy) o współczynniku γ równemu 0.99, który zmniejsza krok uczenia o γ co epokę.

W dalszej części przedstawione zostaną konkretne eksperymenty, które rozważano w pracy.

3.2.1. Uczenie wielozadaniowe

Uczenie wielozadaniowe zostało zrealizowane przez architekturę z rysunku 3.5. Trening polegał na aktualizowaniu wag całego dostępnego modelu zgodnie z propagacją wsteczną agregowanej funkcji straty λ . Zaimportowano ją jako sumę funkcji strat na każdym z zadań tak jak w przypadku [14]. Nie stosowano ważenia zadań wspomnianego w [15]. Ważenia zadań nie należy mylić z ważeniem etykiet w funkcji straty

$$\lambda = \lambda_{segmentacja} + \lambda_{klasyfikacja}$$

3.2.2. Wyłącznie klasyfikacja

W celu określenia punktu odniesienia wytrenowano model, zapominając o podsieci do wyznaczania segmentacji semantycznej. Technicznie skorzystano z modelu wielozadaniowego. Parametry modułów architektury takie jak dekoder zostały zamrożone, oraz nie zostały podawane optymalizatorowi w trakcie treningu. Funkcja straty λ została ograniczona wyłącznie do straty na klasyfikacji poprzez wyzerowanie w każdym kroku straty na segmentacji.

$$\begin{aligned}\lambda &= \lambda_{segmentacja} + \lambda_{klasyfikacja} \\ \lambda_{segmentacja} &= 0\end{aligned}$$

3.2.3. Wyłącznie segmentacja

Analogicznie jak w przypadku klasyfikacji należało określić punkt odniesienia również w przypadku segmentacji. Procedura była taka sama jak w przypadku klasyfikacji. Model wielozadaniowy zamrożono w części klasyfikacyjnej oraz wyłączono zamrożone parametry z optymalizacji. Funkcja straty λ została przedstawiona jako

$$\begin{aligned}\lambda &= \lambda_{segmentacja} + \lambda_{klasyfikacja} \\ \lambda_{klasyfikacja} &= 0\end{aligned}$$

3.2.4. Finetuning

Znaną techniką transferu wiedzy jest finetuning. W tym przypadku skorzystano z wytrenowanego enkodera ResNet wytrenowanego na dużej bazie ImageNet. Uczenie przebiegało w dwóch fazach. W pierwszej zamrożono enkoder i starano się osiągnąć jak najlepsze rezultaty, dysponując podsiecią klasyfikacyjną i segmentacyjną. Wynika z tego, że pierwszy etap to nic innego niż uczenie wielozadaniowe, ale z wyłączonym enkoderem. Dopiero w drugim etapie odmrażany jest również enkoder. Sytuacja wtedy przypomina wcześniej omawiane uczenie wielozadaniowe. Jednakże kluczowy jest dobór hiperparametrów. W pierwszym etapie uczenie przebiega z pewnym krokiem, który w drugim jest już znacznie mniejszy.

$$\lambda = \lambda_{segmentacja} + \lambda_{klasyfikacja}$$

3.2.5. Pośrednia klasyfikacja z segmentacją

Podejście transferu wiedzy można lekko zmodyfikować. Skorzystano z wcześniejszych przygotowanych wag będących wynikiem wcześniej wspomnianej wyłącznej segmentacji. Zamrożono enkoder oraz podsieć segmentacyjną oraz wyłączono te parametry z optymalizacji. Następnie dysponując samą podsiecią klasyfikacyjną przeprowadzono trening.

Funkcja straty była następująca:

$$\lambda = \lambda_{segmentacja} + \lambda_{klasyfikacja}$$

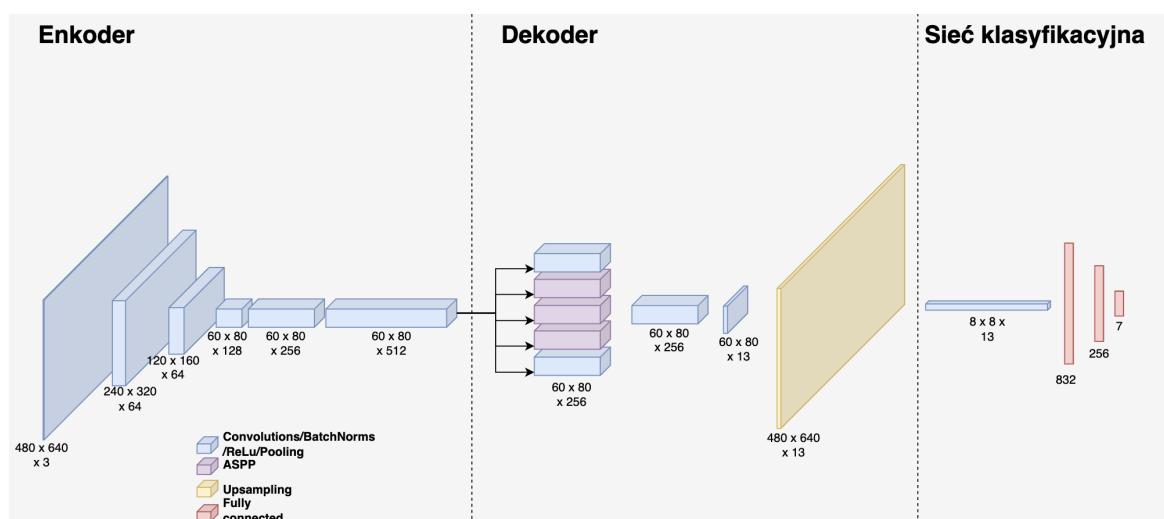
$$\lambda_{segmentacja} = 0$$

3.2.6. Bezpośrednia klasyfikacja z segmentacji

Rozwiązańem odbiegającym od reszty jest przeprowadzenie szeregowej klasyfikacji z segmentacją. Architektura przedstawia się zgodnie z rysunkiem 3.6. W tym eksperymencie sprawdzono, jak można skorzystać z gotowych predykcji dotyczących segmentacji. Model aż do głowy segmentacyjnej włącznie został zamrożony oraz wyłączony z optymalizacji. Zmieniają się tylko wagi części klasyfikacyjnej.

$$\lambda = \lambda_{segmentacja} + \lambda_{klasyfikacja}$$

$$\lambda_{segmentacja} = 0$$



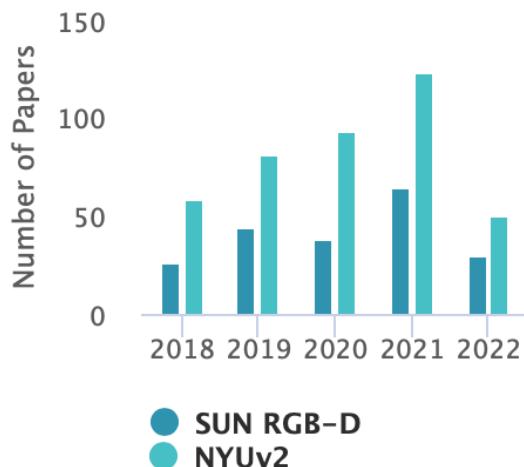
Rysunek 3.6. Architektura sieci szeregowej.

3.3. Zbiór danych

Dane są kluczową częścią głębokiego uczenia. Duży zbiór danych oznaczonych adnotacjami na poziomie pikseli jest potrzebny do wytrenowania wydajnego modelu segmentacji semantycznej. Typowe zestawy danych do segmentacji semantycznej to Cityscapes, PASCAL VOC i ADE20K. Podobnie w przypadku klasyfikacji sceny wymagany jest duży zbiór danych z odpowiednią informacją o etykiecie. Popularne zestawy danych do klasyfikacji scen obejmują NYUv2, SUN RGB-D, Matterport3D i ScanNet.

3.3.1. Wybór zbioru danych

Po prześledzeniu wielu zbiorów danych udało się sprostać wymaganiom pracy, uzy- skując dwa podobne zbiory danych - NYUv2 oraz SUN RGBD. Ostatecznie wybrano NYUv2. Trudno jednozycznie odpowiedzieć, który zbiór jest lepszy. Wykorzystano fakt cytowalno-ści. Okazuje się, że NYUv2 jest też częściej cytowany niż SUN RGBD (rys. 3.7), zatem to ten zbiór właśnie wybrano.



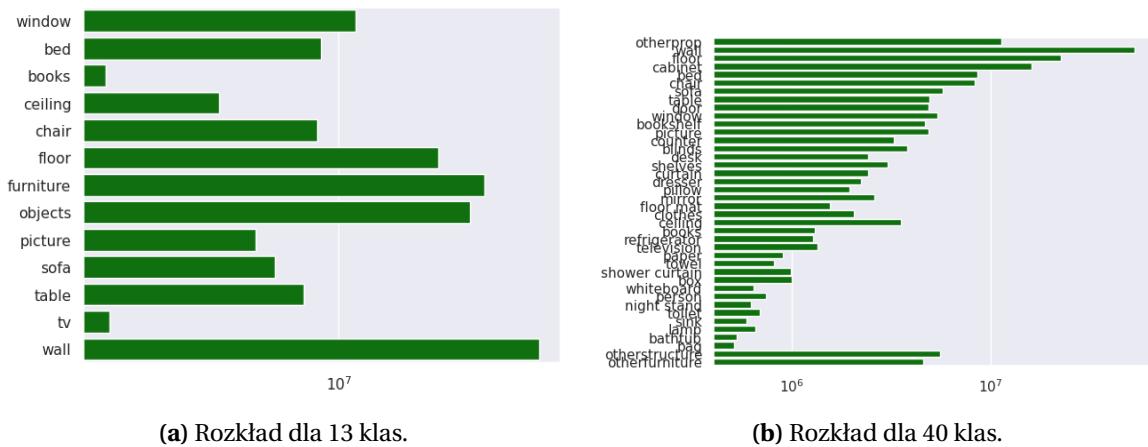
Rysunek 3.7. Szacowana liczba cytowań w latach 2018-2022 [paperswithcode.com]

3.3.2. Analiza zbioru danych

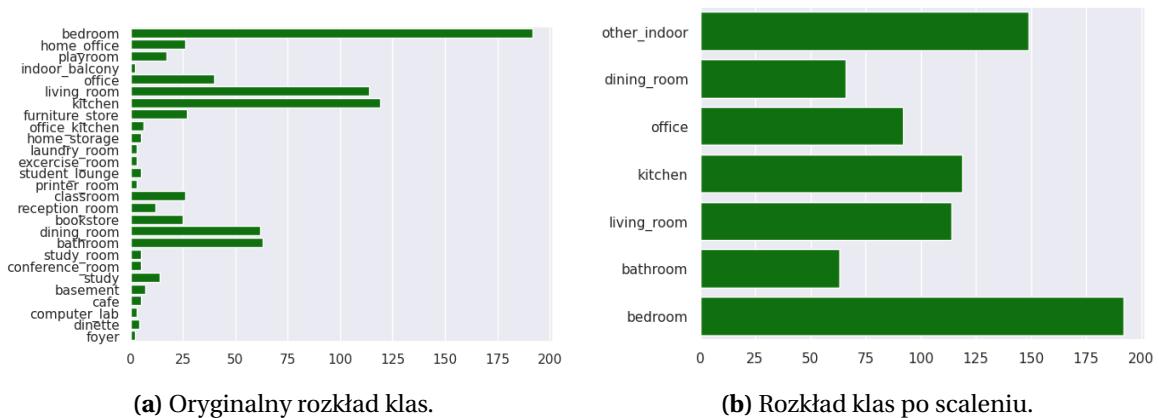
Eksploracyjna analiza danych (ang. EDA) to proces eksploracji i zrozumienia cech zbioru danych przed zbudowaniem modelu. Jednym z głównych powodów, dla których proces ten jest ważny w wizji komputerowej to fakt, że może pomóc w identyfikacji problemów ze zbiorem danych, takich jak nieprawidłowe etykiety. Co więcej, może ono być również wykorzystane do identyfikacji skośnych rozkładów klas prowadzących do niesprawiedliwych prognoz. EDA może być również wykorzystana do określenia, które kroki przetwarzania wstępnego (ang. preprocessing), takie jak augmentacja, są niezbędne do poprawy wydajności modelu wizji komputerowej. Badając dane i rozumiejąc ich charakterystykę, możemy uzyskać głębsze ich zrozumienie i zidentyfikować wszelkie problemy, które należy rozwiązać przed zbudowaniem modelu.

EDA przeprowadzone na zbiorze NYUv2 dostarczyło wielu interesujących obserwacji. W zbiorze domyślnie znajduje się 795 przykładów trenujących oraz 654 przykłady testowe. Ze zbioru testowego wyodrębniono zbiór walidacyjny stanowiący 20% zbioru testowego. Ponadto sprawdzono rozkład klas na przestrzeni całego zbioru danych. W przypadku za- dania segmentacji semantycznej do dyspozycji był wybór 894, 40 lub 13 klas przedmiotów. Im rozróżnialność była większa, tym większe okazywały się dysproporcje w rozkładzie. Histogramy dla 13 i 40 klas przedstawiono na rysunku 3.8. Podobna sytuacja miała miejsce

dla zadania klasyfikacji z tą różnicą, iż scalania klas należało dokonać ręcznie. Taki krok był kluczowy, gdyż pierwotny rozkład był silnie zdominowany przez kilka klas. Ostatecznie wybrano 13 klas dla klasyfikacji (rys. 3.9b) oraz scalone 7 dla segmentacji (rys. 3.9b).



Rysunek 3.8. Porównanie rozkładu ilości pikseli dla zadania segmentacji semantycznej.



Rysunek 3.9. Porównanie rozkładu klas dla zadania klasyfikacji sceny.

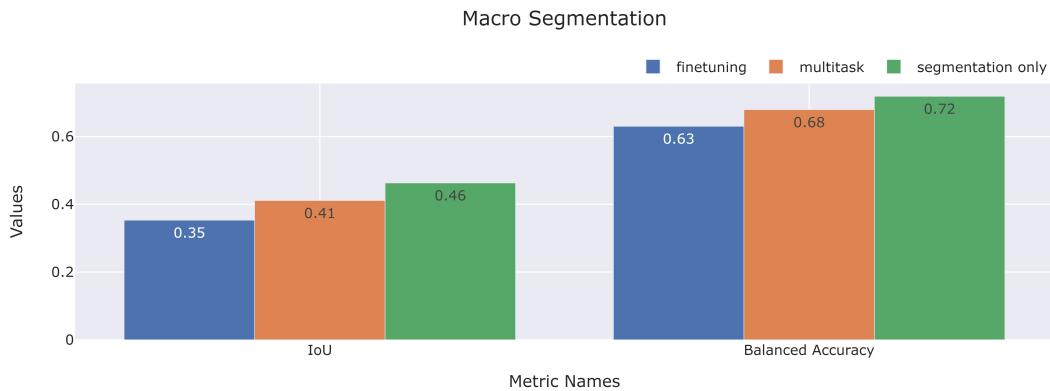
4. Wyniki

W tym rozdziale zostaną przedstawione empiryczne wyniki badań nad wspólną segmentacją semantyczną i klasyfikacją sceny w środowiskach wewnętrznych. Badania mają na celu opracowanie i ocenę różnych znanych i aktualnych technik uczenia głębokich sieci neuronowych. Aby to osiągnąć, przeprowadzono serię eksperymentów na zbiorze na reprezentacyjnych zbiorach danych. Analiza dotyczyła zarówno miar jakości sensu stricto, jak i miar wydajnościowych proponowanych metod. Rozważono różne metryki oceny, takie jak ogólna dokładność, indeks Jaccarda znany w literaturze jako intersection over union (IoU), miara F1 i wydajność obliczeniowa. Wyniki uzyskane w tym rozdziale zapewnią cenny wgląd w mocne strony i ograniczenia proponowanych metod.

4.1. Analiza miar jakości

W pierwszej kolejności metody zostaną zbadane pod względem wymienionych wcześniej miar jakości w postaci ogólnej — niezagregowanej, osobno dla segmentacji oraz klasyfikacji. Omawiane metryki należy rozumieć jako średnia miara jakości na każdej z klas, a więc makrośrednie. Makrośrednie metryki są stosowane przy ocenie wydajności algorytmów, ponieważ zapewniają bardziej wszechstronną ocenę ogólnej jakości algorytmu. Metryki makrośrednie uwzględniają wydajność algorytmu na wszystkich klasach obiektów i regionów w obrębie sceny, a nie tylko koncentrują się na jakości na najbardziej powszechnych lub najłatwiejszych do sklasyfikowania klasach. W przypadku stosowania metryki makrośredniej, jakość dla każdej klasy jest obliczana oddzielnie, a ogólna jakość jest obliczana jako średnia jakości poszczególnych klas. Stanowi to kontrast do metryki mikrośredniej, która oblicza ogólną jakość poprzez zsumowanie całkowitej liczby wyników dla wszystkich klas. Użycie makrośrednich metryk może być szczególnie ważne w scenariuszach, w których liczba instancji każdej klasy jest niezrównoważona lub gdy istnieje duża liczba klas. W takich przypadkach mikrośrednie metryki mogą być mylące, ponieważ mogą być pod silnym wpływem najbardziej powszechnych klas, podczas gdy zaniedbują te mniej powszechnie. Zatem makro analiza pokaże generalne rezultaty oraz otworzy dyskusję do dalszych, bardziej pogłębionych badań na rozważanym problemem.

Rozpoczynając od segmentacji, rozważamy 3 scenariusze testowe. Pierwszym z nich jest uczenie wyłącznie klasyfikacji rozumianej jako uczenie enkodera i sieci segmentacyjnej z pominięciem części klasyfikacyjnej. Pozwoli to odpowiedzieć na pytanie, czy bardziej zaawansowane techniki uczenia polepszają, a może pogorszą działanie modelu. Drugim scenariuszem jest uczenie wielozadaniowe, gdzie cały model jest odmrożony, a błąd jest propagowany zarówno przez segmentację, jak i klasyfikację. Ostatnim eksperymentem jest sprawdzenie technik transferu wiedzy, a szczególnie tak zwanego finetuningu. Model w pierwszym etapie uczy się przy zamożnym enkoderze, dopiero na koniec jest odmrażany w celu dostrojenia wyników.



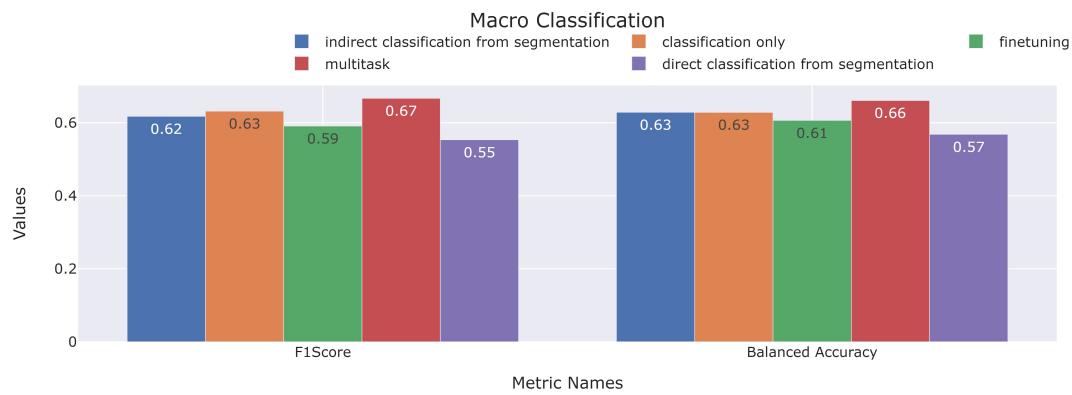
Rysunek 4.1. Porównanie miar IoU oraz dokładności dla segmentacji sceny.

Analizując rysunek 4.1 nie trudno zauważyc, że najlepsze rezultaty otrzymano w dla uczenia wyłącznie segmentacji. Kolejnym wynikiem jest uczenie wielozadaniowe. Jako najsłabsze podejście okazuje się metoda finetunowania. Relacje jakości są zachowane dla każdej z metryk, a więc zarówno dla miary IoU, jak i zbilansowanej dokładności (bAcc). Widać, że miara IoU wypada gorzej niż bAcc. Wyniki mogą sugerować, że trudno jest przeprowadzić transfer wiedzy z ImageNetu, gdyż finetuning wypada najsłabiej. Jest to najprawdopodobniej spowodowane zupełnie innym rozkładem klas na wspomnianej bazie. Analiza sceny w przeciwnieństwie do klasyfikacji najczęściej cechuje się długogonowym rozkładem klas. Drugim istotnym szczegółem jest fakt, iż wagi dekodera i głowy segmentacyjnej są losowe. Uczenie wielozadaniowe zgodnie z zakładanymi wynikami nie polepsza segmentacji, gdyż łączna przestrzeń segmentacji i klasyfikacji jest niewątpliwie bardziej skomplikowana i trudniejsza do optymalizacji.

Przechodząc do klasyfikacji, wyróżniamy 5 scenariuszy testowych. Pierwszym jest uczenie wyłącznie klasyfikacji, analogicznie jak wyżej, a więc przy wyłączonej części segmentacyjnej. Kolejnymi są wspomniane wcześniej uczenie wielozadaniowe oraz finetuning. Do nowych scenariuszy zaliczamy bezpośrednią oraz pośrednią klasyfikację z segmentacją.

Rezultaty przedstawia rysunek 4.2. Od razu da się zauważyć, że wyniki cechują mniejsze odchylenie standardowe oraz, analizując łącznie miarę F1 oraz zbalansowaną dokładność, średnia. Fakt ten jest prawdopodobnie wynikiem znacznie mniejszej ilości parametrów uczących. Najlepszy rezultat uzyskuje uczenie wielozadaniowe. Ciekawym wydaje się fakt, że uczenie wyłącznie klasyfikacji jest słabsze w tym przypadku. Prawdopodobnie poprzez uczenie wielozadaniowe enkoder wygenerował lepszą przestrzeń reprezentacji, co bezpośrednio wpływa na klasyfikację sceny. Najgorszym przypadkiem jest uczenie klasyfikacji bezpośrednio z segmentacji. Nie jest to dziwne, gdyż w tym przypadku klasyfikator korzystał z zaledwie 13 kanałów.

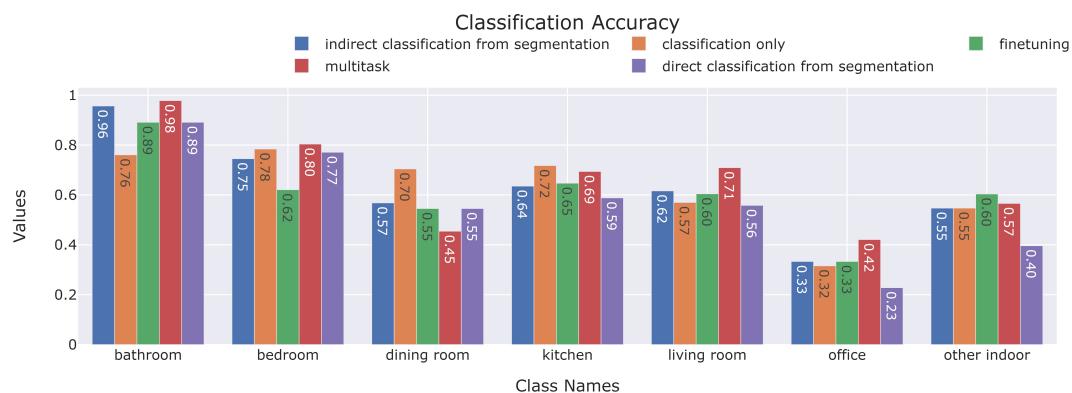
4. Wyniki



Rysunek 4.2. Porównanie miar F1 oraz dokładności dla klasyfikacji sceny.

Analizowanie jakości algorytmu dla każdej z klas osobno jest ważne, ponieważ pozwala na bardziej szczegółowe zrozumienie mocnych i słabych stron algorytmu. Rozważając ogólną jakość algorytmu przy użyciu metryki makrośredniej, nie jest od razu jasne, w których klasach algorytm radzi sobie dobrze, a w których źle. Analizując jakość każdej klasy osobno, można zidentyfikować konkretne klasy, z którymi algorytm ma problemy i podjąć kroki w celu poprawy wydajności w tych klasach.

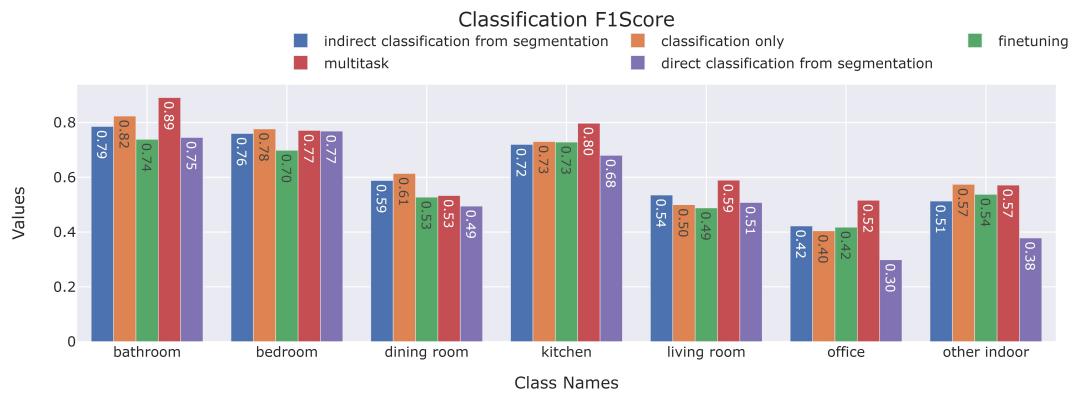
Rysunek 4.3 przedstawia dokładność dla każdej z klas dla zadania klasyfikacji sceny. Trudno jednoznacznie określić, która z metod sprawdza się tutaj najlepiej. Uczenie wielozadaniowe wypada najlepiej dla klas: łazienka, sypialnia, salon, biuro. Uczenie wyłącznej klasyfikacji jest najlepsze dla klas jadalnia oraz kuchnia. W pozostałych przypadkach klasa inne pomieszczenia jest najlepiej wykrywana przez scenariusz finetunowania. Uczenie klasyfikacji z segmentacji nigdy nie osiąga najlepszego wyniku. Biorąc pod uwagę miarę F1



Rysunek 4.3. Porównanie dokładności klasyfikacji sceny z rozróżnieniem konkretnych klas.

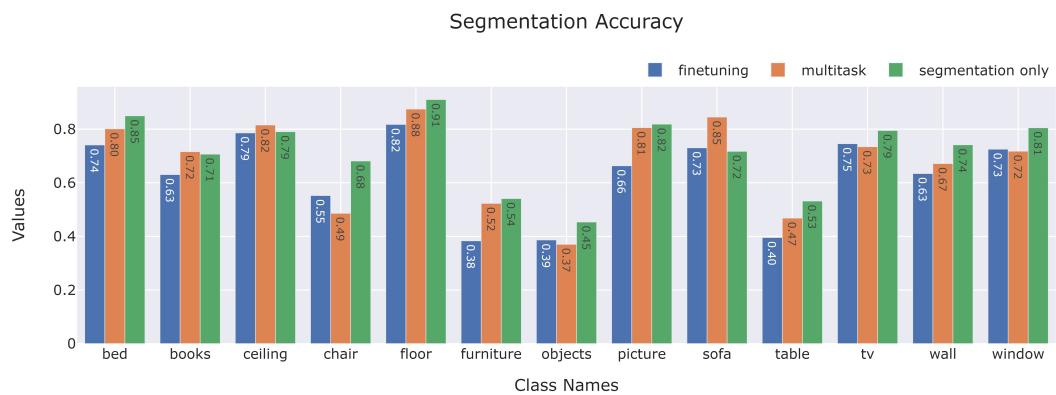
(rys. 4.4) również nie jesteśmy w stanie wyróżnić faworyzowanej metody. W porównaniu

z wcześniej analizowaną dokładnością widać, że uczenie wielozadaniowe utrzymuje w większości przypadków bardzo dobre rezultaty. Widać też, że wyniki w obrębie każdej z klas mało różnią się między sobą.



Rysunek 4.4. Porównanie miary F1 dla klasyfikacji sceny z rozróżnieniem konkretnych klas.

Analizując rysunek 4.5 przedstawiający dokładność w zadaniu segmentacji semantycznej, widać, że niektóre z zadań wypadają znacznie gorzej niż pozostałe. Sytuacja ta dotyczy klas meble, stoły, obiekty. Uczenie wyłącznie segmentacji okazało się najlepsze dla klas łóżko, podłoga, meble, obiekty, tv, ściana oraz okno. Stanowi to ponad połowę wszystkich możliwych klas. Uczenie wielozadaniowe uzyskało najlepsze wyniki dla klas książka, sufit, sofa. Przypadek funetunowania nigdy nie osiągnął najlepszego rezultatu.

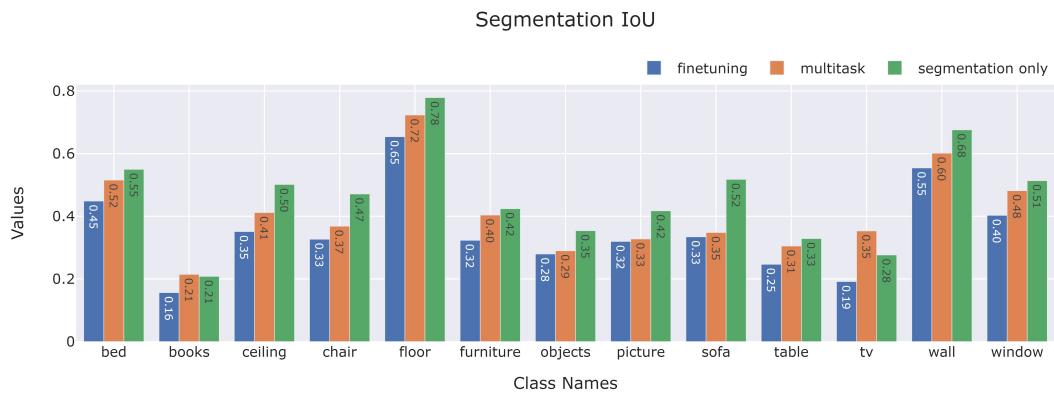


Rysunek 4.5. Porównanie dokładności segmentacji z rozróżnieniem konkretnych klas.

Na rysunku 4.6 przedstawiono IoU dla segmentacji semantycznej. Widać tutaj dużą dysproporcję między klasami podłoga, ściana, a pozostałymi klasami. Jest to zrozumiałe, klas te występują stosunkowo często na obrazie. Uczenie wyłącznie segmentacji uzyskuje

4. Wyniki

najlepsze wyniki na wszystkich klasach z wyłączeniem książek oraz telewizorów. W tych przypadkach najlepsze okazuje się uczenie wielozdaniowe.



Rysunek 4.6. Porównanie mocy IoU segmentacji z rozróżnieniem konkretnych klas.

4.2. Analiza czasowa

Ostatnio coraz częściej mówi się o zapotrzebowaniu na zasoby sprzętowe podczas uczenia maszynowego. Głębokie sieci neuronowe, a szczególnie te przetwarzające obrazy wymagają coraz więcej zasobów obliczeniowych do prawidłowego działania. Wynika to z dwóch głównych czynników. Po pierwsze duże modele wizji komputerowej posiadają miliony parametrów. Drugim powodem jest przetwarzanie wielu obrazów, które de facto są zbiorem macierzy. Wiedzie to do większego zainteresowania zużywanymi zasobami podczas treningu oraz wnioskowania. W tym podrozdziale przedstawiona zostanie analiza czasu treningu oraz wnioskowania.

Analizując czas uczenia w przypadku kolejnych metod, odkrywamy zalety finetuningu oraz uczenia wielozadaniowego (tab. 4.1). Suma czasów uczenia wyłącznie segmentacji oraz wyłącznie klasyfikacji (około 360s) znacznie przewyższa pozostałe metody. Najbardziej opłacalną czasowo metodą okazuje się finetuning. Jednakże na podstawie wyników mierzących jakość nie można uznać go za optymalny. Pozostają jeszcze dwie metody - nauczenie segmentacji oraz dalsze uczenie klasyfikacji (ok. 260 s. i 275s) oraz uczenie wielozdaniowe (ok. 211s). Segmentacja, a potem klasyfikacja osiąga najlepsze wyniki na segmentacji oraz przeciętnie na klasyfikacji. Z drugiej strony uczenie wielozdaniowe osiąga najlepsze rezultaty na klasyfikacji oraz drugi najlepszy wynik na segmentacji. Łącząc to z faktem znacznie krótszego uczenia, można wysunąć wniosek, że uczenie wielozdaniowe jest optymalne pod względem czasu treningu oraz uzyskiwanych rezultatów.

Porównanie czasu wnioskowania jest kluczowe z punktu widzenia korzystania z potencjału uczenia maszynowego. Tabela 4.2 przedstawia zestawienie czasu wnioskowania dla zestawu dwóch szeregowych sieci oraz jednej architektury wykonującej dwa zadania

nazwa zadania	czas[s]
wyłącznie segmentacja + wyłącznie klasyfikacja	~360
wyłącznie segmentacja + pośrednia klasyfikacja	~260
wyłącznie segmentacja + bezpośrednia klasyfikacja	~275
uczenie wielozadaniowe	~211
finetuning	~160

Tabela 4.1. Porównanie czasu uczenia względem całości.

naraz. Pierwszy przypadek obejmuje wykorzystanie wyłącznie klasyfikacji, a następnie wyłącznie segmentacji. W praktyce oznacza to dwukrotne podawanie danych do modelu i przejście przez 2 razy więcej parametrów niż w pozostałych przypadkach. Rozważając jedną architekturę osiągamy prawie dwukrotnie krótszy czas wnioskowania. Do tego przypadku zaliczamy wszystkie metody z uczeniem klasyfikacji na segmentacji, finetuning oraz uczenie wielozadaniowe.

rodzaj sieci	czas[s]
dwie szeregowe sieci	15.7
jedna architektura	8.6

Tabela 4.2. Porównanie czasu wnioskowania.

4.3. Analiza konkretnych przykładów

Analiza metryk, czy różnych miar jakości jest niezbędna do ewaluacji zadań uczenia maszynowego. Odpowiedni wybór tych miar gwarantuje pełen informacji wgląd, stanowiąc cenny wskazówka ewaluacyjne. Nie mniej nie wyklucza to istoty sprawdzenia rezultatów przez ludzkie oko. Mimo że trudno byłoby przeglądać i ewaluować wiele zdjęć w dużych zbiorach danych, przekrojowe sprawdzenie jest kluczowe w analizie. Dostarcza bowiem wielu cennych, nieujętych w matematycznych formułach obserwacji. W tym podrozdziale przedstawione zostaną rezultaty na wybranych zdjęciach.

4.3.1. Segmentacja semantyczna

Segmentacja semantyczna jest zadaniem niewątpliwie trudnym. Jednocześnie równie ciężko jest określić dobrą funkcję jakości, uwzględniającą takie właściwości jak gładkość, dokładność czy precyzja segmentacji. Można oczywiście korzystać z wielu funkcji jakości, jednak ostateczny werdykt warto przejrzeć ręcznie. W połączeniu z wiedzą dotyczącą między innymi trudności klasyfikacji danej grupy pikseli lub niejednoznacznością niektórych grup pikseli po obejrzeniu nawet kilkunastu zdjęć jesteśmy w stanie wysnuć pewne wnioski.

Łazienka

4. Wyniki

Analizując rysunek 4.7 widzimy, że klasa przedmioty (ang. objects) jest bardzo szeroko rozumiana przez twórców zbioru danych. Wynika z tego fakt, że grupa ta nie posiada ścisłe określonych cech, które byłyby łatwo identyfikowalne. Model w tym przypadku połączył w sposób szeroki omawianą klasę. Ciekawą obserwacją jest zaznaczenie przez model klasy krzesło. Po głębszej analizie można przypuszczać, że zlew ma podobną teksturę oraz kształt to metalowego krzesła. Klasy ściana, podłoga oraz meble zostały dość precyjnie sklasyfikowane.



Rysunek 4.7. Porównanie jakości segmentacji dla klasy łazienka.

Sytuacja jest równie interesująca w przypadku rysunku 4.8. Model dopatruje się klas meble w okolicach drzwi oraz przy zlewie. Pierwszy przypadek jest całkiem zrozumiały. Drewniane drzwi co do faktury mogą przypominać meble, na przykład drzwi od szafki. W drugiej sytuacji można domniemywać, że meble były często związane z umywalką czy nawet zlewem kuchennym, stąd model chętnie te klasy przydziela. Interesujące jest przydzielenie przez model etykiety obraz do włącznika światła.



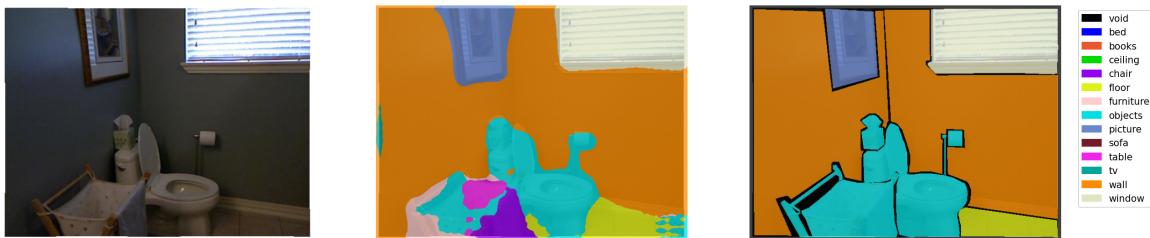
Rysunek 4.8. Porównanie jakości segmentacji dla klasy łazienka.

Ostatnim obraz, przedstawiający łazienkę pokazuje rysunek 4.9. Tak jak wcześniej wspomniano ściany oraz podłoga są często dobrze klasyfikowane. Tak też jest w tym przypadku. Kosz na pranie okazał się wyzwaniem. Model doszukiwał się tu takich obiektów jak stół, krzesło czy mebel.

Salon

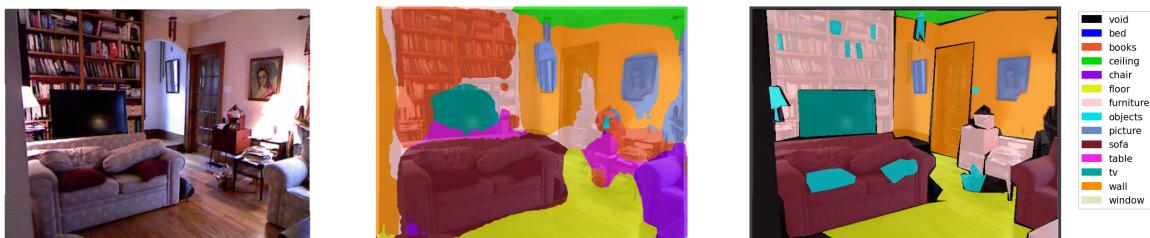
Salon jest najczęściej reprezentowany przez duży pokój, w którym znajdują się kanapa, stolik z przedmiotami, krzesła/fotele oraz ściany z zawieszonymi obrazkami. Nie brakuje tutaj mebli i wielu obiektów.

Rysunek 4.10 jest przykładem częstego problemu adnotacji zdjęć. Często okazuje się, że dana grupa pikseli przedstawia więcej niż jedną klasę. Obraz oczekiwany przedstawia



Rysunek 4.9. Porównanie jakości segmentacji dla klasy łazienka.

regał z książkami jako mebel. Model stwierdził jednak wcale się nie myląc, że są to książki. Trudno się nie zgodzić z tą predykcją. Oznacza to, że zbiór jest poniekąd wewnętrznie sprzeczny w jakiejś części. Widać, że częstym kłopotem jest odróżnienie mebli od stołu. Zadowala fakt pierwszoplanowej kanapy, która bez poduszek została bardzo dobrze sklasyfikowana. Równie dobre rezultaty otrzymujemy dla klasy podłoga, sufit oraz obrazy. Dziwi natomiast fakt zaznaczenia fotela jako krzesła.



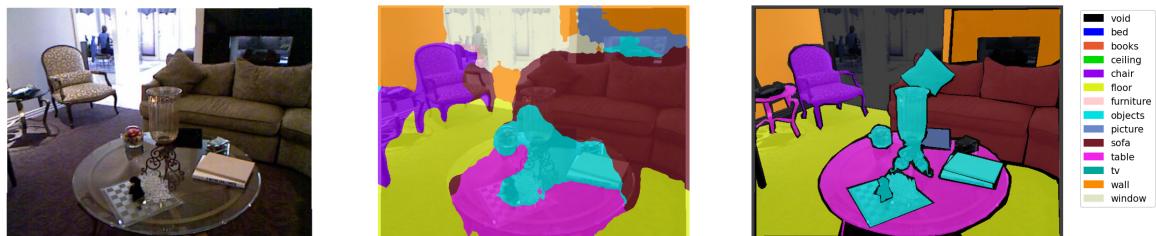
Rysunek 4.10. Porównanie jakości segmentacji dla klasy salon.

Na kolejnym rysunku 4.11 sytuacja jest nieco gorsza. Model miał trudności ze wskazaniem stolika na środku, który klasyfikuje jako część kanapy. Sporo problemów wygenerowała klasa krzesło. Obrazy, ściany i podłoga zostały poprawnie sklasyfikowane. Przedmioty drugoplanowe, szczególnie dalsze, a więc mniejsze pozostały dla modelu jednakie.



Rysunek 4.11. Porównanie jakości segmentacji dla klasy salon.

Scena salonu (rys. 4.12) jest znacznie lepiej sklasyfikowana niż poprzednia. Oprócz całkiem dobrze sklasyfikowanego stołu, obiektów i kanapy jest jeden ciekawy aspekt. Mianowicie obrazy docelowe znajdujące się w głębi obrazu zostały pominięte, czyli przedstawione jako pusta (ang. void). Mimo to klasyfikator celnie nadaje im klasy ściana oraz okna. To bardzo dobry prognostyk.



Rysunek 4.12. Porównanie jakości segmentacji dla klasy salon.

Sypialnia

Sypialnia to miejsce bardzo złożone. Jednak do charakterystycznych punktów tej sceny należą: łóżko, krzesło, meble oraz okno. Pożądany byłoby zatem osiągać na tych klasach satysfakcjonujące rezultaty. Na pierwszym planie rysunku 4.13 widać krzesło, stół oraz szafkę, które w przybliżeniu zostały całkiem dobrze sklasyfikowane. Brak zastrzeżeń budzą również klasę łóżko, podłoga oraz obiekty. Niewątpliwe ciekawe jest poprawne zaznaczenie okna, nawet w porze nocy. Jest to szczególnie cenna informacja, bo czarny prostokąt mógłby być zaklasyfikowany jako na przykład telewizor. Okno zostało zaznaczone zbyt szeroko, mianowicie fałszywie uznając prawdopodobnie lampa za okno. Prawdopodobnie kolor miał tu duże znaczenie.



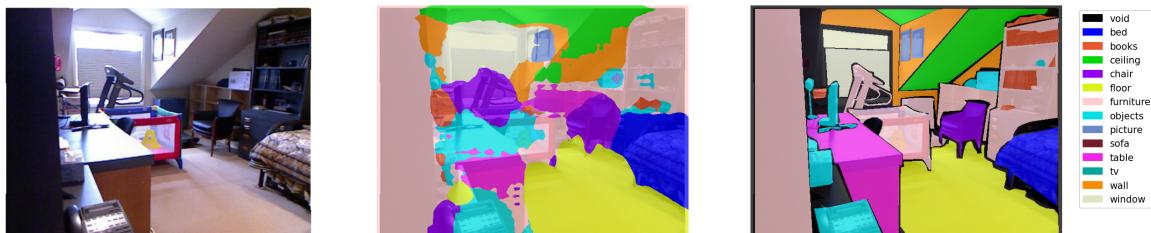
Rysunek 4.13. Porównanie jakości segmentacji dla klasy sypialnia.

Rysunek 4.14 to typowe zdjęcie sypialni. Czarną ramę łóżka model uznał za telewizor. Gdyby wyciąć samą tę ramę, wybór rzeczywiście nie byłby oczywisty. Poza tym łóżko zostało oznaczone całkiem poprawnie. Obrazy zostały poprawnie oznaczone. Na zdjęciu widać, całkiem poprawną próbę klasyfikacji krzesła.



Rysunek 4.14. Porównanie jakości segmentacji dla klasy sypialnia.

Ostatni rysunek (rys. 4.15) był większym wyzwaniem dla modelu. Widać to szczególnie w przypadku pierwszoplanowego biurka. Model nie był w stanie podjąć decyzji co do ostatecznej klasy. Standardowo podłoga oraz sufit zostały sklasyfikowane prawidłowo. Nie inaczej było w przypadku klasy łóżko.



Rysunek 4.15. Porównanie jakości segmentacji dla klasy sypialnia.

Jadalnia

Obrazy związane z jadalnią to głównie sceny związane ze stołami oraz krzesłami.

Taka sytuacja ma też miejsce na rysunku 4.16. Właściwie trudno tutaj znaleźć coś szczególnie interesującego. Cały obraz został całkiem dobrze pogrupowany. Wątpliwości budzi jedynie przypisanie do żyrandola klasy obrazy. Prawdopodobnie obrazy znajdujące się obok miały na to wpływ.



Rysunek 4.16. Porównanie jakości segmentacji dla klasy jadalnia.

Przypadek rysunku 4.17 wydaje się ciekawszym. Szczególnie warte uwagi są tutaj okna, na których znajdują się odbicia lustrzane. Refleksy są w wizji komputerowej zagadnieniem od dawna poruszany i znanym. Można jednoznacznie stwierdzić, że trudno sobie poradzić w takich sytuacjach. Model prawdopodobnie mając trudności z tym obszarem, przypisał go do klasy obiekt. Oprócz tego widzimy problemy z krzesłami w prawym dolnym rogu. Jasna, połyskująca skóra rzeczywiście przypomina nieco płytki podłogowe.

Ostatnim analizowanym obrazem w jadalni jest rysunek 4.18. Na pewno sklasyfikacja klas takich jak stół, krzesła czy okno jest tutaj poprawna. Co więcej nie można tego do tego grona niezaliczyć klasy podłoga oraz sufit. Jedyny problem z grupowaniem na tym zdjęciu dotyczy samego roku zdjęcia, gdzie nie przyporządkowano klasy mebel. Pozostałe instancje tej klasy są poprawnie sklasyfikowane.

Kuchnia

4. Wyniki



Rysunek 4.17. Porównanie jakości segmentacji dla klasy jadalnia.



Rysunek 4.18. Porównanie jakości segmentacji dla klasy jadalnia.

Obrazy przedstawiające kuchnie to głównie zabudowa kuchni oraz sprzęt kuchenny. Czasem występuje tutaj na przykład stół z krzesłami.

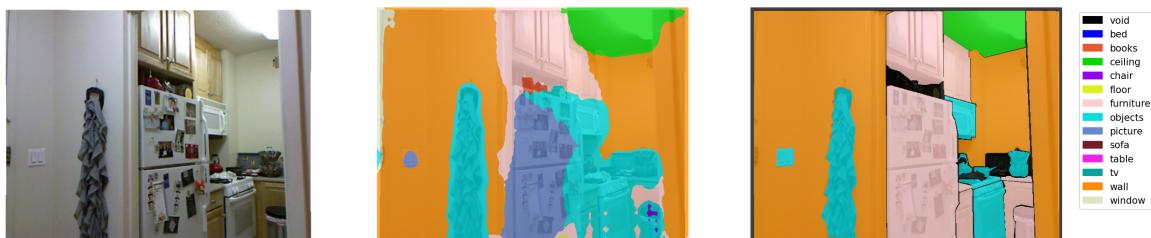
Obraz 4.19 nie wydaje się trudnym do klasyfikacji, jednak pojawiło się tutaj kilka kwestii wartych omówienia. Oprócz problemów z klasyfikacją stołu z prawej strony, który bardziej wygląda jak szafki z blatem w kuchni, obserwujemy błędne przypisanie tapety naściennej jako obrazy. Poza tym drewniane drzwi model klasyfikuje jako bardziej mebel niż ścianę, co ze względu na teksturę nie jest aż tak złym wyborem. Reszta zdjęcia została pogrupowana poprawnie.



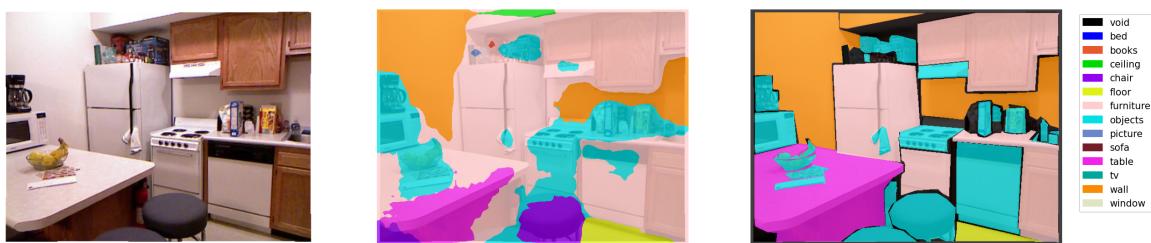
Rysunek 4.19. Porównanie jakości segmentacji dla klasy kuchnia.

Na rysunku 4.20 widzimy typową wąską kuchnię. Rezultaty są w miarę zadowalające poza przypisaniem lodówki do klasy obraz. Prawdopodobnie miały na to wpływ zdjęcia zawieszone na lodówce. Ściany, szafki i sufit zostały zaklasyfikowane prawidłowo.

Trzecim rysunkiem jest rys. 4.21. Największe wyzwanie stanowią tutaj obiekty zlokalizowane w różnych miejscach. Cieszy fakt, że mimo iż autorzy błędnie ocenili krzesło jako obiekt, model i tak zaznaczył je poprawnie. Widzimy tutaj również próbę klasyfikacji stołu. Powraca wtedy dyskusja na temat czy stół jest meblem tak jak został zaklasyfikowany przez model.



Rysunek 4.20. Porównanie jakości segmentacji dla klasy kuchnia.



Rysunek 4.21. Porównanie jakości segmentacji dla klasy kuchnia.

Biuro

Sceny związane z biurem najczęściej przedstawiają biurka z krzesłami, zarówno w faktycznych biurach, o których często świadczy wykładzina, jak i w domowych pokojach typu biuro.

Na rysunku 4.22 widać scenę przedstawiającą pokój z drukarkami. Model dość dobrze radzi sobie ze ścianami oraz z podłogą, której akurat w tym przypadku nie ma zbyt wiele. Ciekawa jest wizja autorów zbioru danych określających mapę jako obiekt zamiast obrazu. Może trudno bez wahania przypisać wiszącej mapie miano obrazu, ale na pewno szybciej można ją określić jako plakat co można tłumazyć na angielski jako picture.



Rysunek 4.22. Porównanie jakości segmentacji dla klasy biuro.

Rysunek 4.23 przedstawia salę konferencyjną. Lewa strona obrazu została zaklasyfikowana całkiem poprawnie. Wyzwaniem dla modelu okazał się prawy dolny róg, gdzie należało przypisać klasy mebel, obiekt, ściana, co model uprościł do po prostu mebla. To zdecydowanie zła klasyfikacja.

Ostatnim rysunkiem 4.24 jest pomieszczenie przedstawiające najprawdopodobniej biuro domowe. Klasyfikacja okna, podłogi oraz ściany była prawie bezbłędna. Gorzej

4. Wyniki



Rysunek 4.23. Porównanie jakości segmentacji dla klasy biuro.

model porawdził sobie z stołem, który po części sklasyfikował jako telewizor ze względu na bardzo ciemny oraz prostokąty charakter.

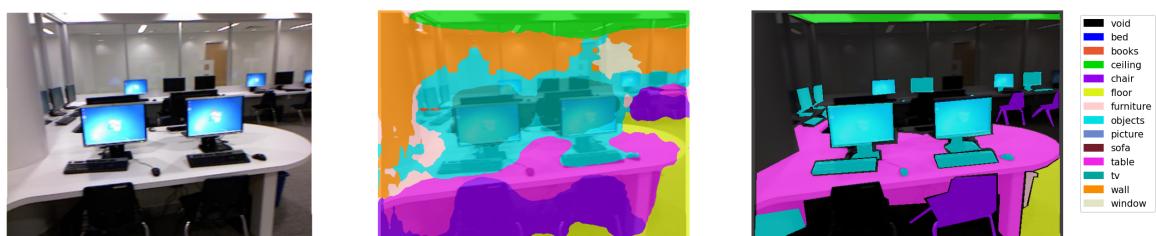


Rysunek 4.24. Porównanie jakości segmentacji dla klasy biuro.

Inne pomieszczenia

Sceny związane z klasą inne pomieszczenia budzą najwięcej wątpliwości. Nie wiadomo bowiem, co dokładnie może się tam znaleźć.

Na rysunku 4.25 znajduje się wspólna przestrzeń biurowa. Widzimy, że znajduje się tutaj wiele obszarów typu void, zatem model dokładnie nie wie co powinno się tam znaleźć. Dziwi to szczególnie w przypadku pierwszego krzesła po prawej stronie. Nie mniej jednak model dość dobrze zgaduję tę klasę. Jest zrozumiałym, że pokój otoczony ścianami. W gruncie rzeczy szklana szyba oczywiście jest ścianą w tym przypadku. Model niezbyt dobrze pogrupował klasę obiekty. Jest tutaj wiele do poprawy.



Rysunek 4.25. Porównanie jakości segmentacji dla klasy inne pomieszczenia.

Rysunek 4.26 przedstawia pomieszczenie biurowe. Wszystkie obrazki zostały zaklasyfikowane poprawnie. Okna zostały przypisane jako obrazy. Model dobrze pogrupował człowieka. Wyzwanie stanowiła klasa obiekty.



Rysunek 4.26. Porównanie jakości segmentacji dla klasy inne pomieszczenia.

Ostatnim analizowanym obrazem w jadalni jest rysunek 4.18. Na pewno klasyfikacja stołu, krzesła czy okna jest tutaj poprawna. Co więcej, nie można tego do tego grona nie zaliczyć klasy podłoga oraz sufit. Jedyny problem z grupowaniem na tym zdjęciu dotyczy samego rogu zdjęcia, gdzie nie przyporządkowano klasy mebel. Pozostałe instancje tej klasy są poprawnie sklasyfikowane.



Rysunek 4.27. Porównanie jakości segmentacji dla klasy inne pomieszczenia.

4.3.2. Klasyfikacja sceny

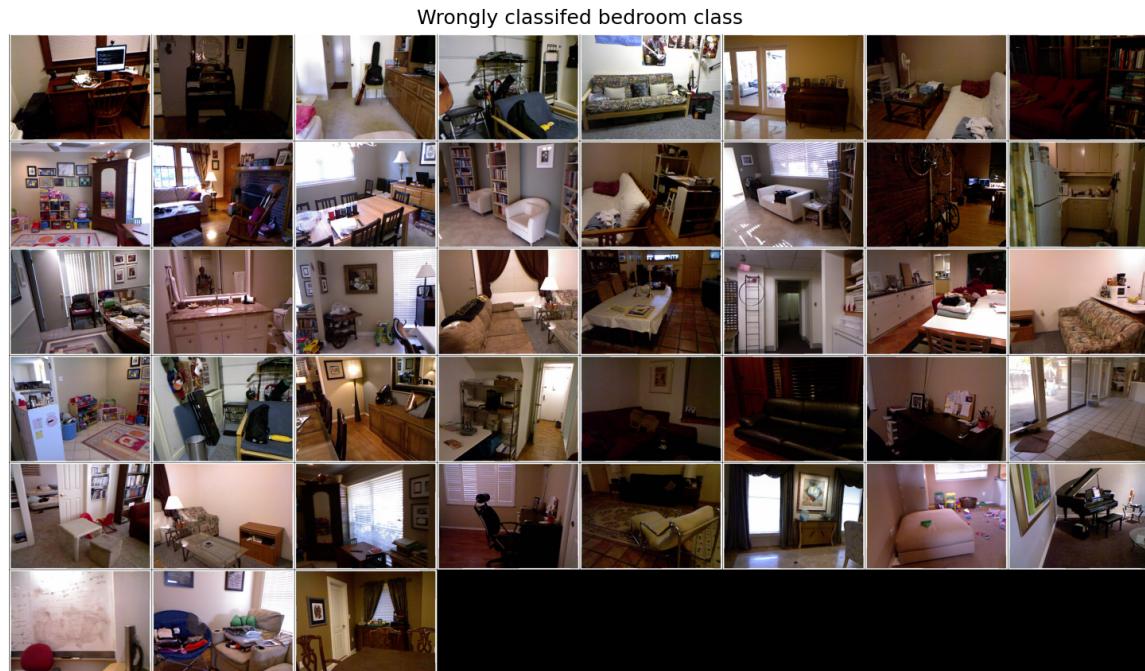
Podobnie jak w przypadku segmentacji semantycznej czasem trudno jest jednoznacznie określić jakość modelu, bazując wyłącznie na miarach jakości. W niniejszym rozdziale zostaną przytoczone wszystkie błędne klasyfikacje z podziałem na konkretne klasy. Pozwoli to wysnuć pewne obserwacje na temat podobieństw tych klas oraz pomoże wysunąć wnioski co do tych błędów. Co więcej, przedstawione zostaną statystyki błędnej klasyfikacji, aby lepiej zobrazować te błędy.

Na rysunku 4.28 przedstawiono 10 błędnych przypisań dla klasy łazienka. Dziewięć z dziesięciu błędów dotyczyło klasy kuchnia. Można doszukiwać się, że kuchnia, jak i łazienka ma poniekąd podobny schemat. Na pewno występuje te same klasy jak zlew czy meble. Tylko raz klasyfikator uznał, że sypialnia jest łazienką. Najwięcej pomyłek algorytm popełnił na klasie sypialnia (rys. 4.29). Na pewno wynika to z faktu, iż była to klasa dominująca. Co drugi błąd następował na klasie sypialnia. Szczególnie często gdy łóżkiem była kanapa lub na zdjęciu występował fotel. Ponad 32% błędów w sumie stanowiły klasy biuro oraz jadalnia. Można przypuszczać, że tym razem kluczowym elementem świadczącym o predykcji był stół. Jadalnia była pomylona w sumie 11 razy (rys. 4.30). Zgodnie z przedstawionym rysunkiem, na większości zdjęć występuje stół i krzesła.

4. Wyniki



Rysunek 4.28. Porównanie jakości klasyfikacji dla klasy łazienka.



Rysunek 4.29. Porównanie jakości klasyfikacji dla klasy sypialnia.



Rysunek 4.30. Porównanie jakości klasyfikacji dla klasy jadalnia.

Najmniej pomyłek jest dla klasy kuchnia (rys. 4.31). Klasy inne pomieszczenia oraz kuchnie zostały błędnie zaklasyfikowane jako biuro w większości przypadków. Trudno określić, skąd akurat wynikają takie rezultaty.

Model najczęściej błędnie przypisywał klasę inne pomieszczenia dla biura w mniej niż połowie przypadków. Pozostałe przypadki należą do klas sypialnia oraz jadalnia. Klasa



Rysunek 4.31. Porównanie jakości klasyfikacji dla klasy kuchnia.



Rysunek 4.32. Porównanie jakości klasyfikacji dla klasy biuro.

inne pomieszczenia jest szczególnie narażona na pomyłki, gdyż to połączenie najróżniejszych klas scen.



Rysunek 4.33. Porównanie jakości klasyfikacji dla klasy inne pomieszczenia.

Analiza błędnie sklasyfikowanych scen dostarczyła wielu ważnych informacji. Najczęściej przyczyną błędów było znaczne podobieństwo występujących klas przedmiotów między różnymi klasami scen.

5. Podsumowanie

W tym rozdziale analizie poddany będzie całokształt pracy. Przedstawione zostaną wnioski z wynikające z poprzedniego rozdziału. Na końcu praca zostanie podsumowana, przedstawiając ostateczne konkluzje.

5.1. Wnioski

Środowiska wewnętrzne to unikalny zestaw wyzwań dla zadania łącznej klasyfikacji oraz segmentacji semantycznej. Wiele różnych elementów w wielu skalach stanowi wyzwanie nawet dla najnowocześniejszych algorytmów sztucznej inteligencji. Nie można ukryć też faktu, że różnice w wyglądzie pomieszczeń nie sprzyjają osiąganiu wysokich wyników. Szczególnie utrudniające okazało się samo oznaczenie danych, które czasem było sprzeczne lub nieodpowiednie. Występował tutaj szereg problemów. Po pierwsze dla zadania segmentacji etykiety takie jak obiekt były bardzo trudne do rozwiązania. Za tą nazwą kryło się wszystko, co nie mieściło się w ramach innych etykiet. Nie jest to pożądane, ponieważ trudno znaleźć wspólną reprezentację dla tak szeroko pojętej etykiety. Innym problem jest występowanie klasy stół oraz mebel. Zachodzi pytanie, czy stół nie jest meblem? Z analizy przykładów wynikało, że klasy te często były mylone. Inny problem okazały się pojedyncze przykłady ze zbioru danych, takie jak regał z książkami. Model zaznacza regał jako książki, co jest rzeczywiście prawdą, gdyż tam znajdowały się książki. Do klasy mebel powinno zaliczyć się tylko drewniane części regału, a nie koniecznie jego zawartość. Innym przykładem jest zaznaczenie plakatu mapy, który został zaklasyfikowany przez autorów jako obiekt, a nie konkretne obraz.

Pomimo tych wyzwań, przedstawiona w niniejszej pracy architektura uczenia wielozadaniowego wykazała zdolność do skutecznego radzenia sobie z tymi trudnościami. Nie można stwierdzić, że uczenie wielozadaniowe pod każdym aspektem osiąga najwyższe rezultaty, jednak uważam, że w wielu zastosowaniach będzie to najbardziej optymalne rozwiązanie. Uczenie wyłącznie klasyfikacji czy segmentacji pozwoliło porównać rezultaty uzyskane na pojedynczych sieciach z architekturą wielozadaniową.

Makrośrednie miary jakości pokazały, że w przypadku segmentacji semantycznej najlepsze jest podejście uczenia samej segmentacji. Uczenie wielozadaniowe jest nieznacznie gorsze. Jednakże uczenie wielozadaniowe osiągnęło lepszy rezultat w zadaniu klasyfikacji sceny niż inne metody uczenia w tym uczenie wyłącznie klasyfikacji. W przytaczanych artykułach ([14], [15]) autorzy osiągają tę samą dokładność na segmentacji i znacznie lepsze rezultaty dla zadania klasyfikacji ucząc te zadania łącznie. Uważam, że na tej podstawie można twierdzić, że uczenie wielozadaniowe konkretnie dla zadania segmentacji semantycznej i klasyfikacji sceny dla domeny scen wewnętrz wpływa dodatkowo na zadanie klasyfikacji. Zadanie segmentacji semantycznej dostarcza dużo więcej informacji dla funkcji straty niż klasyfikacja sceny, gdyż w pierwszym przypadku klasyfikujemy każdy

piksel na obrazie, a w drugim cały obraz. W mojej opinii zachodzi tutaj rozszerzenie zbioru danych oraz tak zwane podsłuchiwanie, o których pisał Ruder [9]. Segmentacja semantyczna dostarcza zadaniu klasyfikacji dodatkowe informacje, które ostatecznie polepszają jakość modelu.

Uczenie wielozadaniowe jest bardzo korzystne z punktu widzenia wydajności czasowej. Model korzysta z 2 razy mniej parametrów. Wpłynęło to bezpośrednio na czas uczenia oraz co najważniejsze wnioskowania. Oczywiście jest, że urządzenia IoT czy robotyka mobilna ograniczają zasoby sprzętowe. Jednocześnie kluczowa czy jest tam czas reakcji. Dwukrotne przyspieszenie jest zatem tym cenniejsze.

Oprócz wniosków bezpośrednio wynikających z uczenia wielozadaniowego, podczas eksperymentów można było zaobserwować różne ciekawe zdarzenia. Ciekawym okazał się transfer wiedzy, który wypadał najsłabiej. Nie jest to dziwne. Baza ImageNet znacząco odbiega od prezentowanych obrazów pomieszczeń. Przede wszystkim obrazy z tej bazy są skoncentrowane na pojedynczym obiekcie. Zdjęcia pomieszczeń to zdjęcie scen, a więc wielu obiektów rozdystrybuowanych na całym obrazie. Poza tym rozkład związany z pomieszczeniami jest długooognowy. Większość obrazu jest często zdominowana przez podłogę czy ściany.

Innym eksperymentem było uczenie pośrednio oraz bezpośrednio z segmentacji. W pierwszym przypadku skorzystano z wyjścia enkodera jako przestrzeni reprezentacji. Wyniki były wprawdzie gorsze niż uczenia wielozadaniowego, ale prawie tak samo dobre, jak w przypadku uczenia wyłącznie klasyfikacji. Oznacza to, że przestrzeń reprezentacji wygenerowana przez segmentację semantyczną nadaje się i może być stosowana do wyuczenia klasyfikacji. Przypadek bezpośredniej klasyfikacji z segmentacji jest najgorszym. Przestrzeń reprezentacji składająca się wyłącznie z 13 kanałów to zdecydowanie za mało, by reprezentować sceny.

5.2. Podsumowanie

W pracy opracowano model oparty o głębokie uczenie, który jednocześnie segmentował semantycznie pomieszczenia oraz przyporządkowywał im rodzaj sceny. Udało się zrealizować łączne uczenie obu zadań. Co więcej, uczenie wielozadaniowe okazało się często bardzo korzystnym rozwiązaniem w porównaniu z metodami klasycznymi. Szczególnie cenne we wspólnej architekturze jest wydajność czasowa, która wpływa na czas uczenia oraz wnioskowanie modelu. Przestrzeń reprezentacji enkodera po wytrenowaniu segmentacji semantycznej może z dużą skutecznością być użyta do zadania klasyfikacji sceny. Korzystanie z metod finetunu i bezpośrednią klasyfikacji nie przynosi oczekiwanych rezultatów.

Cel pracy został zrealizowany z dużym powodzeniem. Co więcej, uzyskana została odpowiedź na pytania badawcze postawione w pierwszej jej części. Uważam, że wiele sce-

5. Podsumowanie

nariuszy testowych pozwoliło rzetelnie odpowiedzieć na wątpliwości związane z jakością wspólnego uczenia, pokazując jej zalety oraz ograniczenia.

Bibliografia

- [1] D. Zeng, M. Liao, M. Tavakolian, Y. Guo, B. Zhou, D. Hu, M. Pietikäinen i L. Liu, “Deep learning for scene classification: A survey”, *arXiv preprint arXiv:2101.10531*, 2021.
- [2] H. Zhang, K. Dana, J. Shi, Z. Zhang, X. Wang, A. Tyagi i A. Agrawal, “Context encoding for semantic segmentation”, w *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2018, s. 7151–7160.
- [3] A. Krizhevsky, I. Sutskever i G. E. Hinton, “Imagenet classification with deep convolutional neural networks”, *Communications of the ACM*, t. 60, nr. 6, s. 84–90, 2017.
- [4] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke i A. Rabinovich, “Going deeper with convolutions”, w *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, s. 1–9.
- [5] K. He, X. Zhang, S. Ren i J. Sun, “Deep residual learning for image recognition”, w *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, s. 770–778.
- [6] J. Long, E. Shelhamer i T. Darrell, “Fully convolutional networks for semantic segmentation”, w *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, s. 3431–3440.
- [7] O. Ronneberger, P. Fischer i T. Brox, “U-net: Convolutional networks for biomedical image segmentation”, w *International Conference on Medical image computing and computer-assisted intervention*, Springer, 2015, s. 234–241.
- [8] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy i A. L. Yuille, “DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, t. 40, nr. 4, s. 834–848, 2018. DOI: 10.1109/TPAMI.2017.2699184.
- [9] S. Ruder, “An overview of multi-task learning in deep neural networks”, *arXiv preprint arXiv:1706.05098*, 2017.
- [10] R. Caruana, “Multitask learning: A knowledge-based source of inductive bias1”, w *Proceedings of the Tenth International Conference on Machine Learning*, Citeseer, 1993, s. 41–48.
- [11] T. Standley, A. Zamir, D. Chen, L. Guibas, J. Malik i S. Savarese, “Which tasks should be learned together in multi-task learning?”, w *International Conference on Machine Learning*, PMLR, 2020, s. 9120–9132.
- [12] J. Yao, S. Fidler i R. Urtasun, “Describing the scene as a whole: Joint object detection, scene classification and semantic segmentation”, w *2012 IEEE conference on computer vision and pattern recognition*, IEEE, 2012, s. 702–709.
- [13] S. Iizuka, E. Simo-Serra i H. Ishikawa, “Let there be color! Joint end-to-end learning of global and local image priors for automatic image colorization with simultaneous classification”, *ACM Transactions on Graphics (ToG)*, t. 35, nr. 4, s. 1–11, 2016.

5. Bibliografia

- [14] S. Mehta, E. Mercan, J. Bartlett, D. Weaver, J. G. Elmore i L. Shapiro, "Y-Net: joint segmentation and classification for diagnosis of breast biopsy images", w *International Conference on Medical Image Computing and Computer-Assisted Intervention*, Springer, 2018, s. 893–901.
- [15] D. Seichter, S. B. Fischedick, M. Köhler i H.-M. Groß, "Efficient Multi-Task RGB-D Scene Analysis for Indoor Environments", w *2022 International Joint Conference on Neural Networks (IJCNN)*, 2022, s. 1–10. DOI: 10.1109/IJCNN55064.2022.9892852.
- [16] L. Huang, J. Qin, Y. Zhou, F. Zhu, L. Liu i L. Shao, "Normalization techniques in training dnns: Methodology, analysis and application", *arXiv preprint arXiv:2009.12836*, 2020.
- [17] T. Akiba, S. Sano, T. Yanase, T. Ohta i M. Koyama, "Optuna: A Next-generation Hyperparameter Optimization Framework", w *Proceedings of the 25rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2019.
- [18] M. Berman, A. R. Triki i M. B. Blaschko, "The lovász-softmax loss: A tractable surrogate for the optimization of the intersection-over-union measure in neural networks", w *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, s. 4413–4421.
- [19] S. Jadon, "A survey of loss functions for semantic segmentation", w *2020 IEEE conference on computational intelligence in bioinformatics and computational biology (CIBCB)*, IEEE, 2020, s. 1–7.

Spis rysunków

2.1 Problem różnorodności wewnętrz klasowej oraz wieloznaczności semantycznej [1].	9
2.2 Segmentacja wewnętrz pomieszczeń [2].	10
2.3 Klasyczna architektura DeepLabV3 z backbonem ResNet34.	13
3.1 Describing the Scene as a Whole: Joint Object Detection, Scene Classification and Semantic Segmentation (2012) [12].	17
3.2 Let there be Color!: Joint End-to-end Learning of Global and Local Image Priors for Automatic Image Colorization with Simultaneous Classification 2016 [13].	18
3.3 Y-Net: Joint Segmentation and Classification for Diagnosis of Breast Biopsy Images 2018 [14].	19
3.4 Efficient Multi-Task RGB-D Scene Analysis for Indoor Environments [15]	20
3.5 Architektura wielozadaniowej sieci.	20
3.6 Architektura sieci szeregowej.	23
4.1 Porównanie miar Iou oraz dokładności dla segmentacji sceny.	27
4.2 Porównanie miar F1 oraz dokładności dla klasyfikacji sceny.	28
4.3 Porównanie dokładności klasyfikacji sceny z rozróżnieniem konkretnych klas.	28
4.4 Porównanie miary F1 dla klasyfikacji sceny z rozróżnieniem konkretnych klas.	29
4.5 Porównanie dokładności segmentacji z rozróżnieniem konkretnych klas.	29
4.6 Porównanie miary IoU segmentacji z rozróżnieniem konkretnych klas.	30
4.7 Porównanie jakości segmentacji dla klasy łazienka.	32
4.8 Porównanie jakości segmentacji dla klasy łazienka.	32
4.9 Porównanie jakości segmentacji dla klasy łazienka.	33
4.10 Porównanie jakości segmentacji dla klasy salon.	33
4.11 Porównanie jakości segmentacji dla klasy salon.	33
4.12 Porównanie jakości segmentacji dla klasy salon.	34
4.13 Porównanie jakości segmentacji dla klasy sypialnia.	34
4.14 Porównanie jakości segmentacji dla klasy sypialnia.	34
4.15 Porównanie jakości segmentacji dla klasy sypialnia.	35
4.16 Porównanie jakości segmentacji dla klasy jadalnia.	35
4.17 Porównanie jakości segmentacji dla klasy jadalnia.	36
4.18 Porównanie jakości segmentacji dla klasy jadalnia.	36
4.19 Porównanie jakości segmentacji dla klasy kuchnia.	36
4.20 Porównanie jakości segmentacji dla klasy kuchnia.	37
4.21 Porównanie jakości segmentacji dla klasy kuchnia.	37
4.22 Porównanie jakości segmentacji dla klasy biuro.	37
4.23 Porównanie jakości segmentacji dla klasy biuro.	38
4.24 Porównanie jakości segmentacji dla klasy biuro.	38

4.25 Porównanie jakości segmentacji dla klasy inne pomieszczenia.	38
4.26 Porównanie jakości segmentacji dla klasy inne pomieszczenia.	39
4.27 Porównanie jakości segmentacji dla klasy inne pomieszczenia.	39
4.28 Porównanie jakości klasyfikacji dla klasy łazienka.	40
4.29 Porównanie jakości klasyfikacji dla klasy sypialnia.	40
4.30 Porównanie jakości klasyfikacji dla klasy jadalnia.	40
4.31 Porównanie jakości klasyfikacji dla klasy kuchnia.	41
4.32 Porównanie jakości klasyfikacji dla klasy biuro.	41
4.33 Porównanie jakości klasyfikacji dla klasy inne pomieszczenia.	41

Spis tabel

4.1 Porównanie czasu uczenia względem całości.	31
4.2 Porównanie czasu wnioskowania.	31