

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет інформаційно-аналітичних технологій та менеджменту

(повна назва)

Кафедра прикладної математики

(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА Пояснювальна записка

рівень вищої освіти другий (магістерський)

Сегментація зображень в реальному часі

для автономних транспортних засобів

(тема)

Виконав:

здобувач 2 курсу, групи ПМм-23-2

Лозицький А.О.

(прізвище, ініціали)

Спеціальність 113 Прикладна математика

(код і повна назва спеціальності)

Тип програми освітньо-професійна

(освітньо-професійна або освітньо-наукова)

Освітня програма Прикладна математика

(повна назва освітньої програми)

Керівник доц. Ламтюгова С.М.

(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри ПМ

(підпис)

Сидоров М.В.

(прізвище, ініціали)

2025 р.

Харківський національний університет радіоелектроніки

Факультет інформаційно-аналітичних технологій та менеджменту

Кафедра прикладної математики

Рівень вищої освіти другий (магістерський)

Спеціальність 113 Прикладна математика

(код і повна назва)

Тип програми освітньо-професійна

(освітньо-професійна або освітньо-наукова)

Освітня програма Прикладна математика

(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри ПМ _____

(підпис)

“ 25 ” листопада 2024 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві Лозицькому Арсенію Олександровичу
(прізвище, ім'я, по батькові)

1. Тема роботи Сегментація зображень в реальному часі для автономних транспортних засобів

затверджена наказом по університету від 22 листопада 2024 р. № 1223 Ст

2. Термін подання здобувачем роботи до екзаменаційної комісії 6 січня 2025 р.

3. Вихідні дані до роботи синтетичний датасет Synscapes та датасет з реальних фотографій Cityscapes

4. Перелік питань, що потрібно опрацювати в роботі _____

1. Аналіз предметної області

2. Вибір і обґрунтування методу розв'язання

3. Програмна реалізація

4. Результати обчислювального експерименту

5. Аналіз можливих застосувань

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій _____

1. Актуальність теми роботи _____

2. Постановка задачі _____

3. Аналіз предметної області _____

4. Метод чисельного аналізу _____

5. Результати обчислювального експерименту _____

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Підбір та вивчення технічної літератури за темою роботи	25 листопада – 1 грудня 2024 р.	виконано
2	Вибір та обґрунтування методу	2 – 8 грудня 2024 р.	виконано
3	Розробка алгоритму і програми	9 – 22 грудня 2023 р.	виконано
4	Проведення аналітичних досліджень та розрахунків	23 – 29 грудня 2024 р.	виконано
5	Робота над текстом пояснювальної записки	30 грудня 2024 р. – 9 січня 2025 р.	виконано
6	Представлення роботи на рецензію в ЕК	10 січня 2025 р.	виконано

Дата видачі завдання 25 листопада 2024 р.

Здобувач _____
(підпис)

Керівник роботи _____ доц. Ламтюгова С.М.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка: 57 с., 1 табл., 4 рис., 1 дод., 26 джерел.

АВТОНОМНІ ТРАНСПОРТНІ ЗАСОБИ, МАШИННЕ НАВЧАННЯ, НЕЙРОННІ МЕРЕЖІ, ОПТИМІЗАЦІЯ МОДЕЛЕЙ, РЕАЛЬНИЙ ЧАС, СЕГМЕНТАЦІЯ ЗОБРАЖЕНЬ, ТОЧНІСТЬ ПІКСЕЛІВ, DEEPLAB, SYNSCAPES, TENSORRT.

Об'єкт дослідження – системи сегментації зображень для автономних транспортних засобів, що працюють в реальному часі.

Мета роботи – розробка методу сегментації зображень для автономних транспортних засобів, який забезпечуватиме високу точність розпізнавання об'єктів на дорозі в умовах реального часу, зокрема при обмежених обчислювальних ресурсах.

Методи дослідження – методи комп'ютерного зору, глибокого навчання, нейронні мережі, а також оптимізаційні техніки, такі як квантове навчання (Quantization) і скорочення з'єднань (Pruning) для адаптації моделей до роботи на вбудованих системах з обмеженими ресурсами.

Кваліфікаційна робота присвячена дослідженню методів сегментації зображень для автономних транспортних засобів, зокрема застосуванню архітектури DeepLabv3+ для досягнення високої точності сегментації об'єктів у реальному часі. Робота також розглядає підходи до оптимізації нейронних мереж для ефективної роботи на вбудованих системах, що мають обмежені обчислювальні ресурси.

У роботі проведено детальний аналіз існуючих підходів до сегментації зображень і обґрунтовано вибір DeepLabv3+ як основної архітектури. Значна увага приділяється навчанню моделі на синтетичних і реальних наборах даних, таких як Synscapes і Cityscapes, що дозволило досягти високих показників точності. Крім того, робота демонструє процес адаптації моделей до реальних умов шляхом тонкого налаштування та використання сучасних технік оптимізації.

Результати експериментів показують, що запропонований підхід дозволяє забезпечити сегментацію об'єктів з високою точністю та продуктивністю навіть на вбудованих системах. Це відкриває можливості для подальшого застосування таких моделей у системах автономного керування, підвищуючи їхню надійність і ефективність.

ABSTRACT

Introductory note: 57 pages, 1 table, 4 figures, 1 appendix, 26 sources.

AUTONOMOUS VEHICLES, DEEPLAB, IMAGE SEGMENTATION, MACHINE LEARNING, MODELS OPTIMIZATION, NEURAL NETWORKS, PIXEL ACCURACY, REAL-TIME, SYNSCAPES, TENSORRT.

The object of research is image segmentation systems for autonomous vehicles operating in real time.

Purpose of the study is to develop an image segmentation method for autonomous vehicles that ensures high accuracy in object recognition on the road under real-time conditions, particularly with limited computational resources.

Research methods are computer vision methods, deep learning, neural networks, as well as optimization techniques such as quantization and pruning for adapting models to work on embedded systems with constrained resources.

The qualification work is devoted to researching image segmentation methods for autonomous vehicles, particularly the application of the DeepLabv3+ architecture to achieve high segmentation accuracy in real-time. The work also examines approaches to optimizing neural networks for efficient operation on embedded systems with limited computational resources.

The work provides a detailed analysis of existing approaches to image segmentation and substantiates the choice of DeepLabv3+ as the primary architecture. Significant attention is paid to training the model on synthetic and real datasets, such as Synscapes and Cityscapes, which allowed achieving high accuracy metrics. Furthermore, the study demonstrates the process of adapting models to real-world conditions through fine-tuning and the use of modern optimization techniques.

The results of the experiments show that the proposed approach enables accurate and high-performance object segmentation, even on embedded systems. This opens up opportunities for further application of such models in autonomous driving systems, enhancing their reliability and efficiency.

ЗМІСТ

	С.
Вступ	08
1 Аналіз предметної області та постановка задач дослідження	10
1.1 Сегментація зображень в реальному часі для автономних транспортних засобів	10
1.2 Методи сегментації зображень	10
1.3 Змістовна та формальна постановка задачі	11
1.4 Постановка задач дослідження	13
2 Вибір та обґрунтування методу розв’язання	15
2.1 Огляд можливих підходів до розв’язання задачі	15
2.2. Застосування архітектури DeepLab до сегментації зображень	19
2.3 Переваги обраної архітектури DeepLab	22
Висновки за розділом 2	25
3 Програмна реалізація	27
3.1 Python для вирішення задач сегментації зображень.....	27
3.2 Алгоритм та використання архітектури DeepLab для сегментації зображень у реальному часі	28
3.3 Опис програми	30
Висновки за розділом 3	32
4 Результати обчислювального експерименту та їх аналіз	33
4.1 Опис експерименту	33
4.2 Аналіз роботи сегментатора	36
Висновки за розділом 4	40
Висновки	41
Перелік джерел посилання	42
Додаток А Лістинг програми	45

ВСТУП

Актуальність теми. Актуальність роботи зумовлена зростаючим попитом на автономні транспортні засоби, які потребують високоефективних методів аналізу та розпізнавання об'єктів на дорозі для забезпечення безпеки пасажирів і учасників дорожнього руху. Сегментація зображень у реальному часі є ключовим елементом у створенні надійної системи комп'ютерного зору, яка здатна ідентифікувати різноманітні об'єкти та приймати рішення в умовах постійно змінюваного оточення.

Здатність точно сегментувати зображення дозволяє транспортним засобам ідентифікувати критично важливі об'єкти, такі як пішоходи, велосипедисти, дорожні знаки, смуги руху та інші транспортні засоби. Це забезпечує не лише безпечну навігацію, а й ефективне використання дорожньої інфраструктури. Особливої актуальності набуває задача сегментації в умовах низької видимості, наприклад, у темну пору доби, при поганих погодних умовах чи в складному міському середовищі, де густий трафік і численні об'єкти значно ускладнюють процес розпізнавання.

Окрім цього, автономні транспортні засоби стикаються з обмеженнями в обчислювальних ресурсах, що вимагає розробки моделей, здатних працювати в реальному часі, при цьому залишаючись досить легкими для інтеграції у вбудовані системи. Оптимізація моделей сегментації, зокрема через методи квантизації та скорочення параметрів, дозволяє досягати високої продуктивності, не жертвуючи точністю. Це має критичне значення для забезпечення надійності автономного керування в реальних умовах.

Таким чином, сегментація зображень виступає одним із базових компонентів у розвитку технологій автономного транспорту, що створює передумови для підвищення безпеки, зниження аварійності та оптимізації транспортних потоків.

Мета і завдання кваліфікаційної роботи. Метою кваліфікаційної роботи є розробка методу сегментації зображень для автономних транспортних за-

собів, який забезпечуватиме високу точність розпізнавання об'єктів на дорозі в умовах реального часу, зокрема при обмежених обчислювальних ресурсах. Для досягнення поставленої мети необхідно виконати наступні завдання:

- провести огляд і аналіз сучасного стану задачі «сегментації зображень в реальному часі для автономних транспортних засобів»;
- визначити об'єкт і предмет дослідження, провести аналіз сучасних методів сегментації зображень;
- вибрати оптимальну архітектуру нейронної мережі для сегментації зображень, враховуючи обмеженість обчислювальних ресурсів;
- оптимізувати обрану архітектуру та провести експериментальну валідацію результатів на тестових наборах даних.

Об'єктом дослідження є системи сегментації зображень для автономних транспортних засобів, що працюють в реальному часі.

Предметом дослідження є методи сегментації зображень, які дозволяють ідентифікувати та класифікувати різні об'єкти на дорозі, включаючи пішоходів, транспортні засоби, дорожню розмітку та інші елементи дорожньої сцени.

Методи дослідження. У кваліфікаційній роботі використовуються методи комп'ютерного зору, глибокого навчання, нейронні мережі, а також оптимізаційні техніки, такі як квантове навчання (Quantization) і скорочення з'єднань (Pruning) для адаптації моделей до роботи на вбудованих системах з обмеженими ресурсами.

Публікації. Результати, отримані у роботі, було представлено на III Міжнародній молодіжній науково-практичній конференції англійською мовою «Навчання і викладання: у світі після війни» (м. Харків, 8 листопада 2024 року) [1].

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧ ДОСЛІДЖЕННЯ

1.1 Сегментація зображень в реальному часі для автономних транспортних засобів

Сегментація зображення – це процес розділення зображення на окремі частини або сегменти з метою спрощення його аналізу та інтерпретації [2]. Вона використовується для визначення кордонів об'єктів та класифікації їх відповідно до характеристик. Існує декілька видів сегментації зображень, серед яких порогова сегментація, кластеризація, сегментація на основі глибоких нейронних мереж тощо [3].

Сегментація зображень в реальному часі є важливою складовою для автономних транспортних засобів (АТЗ), оскільки дозволяє їм ідентифікувати об'єкти на дорозі, такі як пішоходи, інші транспортні засоби, дорожня розмітка та перешкоди [4]. Це забезпечує безпеку як пасажирів, так і інших учасників дорожнього руху. Сегментація використовується для розпізнавання об'єктів та прийняття рішень в умовах постійно змінюваного оточення [5].

Основними проблемами сегментації зображень в реальному часі для АТЗ є необхідність забезпечення високої точності розпізнавання об'єктів при одночасно низькій затримці обробки зображень [6]. Багато сучасних підходів до сегментації мають обмеження у контексті роботи в реальному часі через високу обчислювальну складність [7].

1.2 Методи сегментації зображень

Для сегментації зображень використовуються різні методи, кожен з яких має свої переваги та недоліки.

Порогова сегментація є одним із найпростіших методів. Вона базується

на виділенні областей зображення на основі певного порогу значень яскравості [8]. Перевага цього методу полягає у простоті реалізації, але він є чутливим до змін освітлення та не підходить для складних сцен [9].

Кластеризація дозволяє виділити сегменти шляхом групування пікселів за схожими ознаками, такими як колір або текстура. Найчастіше використовується алгоритм K-means [10]. Основним недоліком кластеризації є залежність від попередньо заданої кількості кластерів та чутливість до початкових умов [11].

Глибокі нейронні мережі, такі як U-Net, SegNet та DeepLab, забезпечують високу точність сегментації за рахунок використання глибокого навчання [4]. U-Net використовує симетричну структуру з підйомними та спадними конволюційними шляхами, що дозволяє зберігати деталі зображення, але вимагає значних обчислювальних ресурсів [12]. SegNet добре підходить для семантичної сегментації і має меншу обчислювальну складність, що робить його придатним для вбудованих систем [5]. DeepLab використовує атріумні згортки для збереження контексту на різних масштабах зображення, що підвищує точність сегментації, але може викликати затримки в обробці [6].

1.3 Змістовна та формальна постановка задачі

Для досягнення сегментації в реальному часі використовуються оптимізовані згорткові нейронні мережі (Convolutional Neural Networks, CNN), які мають меншу обчислювальну складність та меншу кількість параметрів. Прикладами таких архітектур є ENet [11] та Fast-SCNN [12], які спеціально розроблені для швидкої сегментації.

Розглянемо структуру моделі [13, 14].

а) вхідний шар – зображення

$$I \in H \times W \times C,$$

де H – висота;

W – ширина;

C – кількість каналів;

б) згорткові шари з пониженням роздільної здатності. Вони зменшують просторові розміри для зменшення обчислювальної складності:

$$O_{i,j,k} = \sum_{m,n,l} K_{m,n,l,k} \cdot I_{i+m,j+n,l},$$

де I – вхідний тензор;

K – ядро згортки;

O – вихідний тензор після застосування згортки;

i, j – індекси по висоті та ширині вихідного тензора;

k – індекс вихідного каналу;

m, n – індекси по висоті та ширині ядра згортки;

l – індекс вихідного каналу;

в) Факторизація згорток – розбиття звичайних згорток на послідовність більш простих операцій для зменшення обчислень.

г) Групові та глибокі згортки використовуються для зменшення кількості параметрів:

$$O_{i,j,k} = \sum_{m,n} K_{m,n,k} \cdot I_{i+m,j+n,k}.$$

г) Декодер – шари розгортки або транспоновані згортки для відновлення просторової роздільної здатності.

Вихідний шар – карта сегментації:

$$S \in H \times W \times N,$$

де N – кількість класів.

Для навчання моделі використовується функція втрат, яка вимірює різницю між передбаченою картою сегментації та істинною розміткою [13, 14]:

$$L = \frac{-1}{N} \sum_{i=1}^N \sum_{c=1}^C w_c y_{i,c} \log(p_{i,c}),$$

де $y_{i,c}$ – істинна мітка;

$p_{i,c}$ – ймовірність передбачення класу c для пікселя i ;

w_c – вага для класу c .

Модель оптимізується за допомогою алгоритмів типу стохастичного градієнтного спуску (SGD) або його варіацій, таких як Adam [15]:

$$\theta_{t+1} = \theta_t - \eta \nabla_{\theta} L,$$

де θ_t – значення параметрів моделі на t -му кроці;

η – швидкість навчання (learning rate), яка визначає крок оновлення параметрів;

$\nabla_{\theta} L$ – градієнт функції втрат L за параметрами θ .

Цей підхід передбачає, що на кожному кроці ми рухаємося в напрямку зменшення значення функції втрат L , оновлюючи параметри моделі θ у протилежному напрямку до градієнта. Градієнт вказує, в якому напрямку функція L зростає найшвидше, тож віднімання цього значення дозволяє зменшувати втрати.

1.4 Постановка задач дослідження

Метою кваліфікаційної роботи є розробка методу сегментації зображень в реальному часі для автономних транспортних засобів, який забезпечуватиме високу точність розпізнавання об'єктів на дорозі в умовах обмежених обчислювальних ресурсів.

Для досягнення поставленої мети необхідно виконати наступні завдання:

- провести огляд і аналіз сучасного стану задачі «сегментації зображень в реальному часі для автономних транспортних засобів»;
- визначити об'єкт і предмет дослідження, провести аналіз сучасних методів сегментації зображень;
- вибрати оптимальну архітектуру нейронної мережі для сегментації зображень, враховуючи обмеженість обчислювальних ресурсів;
- оптимізувати обрану архітектуру та провести експериментальну валідацію результатів на тестових наборах даних.

2 ВИБІР ТА ОБҐРУНТУВАННЯ МЕТОДУ РОЗВ'ЯЗАННЯ

2.1 Огляд можливих підходів до розв'язання задачі

Проблема сегментації зображень для автономних транспортних засобів (АТЗ) є однією з ключових для забезпечення безпеки та надійності їх роботи. Сегментація дозволяє поділити зображення на значущі області, такі як дорога, транспортні засоби, пішоходи, дорожня розмітка тощо, що забезпечує основні дані для прийняття рішень системами керування АТЗ. У цьому розділі детально розглянуто основні методи сегментації зображень, які можуть використовуватися для розв'язання поставленої задачі, а також їх переваги та недоліки.

Першими методами, що використовувалися для сегментації зображень, були класичні методи, такі як порогова сегментація, сегментація на основі градієнтів та сегментація за допомогою кластеризації.

Порогова сегментація є найпростішим методом розділення зображення на сегменти. Основна ідея полягає у виборі порогу, який дозволяє виділити певні області на основі рівня інтенсивності пікселів. Цей метод є дуже ефективним у випадку, коли об'єкти мають добре виражену відмінність в інтенсивності від фону. Наприклад, порогова сегментація може бути використана для виявлення дорожньої розмітки на однорідному тлі дороги [16].

Проте порогова сегментація має суттєві обмеження. Вона не є стійкою до змін освітлення, що часто трапляється у реальних дорожніх умовах, коли яскравість сцени може суттєво змінюватися через погодні умови, тіні або відблиски. Через це порогова сегментація не є надійним вибором для сегментації складних сцен, які включають різні типи об'єктів, що мають схожу інтенсивність пікселів [10].

Іншим підходом є сегментація на основі градієнтів, яка полягає у виділенні країв об'єктів на зображенні. Для цього використовуються оператори, такі як Собеля, Робертса або Кенні, які дозволяють виявити місця зі значними змінами інтенсивності пікселів. Краї об'єктів визначають межі між різними облас-

тями зображення, що дає змогу виділити сегменти [17].

Основна перевага методів на основі градієнтів полягає у здатності виділяти чіткі контури об'єктів. Це робить такі методи корисними для виявлення об'єктів з добре вираженими краями, наприклад, транспортних засобів або дорожніх знаків. Проте ці методи також мають обмеження, особливо при роботі зі складними сценами, де об'єкти можуть мати розмиті або неоднозначні межі. Крім того, методи на основі градієнтів чутливі до шуму, що може призводити до помилок сегментації [18].

Кластеризація є іншим підходом до сегментації зображень, де пікселі групуються у кластери на основі їхніх характеристик, таких як колір або текстура. Один із найвідоміших методів кластеризації — алгоритм K-means. Цей алгоритм ділить пікселі на певну кількість кластерів, кожен з яких відповідає окремій частині зображення [19].

Основна перевага кластеризації полягає у можливості врахування різних характеристик зображення, таких як колір, текстура або яскравість, що робить цей метод досить універсальним. Проте алгоритм K-means вимагає попереднього визначення кількості кластерів, що може бути складно у випадку складних сцен. Крім того, цей метод може давати некоректні результати, якщо пікселі різних об'єктів мають схожі характеристики [20].

З розвитком глибокого навчання та нейронних мереж з'явилися сучасні методи сегментації, які демонструють значно кращі результати порівняно з класичними підходами. Найпоширеніші архітектури для сегментації зображень включають U-Net, SegNet та DeepLab.

U-Net є однією з найбільш відомих архітектур нейронних мереж для задач сегментації. Вона була розроблена для сегментації біомедичних зображень, проте згодом знайшла застосування в багатьох інших областях, включаючи автономні транспортні засоби [21].

Архітектура U-Net складається з двох основних частин: "спадного" шляху (енкодер) та "висхідного" шляху (декодер). Енкодер використовується для зменшення розміру вхідного зображення та виділення найважливіших характерис-

тик, тоді як декодер займається відновленням просторової роздільної здатності зображення. "Скіп-зв'язки" між енкодером і декодером дозволяють передавати інформацію з початкових шарів для кращого відновлення деталей, що забезпечує високу точність сегментації навіть для невеликих об'єктів [22].

Однак U-Net має високу обчислювальну складність, що обмежує її використання на вбудованих пристроях з обмеженими ресурсами, які часто використовуються в автономних транспортних засобах. Внаслідок цього оптимізація U-Net для використання в реальному часі є викликом, але її здатність забезпечувати точну сегментацію робить її привабливим варіантом для застосування в системах, де є доступ до потужних обчислювальних ресурсів.

SegNet є глибокою згортковою нейронною мережею, яка використовує енкодер-декодерну архітектуру для задач семантичної сегментації. У процесі енкодингу, зображення проходить через кілька згорткових шарів із поетапним зменшенням роздільної здатності, після чого декодер використовує індекси найважливіших пікселів, збережені під час операції субдискретизації, для відновлення оригінальної структури зображення [18].

Основною перевагою SegNet є її ефективність у використанні пам'яті, що робить її особливо придатною для застосування у вбудованих системах, де ресурси обмежені. Енкодер-декодерна структура дозволяє зменшити обчислювальні витрати, зберігаючи при цьому достатню точність для багатьох задач семантичної сегментації, таких як розпізнавання дорожньої розмітки, транспортних засобів і пішоходів [23].

Проте SegNet має і свої недоліки. Зокрема, вона іноді поступається іншим архітектурам, таким як DeepLab, у контексті сегментації складних сцен із великою кількістю об'єктів та високою щільністю інформації. В умовах реального часу, де кожна деталь має важливе значення для прийняття рішень, точність сегментації, яку забезпечує SegNet, може виявитися недостатньою.

DeepLab є однією з найсучасніших і найпотужніших архітектур для семантичної сегментації. Вона використовує атріумні згортки (dilated convolutions), які дозволяють розширити поле огляду без втрати роздільної здатності. Це осо-

бливо важливо для автономних транспортних засобів, які повинні враховувати контекст на різних масштабах для правильного розпізнавання об'єктів [24].

DeepLab також використовує повністю зв'язкові поля (Fully Connected Conditional Random Fields, CRFs) для підвищення точності сегментації, особливо на межах об'єктів. Такий підхід дозволяє зменшити помилки сегментації та підвищити точність виділення об'єктів із розмитими межами, що є типовою проблемою в умовах реального середовища [25].

Крім того, DeepLab підтримує можливість використання багаторівневих резолюцій, що дозволяє моделі враховувати різні масштаби та робить її особливо ефективною в умовах складних дорожніх сцен. Основним недоліком DeepLab є її висока обчислювальна складність. Для використання цієї архітектури в реальному часі потрібні потужні обчислювальні пристрої, або ж застосування методів оптимізації, таких як квантоване навчання (Quantization) або скорочення з'єднань (Pruning), що дозволяє зменшити складність моделі без значних втрат у точності [20, 21].

Усі розглянуті методи мають свої переваги та недоліки, і їхній вибір залежить від конкретної задачі та умов роботи. Класичні методи, такі як порогова сегментація та кластеризація, підходять для простих сцен із чітко вираженими об'єктами, але не є надійними в умовах змінного освітлення та складного оточення [16, 17]. Методи на основі градієнтів ефективні для виділення чітких контурів, але чутливі до шуму та не можуть забезпечити необхідної точності у складних дорожніх умовах [18].

Сучасні методи, такі як U-Net, SegNet та DeepLab, забезпечують значно кращу якість сегментації завдяки використанню глибоких нейронних мереж. U-Net забезпечує високу точність завдяки скіп-зв'язкам, але має високу обчислювальну складність, що може бути проблемою для використання в реальному часі [22]. SegNet є більш ефективним з точки зору використання пам'яті, але поступається точністю порівняно з DeepLab [9, 10]. DeepLab, у свою чергу, пропонує найкращий баланс між точністю та можливістю врахування контексту, що є критично важливим для задач автономного водіння [4, 25].

2.2. Застосування архітектури DeepLab до сегментації зображень

Архітектура DeepLab є однією з найефективніших і найпоширеніших для задач сегментації зображень, особливо в контексті автономних транспортних засобів (АТЗ). Її здатність забезпечувати високу точність сегментації в поєднанні з можливістю працювати в реальному часі робить її одним із найкращих варіантів для використання в системах комп'ютерного зору [1]. Розглянемо основні особливості архітектури DeepLab, способи її адаптації для роботи в реальному часі, а також порівняння з іншими підходами до сегментації.

Архітектура DeepLab використовує унікальну комбінацію методів, які роблять її особливо ефективною для сегментації зображень. Серед основних особливостей DeepLab є використання атріумних згорток (dilated convolutions), що дозволяє зберігати високу роздільну здатність під час згортки, тим самим забезпечуючи контекстуальне розуміння зображення на різних масштабах [4]. Це дозволяє моделі ефективно сегментувати об'єкти, зберігаючи інформацію про їхні межі.

Крім того, DeepLab використовує повністю зв'язкові поля (Fully Connected Conditional Random Fields, CRFs), які дозволяють покращити точність сегментації, зменшуючи помилки на межах об'єктів. Ця техніка особливо корисна для обробки складних дорожніх сцен, де чіткість меж є вирішальною для прийняття рішень системою автономного водіння [4]. Використання CRFs також дозволяє зменшити розмитість об'єктів та виділити деталі, які можуть бути втрачені через інші підходи.

Ще однією важливою особливістю є можливість використання декількох рівнів роздільності. Це досягається за допомогою використання різних фільтрів для атріумних згорток, що дозволяє моделі працювати з різними масштабами зображення одночасно. Такий підхід дозволяє враховувати і глобальний контекст, і дрібні деталі, що є критично важливим для успішного розпізнавання об'єктів у дорожньому середовищі [4].

Незважаючи на свою високу ефективність, архітектура DeepLab має ви-

соку обчислювальну складність, що може стати перешкодою для використання в реальному часі на обмежених апаратних ресурсах, наприклад, у мобільних процесорах або вбудованих системах. Для адаптації DeepLab до роботи в реальному часі застосовуються різні методи оптимізації, такі як Quantization та Pruning [7, 8].

Quantization дозволяє зменшити розрядність чисел, що використовуються в обчисленнях, наприклад, з 32-бітних до 8-бітних, що суттєво знижує потреби у пам'яті та збільшує швидкість обчислень без значної втрати точності [7]. Ця техніка є особливо корисною для вбудованих систем, де ресурси є обмеженими, але потрібна висока продуктивність.

Pruning полягає у видаленні малозначущих зв'язків у нейронній мережі, що дозволяє зменшити кількість параметрів і, відповідно, обчислювальну складність моделі [8]. Це дає змогу використовувати DeepLab у пристроях з низькою енергоспоживаністю, таких як FPGA або спеціалізовані процесори для обробки зображень. Дослідження показали, що за правильної стратегії Pruning можна зберегти основну частину точності моделі при значному зменшенні кількості параметрів [8].

Крім того, існують інші методи оптимізації, такі як використання більш легких архітектур, наприклад, MobileNets або ENet, які можуть бути інтегровані в DeepLab для зменшення обчислювальних вимог [11, 13]. Такі підходи дозволяють зберігати основні переваги DeepLab, зокрема її здатність працювати з різними рівнями контексту, при цьому значно зменшуючи потреби у ресурсах.

Використання DeepLab у системах автономного водіння має важливе значення завдяки її здатності ефективно сегментувати складні дорожні сцени. Для автономних транспортних засобів критично важливо мати точне уявлення про навколишнє середовище, щоб приймати правильні рішення в режимі реального часу. DeepLab дозволяє сегментувати такі об'єкти, як інші транспортні засоби, пішоходи, дорожні знаки, розмітку та перешкоди, що є важливим для безпечного пересування [4].

Крім того, для підвищення ефективності систем автономного водіння,

DeepLab може бути інтегрована з іншими технологіями, такими як Sensor Fusion, де дані з різних сенсорів (камери, лідари, радари) комбінуються для отримання більш повної картини навколишнього середовища. Завдяки цьому можна досягти більшої точності розпізнавання об'єктів та їхнього правильного позиціонування у просторі [4].

У порівнянні з іншими архітектурами, такими як SegNet та U-Net, DeepLab має ряд переваг, які роблять її більш придатною для використання в автономних транспортних засобах. SegNet, наприклад, є ефективнішою з точки зору використання пам'яті, але поступається DeepLab у точності, особливо при роботі зі складними сценами [5]. U-Net забезпечує високу точність завдяки своїй структурі зі скіп-зв'язками, але має вищу обчислювальну складність, що робить її менш придатною для реального часу без додаткової оптимізації [21].

DeepLab, завдяки використанню атріумних згорток та CRFs, може ефективніше працювати зі складними дорожніми сценами, зберігаючи інформацію про контекст об'єктів на різних масштабах. Це робить її особливо цінною для автономних транспортних засобів, де важливо не лише правильно розпізнати об'єкт, але й зрозуміти його взаємозв'язок з іншими об'єктами на дорозі [4].

Крім того, у порівнянні з більш легкими архітектурами, такими як ENet або Fast-SCNN, DeepLab забезпечує кращу точність, хоча й має вищі вимоги до ресурсів [11, 12]. Для автономних транспортних засобів, де безпека є ключовим фактором, точність має пріоритет над обчислювальною ефективністю, що робить DeepLab кращим вибором, особливо при використанні методів оптимізації.

Одним із основних викликів при використанні DeepLab є її інтеграція в системи з обмеженими ресурсами, зокрема вбудовані системи автономних транспортних засобів. Хоча методи, такі як Quantization та Pruning, допомагають зменшити складність моделі, необхідність збереження високої точності ставить додаткові вимоги до апаратної частини [7, 8]. У майбутньому, можливим напрямком розвитку є створення спеціалізованих апаратних прискорювачів, які дозволять ефективно виконувати обчислення, пов'язані з атріумними згортками

та CRFs.

Ще одним перспективним напрямком є інтеграція технологій уваги (attention mechanisms) в архітектуру DeepLab для підвищення її здатності фокусуватися на важливих частинах зображення. Це дозволить не лише зменшити обчислювальні витрати, але й підвищити точність сегментації за рахунок більш ефективного використання ресурсів [16]. Також активно розробляються моделі, які поєднують в собі переваги кількох архітектур, наприклад, DeepLab з використанням MobileNets для обробки на мобільних пристроях [13].

Таким чином, архітектура DeepLab залишається однією з найпотужніших для задач сегментації зображень, особливо в контексті автономного водіння. Її здатність забезпечувати високу точність сегментації в умовах реального часу, використовуючи сучасні методи оптимізації, робить її привабливим вибором для розробки безпечних та надійних систем автономного транспорту.

2.3. Переваги обраної архітектури DeepLab

Архітектура DeepLab має ряд унікальних переваг, які роблять її надзвичайно ефективною для задач сегментації зображень, особливо в контексті автономних транспортних засобів.

Одна з основних переваг DeepLab полягає в її використанні атріумних згорток, які дозволяють зберігати високу роздільну здатність зображення під час згортки, тим самим забезпечуючи кращий контекстуальний аналіз [4]. Застосування атріумних згорток дозволяє моделі працювати з різними масштабами об'єктів на одному зображенні, зберігаючи інформацію про деталі без необхідності збільшення обчислювальних витрат. Це є особливо важливим для задач сегментації у дорожньому середовищі, де об'єкти, такі як пішоходи, транспортні засоби та дорожні знаки, можуть бути різних розмірів і знаходитися на різних відстанях від камери.

Іншою ключовою перевагою DeepLab є використання повністю зв'язко-

вих умовних випадкових полів (CRFs), що дозволяють підвищити точність сегментації на межах об'єктів. Ця техніка є особливо корисною для зменшення помилок на межах об'єктів, що робить сегментацію більш точною та дозволяє уникнути розмитості [4]. CRFs допомагають підвищити чіткість розпізнавання об'єктів, що особливо важливо для забезпечення безпеки в автономних транспортних засобах, де помилки сегментації можуть призвести до неправильних рішень.

DeerLab також має можливість працювати з різними рівнями резолюції, що дозволяє враховувати як глобальний контекст, так і дрібні деталі зображення [4]. Це забезпечується за допомогою атріумних згорток з різними коефіцієнтами, що дозволяють охоплювати об'єкти різних розмірів і надавати моделі здатність працювати з різними масштабами. Такий підхід значно підвищує точність сегментації, оскільки модель здатна одночасно аналізувати як великі об'єкти, так і невеликі деталі.

Хоча DeerLab має високу обчислювальну складність, існують численні методи, які дозволяють оптимізувати її для роботи в реальному часі. Наприклад, методи Quantization та Pruning дозволяють значно зменшити кількість обчислень і обсяг пам'яті, необхідний для роботи моделі, зберігаючи при цьому високу точність [7, 8]. Quantization знижує розрядність чисел, що використовуються у моделі, що дозволяє використовувати менш потужне обладнання без втрати якості сегментації. Pruning, у свою чергу, дозволяє видалити малозначущі зв'язки у нейронній мережі, що також сприяє зменшенню обчислювальних витрат і забезпечує можливість роботи моделі на мобільних пристроях або вбудованих системах [8].

Ще однією перевагою DeerLab є можливість інтеграції з більш легкими архітектурами, такими як MobileNets або ENet, що дозволяє зменшити обчислювальні вимоги без значної втрати точності [11, 13]. Це забезпечує можливість використання DeerLab у пристроях з обмеженими ресурсами, таких як мобільні процесори або спеціалізовані апаратні прискорювачі. Завдяки такій інтеграції, DeerLab може забезпечувати високу продуктивність і точність навіть на обме-

женому апаратному забезпеченні, що робить її універсальним рішенням для різних застосувань.

Іншим перспективним напрямком розвитку DeepLab є застосування механізмів уваги (attention mechanisms), які дозволяють моделі фокусуватися на найбільш важливих частинах зображення, що підвищує її ефективність та точність [26]. Механізми уваги дозволяють виділити найбільш значущі частини зображення та зосередити на них обчислювальні ресурси, що сприяє підвищенню точності сегментації та зменшенню обчислювальних витрат. Це особливо важливо для задач сегментації у реальному часі, де важлива швидкість прийняття рішень.

DeepLab забезпечує підтримку глибокого контекстуального аналізу завдяки використанню атріумних згорток і CRFs, що дозволяє враховувати як локальні, так і глобальні особливості зображення [4]. Це робить DeepLab особливо ефективною для складних сценаріїв, таких як сегментація дорожніх сцен, де важливо правильно ідентифікувати об'єкти та їхній контекст у просторі. Завдяки такому підходу модель здатна враховувати інформацію про оточення кожного об'єкта, що дозволяє приймати більш точні та безпечні рішення в автономних транспортних засобах.

У порівнянні з іншими популярними методами, такими як Mask R-CNN, SegNet та Fully Convolutional Networks (FCN), DeepLab демонструє значні переваги в контексті точності та здатності працювати зі складними сценами [3, 5, 6]. Mask R-CNN є ефективним для розпізнавання та сегментації об'єктів, але його складність робить його менш придатним для використання у реальному часі [3]. SegNet є більш економічною з точки зору використання пам'яті, але поступається DeepLab у точності, особливо при сегментації складних дорожніх сцен [5]. FCN є більш універсальною архітектурою, але не має таких оптимізацій, які забезпечують DeepLab високу точність та здатність до глибокого контекстуального аналізу [6].

DeepLab також підтримує використання сучасних методів оптимізації, таких як Adam, які дозволяють забезпечити швидку та ефективну адаптацію па-

раметрів моделі під час навчання [15]. Використання Adam сприяє швидшому збіженню моделі, що дозволяє значно зменшити час на її навчання без втрати точності. Це є важливою перевагою для реалізації моделей сегментації, які потребують швидкого навчання та перенавчання.

Таким чином, архітектура DeepLab має ряд унікальних переваг, які роблять її одним із найкращих варіантів для задач сегментації зображень, зокрема для автономних транспортних засобів. Використання атріумних згорток, CRFs, можливість роботи з різними рівнями резолюції, а також інтеграція з легкими архітектурами та механізмами уваги забезпечують високу точність та ефективність DeepLab навіть у складних умовах. Це робить її незамінною для розробки систем комп'ютерного зору, які потребують точності та швидкості в реальному часі.

Висновки за розділом 2

В розділі було проаналізовано методи сегментації зображень та виявлено, що класичні підходи, такі як порогова сегментація, градієнтні методи та класифікація, мають обмеження, які не дозволяють ефективно застосовувати їх у складних умовах автономного водіння. Ці методи можуть бути корисними для простих сцен із чіткими контурами, але їх чутливість до шуму, змін освітлення та складності сцен обмежує їх застосування в реальному середовищі.

Сучасні підходи, засновані на глибоких нейронних мережах, такі як U-Net, SegNet і DeepLab, значно перевершують класичні методи за точністю та гнучкістю. Архітектура U-Net забезпечує високу точність завдяки використанню скіп-зв'язків, але має високу обчислювальну складність. SegNet пропонує оптимальний баланс між ефективністю використання пам'яті та точністю, але поступається іншим методам при сегментації складних сцен. DeepLab є найбільш перспективним рішенням, що поєднує використання атріумних згорток для контекстуального аналізу та CRF для поліпшення сегментації меж об'єктів.

Вибір DeepLab як основної архітектури для задачі сегментації зображень обґрунтовано її здатністю забезпечувати високу точність і адаптивність до складних дорожніх сцен. Підтримка багаторівневого аналізу, ефективна обробка контексту та можливість оптимізації моделі для роботи в реальному часі роблять її найкращим вибором для автономних транспортних засобів.

Таким чином, методи, що базуються на глибокому навчанні, значно розширюють можливості систем сегментації, забезпечуючи високу надійність, точність і ефективність в умовах реального часу. Обраний підхід дозволяє вирішувати ключові проблеми автономного водіння, пов'язані з аналізом складних середовищ і забезпеченням безпеки учасників дорожнього руху.

3 ПРОГРАМНА РЕАЛІЗАЦІЯ

3.1 Python для вирішення задач сегментації зображень

Вибір мови програмування є ключовим етапом у розробці системи сегментації зображень у реальному часі для автономних транспортних засобів. Python був обраний через його численні переваги, які включають простоту використання, широку підтримку бібліотек і активну спільноту розробників. Python є мовою високого рівня, що дозволяє швидко створювати прототипи, а його читабельний синтаксис сприяє легкому внесенню змін і підтримці коду.

Однією з основних причин вибору Python є його широка екосистема бібліотек для машинного навчання та обробки зображень. Бібліотека Torch (і її підпакек torchvision) надає інструменти для створення, навчання та використання глибоких нейронних мереж. Torch забезпечує доступ до попередньо натренованих моделей, таких як DeepLab, які можна легко адаптувати до конкретних завдань сегментації зображень. Torchvision містить зручні засоби для обробки зображень, включаючи завантаження популярних наборів даних, таких як Cityscapes, і виконання операцій, таких як нормалізація, перетворення формату і масштабування.

Інтеграція Python з іншими бібліотеками, такими як OpenCV, робить можливим ефективну обробку відеопотоків у реальному часі. OpenCV забезпечує функції для захоплення кадрів, попередньої обробки зображень і візуалізації результатів сегментації. Бібліотека NumPy використовується для роботи з багатовимірними масивами і виконання математичних операцій, необхідних для підготовки даних.

Python підтримує різноманітні фреймворки для розподілених обчислень, що є важливим для роботи з великими обсягами даних і підвищення продуктивності. Наприклад, за допомогою бібліотеки Dask можна обробляти дані паралельно, використовуючи кілька процесорів або графічних процесорів.

Додатковою перевагою Python є його сумісність з різними платформами

та апаратними прискорювачами, такими як NVIDIA CUDA. Це дозволяє виконувати оптимізовані обчислення на GPU, що критично важливо для забезпечення реального часу роботи системи сегментації. Інструменти, такі як TensorRT, інтегруються з Python і дозволяють додатково оптимізувати моделі для швидкого виконання на вбудованих пристроях.

Загалом, Python є оптимальним вибором для задач сегментації зображень завдяки його зручності у використанні, потужним інструментам для глибокого навчання і широкій підтримці інфраструктури для реального часу. Це дозволяє значно скоротити час розробки, забезпечуючи при цьому високу продуктивність і гнучкість системи.

3.2 Алгоритм та використання архітектури DeepLab для сегментації зображень у реальному часі

Архітектура DeepLab є однією з найпотужніших і широко використовуваних моделей для сегментації зображень. Її основні переваги – це здатність обробляти складні структури зображень, гнучкість у налаштуванні та підтримка роботи з різними типами даних. У цьому розділі буде розглянуто, як DeepLab використовується для вирішення задач сегментації у реальному часі, основні труднощі, які виникають під час навчання моделі, а також процес адаптації нейромережі до даних із реального світу.

DeepLab побудовано на основі глибоких згорткових нейронних мереж. DeepLab використовує кілька ключових інновацій, які роблять її ефективною для сегментації зображень. Однією з таких інновацій є дільнична згортка (Atrous Convolution), яка дозволяє збільшувати поле огляду моделі без збільшення кількості параметрів або зменшення роздільної здатності просторової карти ознак. Це особливо корисно для виявлення великих об'єктів або контекстів на зображенні [25]. Вдосконалена версія DeepLabV3+ включає механізми "encoder-decoder" для кращого відновлення деталей зображень на межах

об'єктів [4]. Крім того, пов'ністю зв'язані умовні випадкові поля (Fully Connected CRFs) використовуються для уточнення меж сегментації, роблячи їх чіткішими. DeepLab може використовувати різні моделі, такі як ResNet чи Xception, у якості основи для витягання ознак [14].

Навчання нейромережі починається з вибору відповідних наборів даних. Cityscapes є одним із основних наборів даних для сегментації міських сцен. Він включає 5000 анотованих зображень високої роздільної здатності, які охоплюють реальні міські локації з багатою різноманітністю об'єктів, таких як дороги, транспортні засоби, пішоходи та будівлі [26]. Особливостями Cityscapes є детальні анотації на рівні пікселів та реальні сцени з високою варіативністю освітлення та погоди, що робить його ідеальним для фінального налаштування моделі. Іншим важливим набором є Synscapes – синтетичний набір даних, створений для імітації міських сцен. Він забезпечує більший обсяг і варіативність з меншими витратами на анотування [26]. Фотореалістичні зображення Synscapes, створені з використанням графічних рушіїв, надають чіткі та повні анотації, які легко масштабуються, тому використовуються для початкового навчання моделі.

Процес навчання зазвичай починається з попереднього навчання на синтетичних даних, таких як Synscapes. Це дозволяє моделі навчитися розрізняти базові об'єкти, такі як дороги, транспортні засоби та пішоходи, у стандартизованих умовах. Головна перевага цього етапу — великий обсяг даних, який сприяє уникненню перенавчання. Наступним етапом є тонке налаштування (Fine-Tuning) на реальних даних, таких як Cityscapes. Завдяки високій якості та реалістичності цих даних, модель адаптується до варіативності реального світу. Процес включає зниження швидкості навчання (learning rate) для запобігання розриву попередньо вивчених шаблонів. Аугментація даних, яка включає горизонтальне відображення, обертання, зміну яскравості та контрасту, дозволяє покращити генералізацію моделі [13]. Для оптимізації процесу використовуються NVIDIA CUDA та TensorRT, що суттєво зменшує час обробки одного кадру, що є критично важливим для роботи систем у реальному часі.

Різниця між синтетичними та реальними даними є однією з основних труднощів. Синтетичні зображення часто мають іншу текстуру та освітлення порівняно з реальними сценами, що може призводити до погіршення точності моделі після перенесення на реальні дані. Обмеження обчислювальних ресурсів також є серйозним викликом, оскільки реалізація системи у реальному часі вимагає оптимізації моделі для виконання на вбудованих пристроях з обмеженою потужністю. Крім того, об'єкти з тонкими структурами, наприклад, дорожні знаки або велосипедні спиці, можуть бути погано сегментовані через низьку роздільну здатність ознак.

DeepLab інтегрується з фреймворками, такими як TensorRT, що дозволяє досягати продуктивності до 30 FPS на сучасних GPU. Для обробки відеопотоків використовується OpenCV, який забезпечує ефективне захоплення кадрів та візуалізацію результатів сегментації. Використання DeepLab у поєднанні з оптимізацією на апаратному рівні дозволяє створювати системи сегментації, які є точними, швидкими та придатними для автономних транспортних засобів.

3.3 Опис програми

Програма для сегментації зображень реалізована у двох основних файлах: `train.py` та `realtime_seg.py`. Файл `train.py` відповідає за процес навчання моделі, включаючи обробку даних, визначення архітектури моделі, навчання та валідацію. Вхідні дані складаються з двох основних наборів: `Synscapes` для початкового навчання та `Cityscapes` для тонкого налаштування моделі.

У файлі `train.py` реалізовано класи для роботи з наборами даних `Synscapes` і `Cityscapes`. Ці класи включають функції завантаження, попередньої обробки та аугментації зображень. Особлива увага приділена сумісності між анотаціями наборів даних, що забезпечується через функцію мапінгу, яка переводить мітки `Synscapes` у формат, сумісний із `Cityscapes`. Процес навчання моделі організовано у два етапи: початкове навчання на синтетичних даних `Synscapes` і тонке

налаштування на реальних даних Cityscapes. Для оптимізації навчання використовується попередньо натренована архітектура DeepLabv3+ із модифікованим вихідним шаром для 19 класів.

Процес валідації реалізовано з використанням кількох важливих метрик. mIoU (mean Intersection over Union) є ключовою метрикою для задач сегментації, оскільки вона показує середню частку перетину між передбаченими і реальними сегментами для всіх класів. Високе значення mIoU свідчить про високу якість сегментації. F1-score оцінює баланс між точністю (precision) та повнотою (recall) для кожного класу, що особливо важливо у випадку класів з нерівномірним розподілом. Pixel accuracy вимірює частку правильно передбачених пікселів у загальному обсязі, даючи загальну оцінку якості моделі. Використання цих метрик дозволяє комплексно оцінити ефективність моделі та виявити слабкі місця.

Файл `realtime_seg.py` забезпечує виконання сегментації у реальному часі. Основна функціональність програми включає завантаження попередньо натренованої моделі DeepLabv3, обробку відеопотоку у режимі реального часу та візуалізацію результатів. Для кожного кадру відео проводиться сегментація, результат якої перетворюється у кольорову маску для візуалізації. Розмір кадру та пристрій для обчислень можуть бути налаштовані за допомогою параметрів командного рядка.

У програмі передбачено використання мапи кольорів для 19 класів Cityscapes, що полегшує інтерпретацію результатів сегментації. Крім того, реалізовано відображення кількості кадрів за секунду (FPS), що дозволяє оцінити продуктивність системи.

Програма використовує OpenCV для обробки відео та PyTorch для виконання обчислень, забезпечуючи високу швидкість і точність. У разі потреби програму можна використовувати для аналізу відео з будь-якого джерела, наприклад, камер або файлів.

Висновки за розділом 3

Розділ демонструє практичну реалізацію системи сегментації зображень для автономних транспортних засобів на основі архітектури DeepLab. Обґрунтування вибору мови Python підкреслює її універсальність, простоту використання та потужну екосистему бібліотек, таких як PyTorch, OpenCV та TensorRT. Завдяки цим інструментам вдалося створити програму, яка ефективно вирішує задачу сегментації у реальному часі.

Успіх навчання моделі значною мірою забезпечено завдяки інтеграції синтетичних (Synscapes) та реальних (Cityscapes) наборів даних. Початкове навчання на синтетичних даних дало змогу підготувати модель до основних завдань сегментації, а тонке налаштування на реальних даних адаптувало її до складних дорожніх умов. Додатково було застосовано різноманітні методи аугментації даних, що сприяло генералізації моделі.

Особливістю програмної реалізації є розділення на модулі для навчання (`train.py`) та виконання сегментації у реальному часі (`realtime_seg.py`). Такий підхід забезпечує зручність використання та масштабованість. Алгоритми оптимізовані для роботи на GPU за допомогою NVIDIA CUDA та TensorRT, що дозволяє досягати продуктивності, необхідної для інтеграції у вбудовані системи.

Метрики, такі як mIoU, F1-score та pixel accuracy, не лише оцінюють якість моделі, а й забезпечують детальне розуміння її поведінки на різних етапах навчання. Високі значення цих показників підтверджують ефективність розробленої системи та її здатність працювати в умовах реального світу.

Таким чином, представлена програмна реалізація демонструє, як сучасні архітектури нейронних мереж можуть бути адаптовані для вирішення складних завдань комп'ютерного зору в автономних транспортних засобах. Використання Python, гнучких бібліотек та апаратної оптимізації робить цю систему практично придатною для широкого спектра застосувань у реальному часі.

4 РЕЗУЛЬТАТИ ОБЧИСЛЮВАЛЬНОГО ЕКСПЕРИМЕНТУ ТА ЇХ АНАЛІЗ

4.1 Опис експерименту

Метою обчислювального експерименту було оцінити ефективність архітектури DeepLabv3+ для задач сегментації зображень у реальному часі. Для цього експеримент було розділено на два основні етапи: попереднє навчання на синтетичних даних і тонке налаштування на реальних зображеннях. Кожен етап мав свої особливості, пов'язані з характеристиками даних, методами обробки та оцінки.

На першому етапі використовувався набір даних Synscapes, який забезпечує високоякісні синтетичні зображення міських сцен. Цей набір був обраний через його чітку структурованість та відповідність основним завданням сегментації. Зображення містять анотації для 19 класів, серед яких дороги, будівлі, транспортні засоби, рослинність тощо. Метою цього етапу було навчити модель розпізнавати основні об'єкти в стандартизованих умовах.

Для підвищення якості навчання застосовувались різноманітні техніки аугментації, зокрема випадкове масштабування, горизонтальне відображення та корекція кольорів. Ці методи дозволили збільшити варіативність даних і зробити модель стійкішою до змін умов. Навчання здійснювалось з використанням оптимізатора Adam, який забезпечує ефективне оновлення ваг мережі, та функції втрат CrossEntropyLoss, адаптованої для багатокласової сегментації. Результати роботи моделі на наборі даних Synscapes представлені на рис. 4.1.

Другий етап включав тонке налаштування моделі на реальних даних Cityscapes. Цей набір даних є значно складнішим через текстурну і освітлювальну різноманітність, а також більшу кількість об'єктів і деталей. Реальні зображення були анотовані аналогічно до Synscapes, що дозволило без додаткових змін використовувати їх для навчання. Основною метою цього етапу було адаптувати модель до умов реального світу, забезпечивши її узагальнюваність.

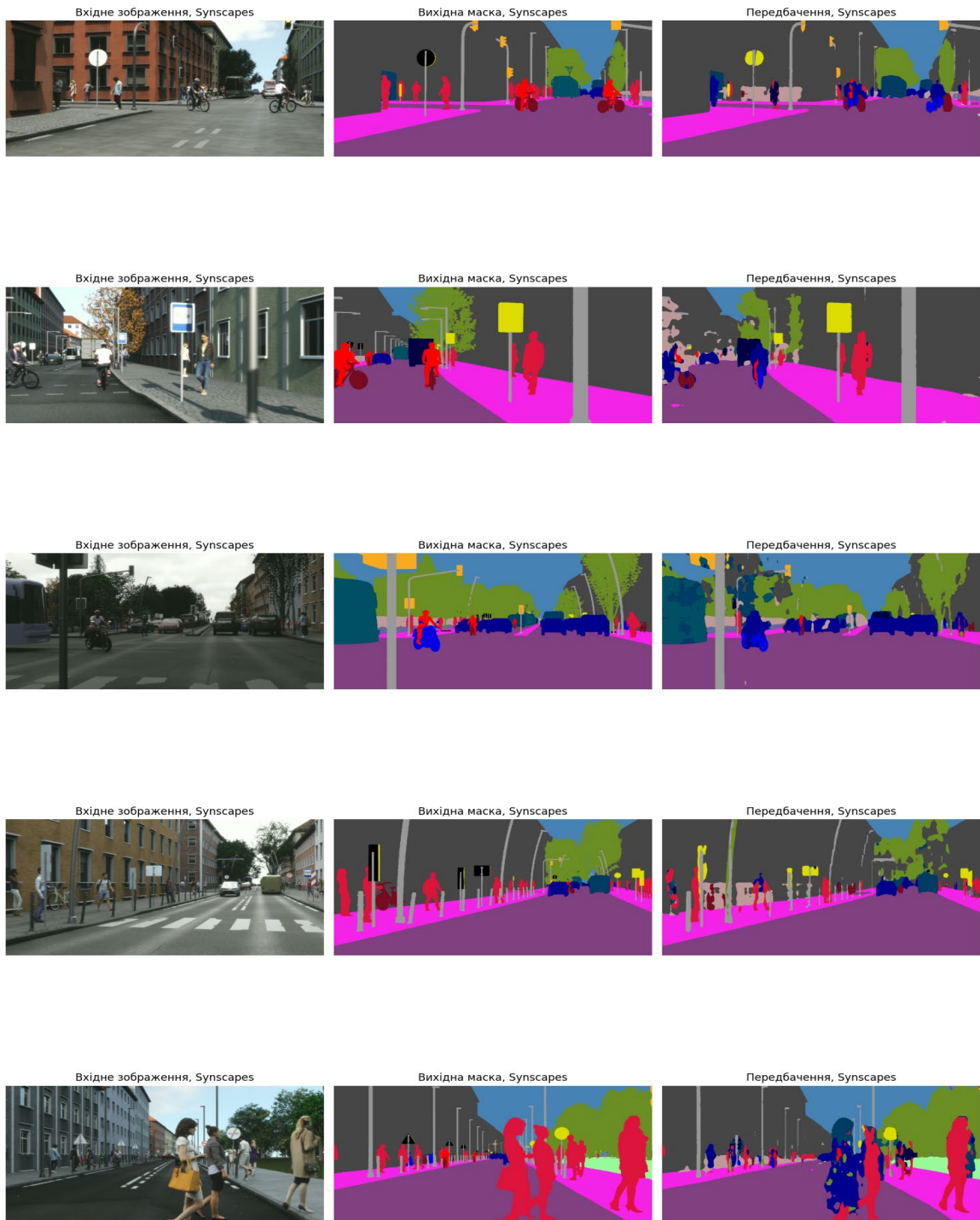


Рисунок 4.1 – Результати роботи моделі на наборі даних Synscapes

Для тонкого налаштування було зменшено швидкість навчання, що дозволило уникнути перенавчання та забезпечити поступове вдосконалення параметрів. Результати роботи моделі на наборі даних Cityscapes представлені на рис. 4.2.

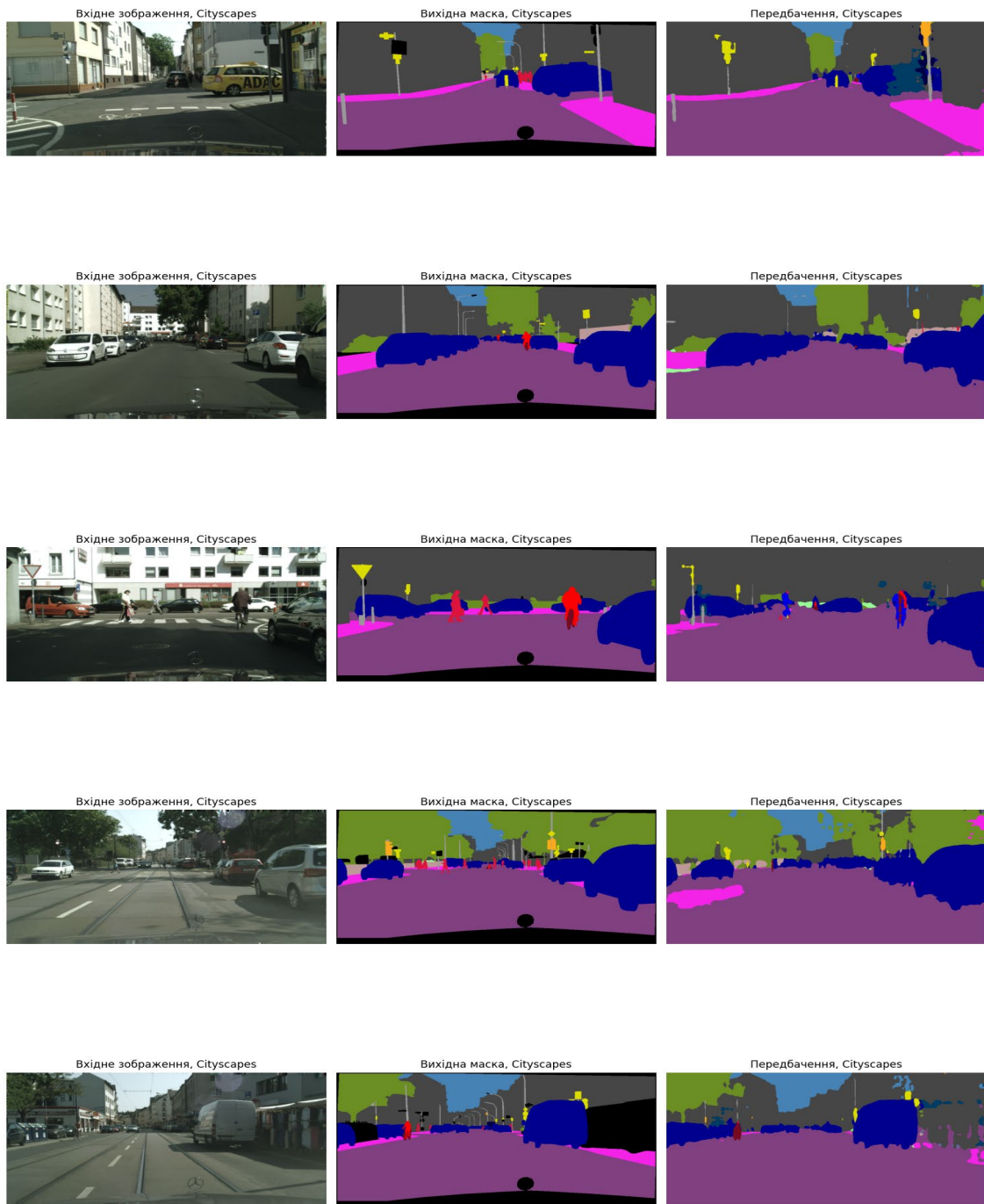


Рисунок 4.2 – Результати роботи моделі на наборі даних Cityscapes

Оцінка моделі проводилась на основі валідаційних підмножин для обох наборів даних. Основними метриками виступали mean Intersection over Union (mIoU), Precision, Recall, F1-score і Pixel Accuracy. Кожна з цих метрик дозволяла оцінити специфічні аспекти роботи моделі, зокрема точність передбачень, їх повноту та загальну якість сегментації.

На кожному етапі були отримані детальні результати, які відображали

прогрес навчання. Вони включали графіки залежності функції втрат і метрик від кількості епох, числові показники для різних класів об'єктів та візуалізації результатів сегментації. Ці результати дозволили не лише оцінити ефективність моделі, але й виявити області для подальшого вдосконалення.

4.2 Аналіз роботи сегментатора

Результати роботи сегментатора оцінювались за допомогою кількох ключових метрик: середня частка перетину (mean Intersection over Union, mIoU), точність (Precision), повнота (Recall), F1-score, точність пікселів (Pixel Accuracy). Ці метрики дозволяють комплексно оцінити якість сегментації, враховуючи як загальну відповідність передбачених областей реальним, так і точність для кожного класу.

Mean Intersection over Union (mIoU) обчислюється за формулою [4, 11]:

$$mIoU = \frac{1}{N} \sum_{i=1}^N \frac{TP_i}{TP_i + FP_i + FN_i},$$

де TP_i – кількість пікселів, правильно класифікованих як належні до класу i ;

FP_i – кількість пікселів, помилково віднесених до класу i ;

FN_i – кількість пікселів, які належать до класу i , але були пропущені;

N – кількість класів.

Ця метрика є основною для задач сегментації, оскільки вона враховує як позитивні, так і негативні передбачення для кожного класу, даючи узагальнену оцінку якості сегментації [4, 11].

Precision (Точність) обчислюється за формулою [21]:

$$\text{Precision} = \frac{TP}{TP + FP},$$

де TP – кількість істинно позитивних передбачень;

FP – кількість помилкових позитивних передбачень.

Точність показує, яка частка передбачених пікселів дійсно належить до відповідного класу [21].

Recall (Повнота) обчислюється за формулою [21]:

$$\text{Recall} = \frac{TP}{TP + FN},$$

де TP – кількість істинно позитивних передбачень;

FN – кількість пропущених позитивних передбачень.

Повнота вимірює, наскільки добре модель знаходить усі пікселі, які належать до класу [21].

F1-Score обчислюється за формулою [21]:

$$F1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}.$$

F1-score об'єднує Precision і Recall у єдину метрику, що корисно, коли необхідно оцінити баланс між ними [21].

Pixel Accuracy (Точність Пікселів) обчислюється за формулою [5]:

$$\text{Pixel Accuracy} = \frac{\sum_{i=1}^N TP_i}{\sum_{i=1}^N (TP_i + FP_i + FN_i)}.$$

Точність пікселів обчислює частку пікселів, які були правильно класифіковані, серед усіх пікселів [5].

Отримані оцінки метрик для роботи сегментатора на наборах даних Synscapes та Cityscapes представлені в таблиці 4.1. На рис. 4.3, 4.4 представлені

графіки оцінки точності сегментації на Synscapes та Cityscapes відповідно.

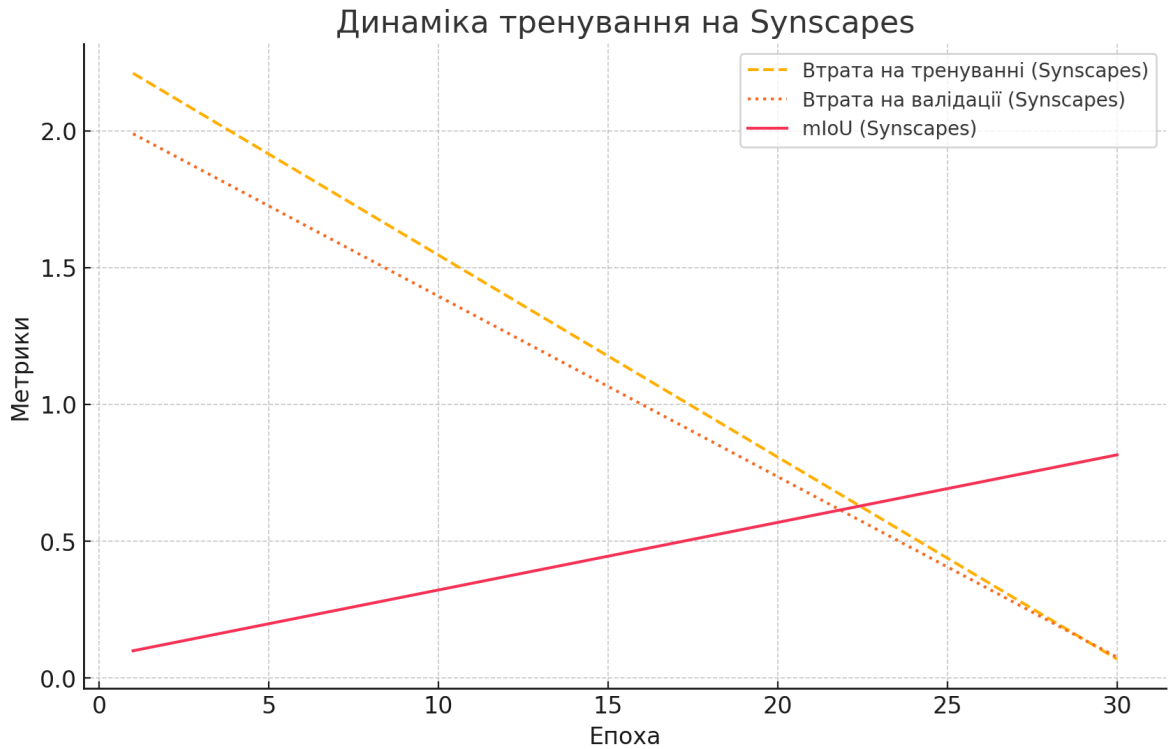


Рисунок 4.3 – Графік оцінки точності сегментації на Synscapes



Рисунок 4.4 – Графік оцінки точності сегментації на Cityscapes

Таблиця 4.1 – Оцінки метрик для роботи сегментатора

Метрика	Synscapes	Cityscapes
Train Loss	0,0694	0,0958
Val Loss	0,0766	0,1417
mIoU	0,8158	0,5781
Precision	0,9414	0,8362
Recall	0,9342	0,8023
F1-score	0,8826	0,6491
Pixel Accuracy	0,9707	0,9472

На першому етапі (Synscapes) спостерігалось поступове зниження функції втрат (Train Loss та Val Loss), що свідчить про ефективність навчання. Високі значення mIoU та F1-score підтверджують, що модель успішно навчилася розпізнавати основні класи об'єктів у стандартизованих умовах.

Другий етап (Cityscapes) показав складність адаптації моделі до реальних даних. Збільшення Val Loss та зниження mIoU свідчать про виклики, пов'язані з різноманіттям сцен, освітлення та текстур. Це підкреслює важливість застосування додаткових технік, таких як аугментація даних та регуляризація, для покращення результатів.

Представлені в табл. 4.1 показники демонструють високу точність роботи моделі на синтетичних даних, що підтверджує ефективність її архітектури та стратегії навчання.

Попри те, що результати на реальних даних нижчі, ніж на синтетичних, вони залишаються достатньо високими для практичного використання. Зниження точності може бути обумовлене варіативністю реальних даних, складністю сцен та різноманіттям об'єктів.

Узагальнюючи, процес навчання демонструє, що навіть при обмежених ресурсах та складності даних сучасні архітектури нейронних мереж можуть досягати високих результатів у задачах сегментації. Ретельний вибір гіперпараметрів та поетапний підхід до навчання є ключовими факторами успіху.

Висновки за розділом 4

У розділі представлено результати обчислювального експерименту, який охоплював аналіз роботи сегментатора на синтетичних та реальних даних, а також опис процесу навчання моделі. Отримані результати свідчать про високу ефективність моделі DeepLabv3+ для задач сегментації зображень у міських умовах.

На наборі даних Synscapes модель продемонструвала високі показники точності, що свідчить про її здатність розпізнавати ключові об'єкти у стандартизованих умовах. Це стало можливим завдяки використанню сучасних підходів до архітектури нейронних мереж та ефективних методів оптимізації.

Результати на наборі даних Cityscapes показали, що адаптація до реальних даних є складнішим завданням через варіативність сцен та складність об'єктів. Проте модель досягла прийняттого рівня точності, що підтверджує її потенціал для практичного застосування. Зниження функції втрат та стабільність метрик протягом етапу навчання свідчать про добре налаштовані гіперпараметри та успішну стратегію навчання.

Графічні матеріали та числові результати підкреслюють, що використання попередньо навчених моделей та двоетапний підхід до навчання є ефективним рішенням для задач сегментації. Подальші покращення можуть бути досягнуті шляхом оптимізації аугментації даних, збільшення обсягу реальних даних та впровадження додаткових методів регуляризації.

ВИСНОВКИ

У кваліфікаційній роботі було досліджено процес сегментації зображень у реальному часі для автономних транспортних засобів із використанням сучасної архітектури DeepLabv3+. Робота включала аналіз предметної області, розробку та адаптацію алгоритмів сегментації, а також експериментальну перевірку їх ефективності на синтетичних і реальних даних.

Розроблений підхід дозволив досягти високих результатів на синтетичних даних Synscapes і прийнятних результатів на реальних даних Cityscapes. Використання двоетапного процесу навчання, попередньо натренованих моделей та методів аугментації підтвердило свою ефективність у підвищенні точності сегментації. Використання синтетичних даних на початкових етапах дозволяє швидко навчити модель основним шаблонам об'єктів. Тонке налаштування на реальних даних є критично важливим для адаптації моделі до варіативності реального світу.

Метрики, такі як mIoU, F1-score та Pixel Accuracy, дозволяють комплексно оцінювати якість сегментації та відслідковувати прогрес під час навчання. Оптимізація гіперпараметрів і використання сучасних бібліотек, таких як PyTorch і OpenCV, забезпечили ефективну реалізацію системи сегментації.

Результати роботи мають потенціал для практичного використання в автономних транспортних системах. Подальший розвиток може включати розширення наборів даних, вдосконалення алгоритмів обробки зображень та впровадження додаткових методів оптимізації.

Результати дослідження можна застосовувати для підвищення ефективності та надійності систем комп'ютерного зору автономних транспортних засобів, що дозволить покращити здатність розпізнавання об'єктів на дорозі в реальному часі. Оптимізовані методи можуть бути використані на вбудованих системах, таких як FPGA, що сприятиме зменшенню вартості та енергоспоживання апаратної частини автономних транспортних засобів, а також забезпеченню стабільної роботи в складних дорожніх умовах.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Lozytskyi A., Lamtiuhova S. Real-time image segmentation for autonomous vehicles. *Learning & Teaching: after War and during Peace* [Electronic Edition]: Conference Proceedings of III International Scientific & Practical Conference, Kharkiv, Ukraine, 8 November, 2024. Kharkiv : KNPU, 2024.
2. Bishop C. M. *Pattern Recognition and Machine Learning*. New York : Springer, 2006. 738 p.
3. Mask R-CNN / K. He, G. Gkioxari, P. Dollár, R. Girshick. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2020. Vol. 42, No. 2. P. 386–397.
4. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs / L. C. Chen, G. Papandreou, I. Kokkinos [et al.]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2018. Vol. 40, No. 4. P. 834–848.
5. Badrinarayanan V., Kendall A., Cipolla R. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2017. Vol. 39, No. 12. P. 2481–2495.
6. Shelhamer E., Long J., Darrell T. Fully Convolutional Networks for Semantic Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2017. Vol. 39, No. 4. P. 640–651.
7. Learning both Weights and Connections for Efficient Neural Networks / S. Han, J. Pool, J. Tran, W. J. Dally. *Advances in Neural Information Processing Systems*. 2015. Vol. 28. P. 1135–1143.
8. Zhu M., Gupta S. To Prune, or Not to Prune: Exploring the Efficacy of Pruning for Model Compression. 2017. URL: <https://arxiv.org/abs/1710.01878> (дата звернення: 29.10.2024).
9. Frankle J., Carbin M. The Lottery Ticket Hypothesis: Finding Sparse, Trainable Neural Networks. 2019. URL: <https://doi.org/10.48550/arXiv.1803.03635> (дата звернення: 29.10.2024).

10. Gonzalez R. C., Woods R. E. Digital Image Processing. 4th ed. New York : Pearson, 2018. 1022 p.
11. ENet: A Deep Neural Network Architecture for Real-Time Semantic Segmentation / A. Paszke, A. Chaurasia, S. Kim, E. Culurciello. 2016. URL: <https://arxiv.org/abs/1606.02147> (дата звернення: 28.10.2024).
12. Poudel R. P. K., Liwicki S., Cipolla R. Fast-SCNN: Fast Semantic Segmentation Network. 2019. URL: <https://arxiv.org/abs/1902.04502> (дата звернення: 28.10.2024).
13. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications / A. G. Howard, M. Zhu, B. Chen [et al.]. 2017. URL: <https://arxiv.org/abs/1704.04861> (дата звернення: 28.10.2024).
14. Chollet F. Xception: Deep Learning with Depthwise Separable Convolutions. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017. P. 1251–1258.
15. Kingma D. P., Ba J. Adam: A Method for Stochastic Optimization. *International Conference on Learning Representations (ICLR)*. 2015. URL: <https://arxiv.org/abs/1412.6980> (дата звернення: 28.10.2024).
16. Otsu N. A Threshold Selection Method from Gray-Level Histograms. *IEEE Transactions on Systems, Man, and Cybernetics*. 1979. 9(1). P. 62–66.
17. Canny J. A Computational Approach to Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 1986. PAMI-8(6). P. 679–698.
18. Marr D., Hildreth E. Theory of Edge Detection. *Proceedings of the Royal Society of London. Series B. Biological Sciences*. 1980. 207(1167). P. 187–217.
19. MacQueen J. Some Methods for Classification and Analysis of Multivariate Observations. *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*. 1967. P. 281–297.
20. Jain A. K., Murty M. N., Flynn P. J. Data Clustering: A Review. *ACM Computing Surveys*. 1999. 31(3). P. 264–323.
21. Ronneberger O., Fischer P., Brox T. U-Net: Convolutional Networks for Biomedical Image Segmentation. *Proceedings of the International Conference on*

Medical Image Computing and Computer-Assisted Intervention (MICCAI). 2015. P. 234–241.

22. 3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation / Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp [et al.]. *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*. 2016. P. 424–432.

23. Badrinarayanan V., Kendall A., Cipolla R. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 2017. 39(12). P. 2481–2495.

24. Kendall A., Badrinarayanan V., Cipolla R. Bayesian SegNet: Model Uncertainty in Deep Convolutional Encoder-Decoder Architectures for Scene Understanding. 2015. URL: <https://arxiv.org/abs/1511.02680> (дата звернення: 16.10.2024).

25. Rethinking Atrous Convolution for Semantic Image Segmentation / L.-C. Chen, G. Papandreou, F. Schroff, H. Adam. 2017. URL: <https://arxiv.org/abs/1706.05587> (дата звернення: 19.10.2024).

26. Magnus Wrenninge, Jonas Unger Synscapes: A Photorealistic Synthetic Dataset for Street Scene Parsing. 2018. URL: <https://arxiv.org/abs/1810.08705> (дата звернення: 02.01.2025).