

МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ "ЛЬВІВСЬКА ПОЛІТЕХНІКА"

КУРСОВА РОБОТА
з дисципліни
“МАШИННЕ НАВЧАННЯ”

на тему: “Класифікація БПЛА за формою та типом на відео та
фотографіях”

Студента 317 групи спеціальності
122 “Комп’ютерні науки”
Орлянського Максима
Андрійовича
Керівник
асистент Яцишин В.П.

Кількість балів: _____ Оцінка _____

Члени комісії

(підпис)

(вчене звання, науковий ступінь, прізвище та ініціали)

(підпис)

(вчене звання, науковий ступінь, прізвище та ініціали)

(підпис)

(вчене звання, науковий ступінь, прізвище та ініціали)

Львів – 2024

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ	3
ВСТУП.....	4
Висновки до розділу	5
РОЗДІЛ 1. АНАЛІЗ ЛІТЕРАТУРНИХ ДЖЕРЕЛ	6
Висновок до розділу	9
РОЗДІЛ 2. АНАЛІЗ МАТЕРІАЛІВ ТА МЕТОДІВ	11
2.1 Опис набору даних	11
Постановка завдання	11
2.2 Попередній огляд ResNet-50	12
2.3 Попередній огляд MobileNetV1	13
2.4 Попередній огляд AlexNet	14
2.5 Попередній огляд GoogleNet	15
2.6 Підготовка даних	17
Висновки до розділу	17
РОЗДІЛ 3. ЕКСПЕРИМЕНТИ	18
3.1 Інструменти розробки	18
3.2 Завантаження та підготовка даних	18
3.2 Класифікація БПЛА через ResNet-50	19
3.3 Класифікація БПЛА через MobileNetV1	20
3.4 Класифікація БПЛА через AlexNet	22
3.5 Класифікація БПЛА через GoogleNet	23
3.6 Співставлення та порівняння результатів моделей	24
Висновки до розділу	26
РОЗДІЛ 4. ОБГОВОРЕННЯ РЕЗУЛЬТАТІВ ЕКСПЕРИМЕНТІВ	27
Висновки до розділу	28
РОЗДІЛ 5. ВИСНОВКИ	29
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	30
ДОДАТОК А.....	32
ДОДАТОК Б	40
ДОДАТОК В	48

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

CNN - згорткові нейронні мережі

SVM – метод опорних векторів

Random Forest – ансамблевий алгоритм класифікації дерев рішень

БПЛА – безпілотний літальний апарат

ВСТУП

Дана робота присвячена проблемі класифікації безпілотних літальних апаратів(БПЛА) за типами моделей на відео та зображеннях. У зв'язку з розвитком цієї індустрії в Україні, що почала активно розростатися у 2023 році через масштабні бойові дії на сході держави з широким використанням цих технологій. Так БПЛА використовуються військовими та службою безпеки у різноманітних задачах розвідки та бою через що існує необхідність в швидкому виявленні та відповідно класифікації ворожих та союзних дронів на основі їх візуальних особливостей в тому числі форми, типу та кольору, що дозволяє вчасно зреагувати на небезпечну ситуацію та запобігти жертвам та втратам.

Наукове дослідження цієї проблеми вимагає використання різних методів та підходів швидкої класифікації на основі відео чи зображення. В даний час, вже є доволі багато досліджень у сфері виявлення БПЛА, проте існує мало інформації у відкритому доступі про методи їх швидкої візуальної класифікації.

Саме тому метою даної роботи є вирішення цієї проблеми шляхом застосування методів машинного та глибокого навчання для аналізу та подальшої класифікації БПЛА за формою та типом до певного класу. Результати та висновки отримані в ході цієї роботи, можуть бути вагомим внеском в розвиток сфери безпеки та повітряної оборони України та світу.

Завданням даного дослідження є:

1. Збір та попередня підготовка даних: Зібрати фотографії та відео популярних видів цивільних та бойових дронів станом на 2024 рік, та підготувати дані у потрібний для подальшого аналізу формат.
2. Проведення аналізу даних: Враховуючи візуальні ознаки БПЛА такі як, їхню форму та тип до якого вони належать, погрупувати зображення за класами, для можливого їхнього подальшого використання в процесі навчання.
3. Вибір моделі класифікації: Застосувати методи глибокого навчання, зокрема алгоритм претренованої моделі CNN, для виявлення ознак дронів з подальшою їх класифікацією. Враховувати можливі візуальні артефакти на зображенні, що можуть вплинути на процес визначення ознак.
4. Оцінка моделі: На основі процесу навчання та тестування моделі оцінити точність і правильність моделі класифікації, з подальшим виявленням числа неправильно класифікованих БПЛА та можливим покращенням точності моделі за рахунок аналізу отриманих результатів.
5. Звіт до отриманих результатів: Підготувати результати дослідження у вигляді документації.

Актуальність обраної теми відображає важливість даної проблеми, та її відповідність сучасним потребам науки та суспільства. Класифікація БПЛА за формами та типами є особливо актуальними в даний час у зв'язку з широким їх застосуванням у зоні бойових дій, станом на 2024 рік. Основною причиною дослідження є висока інтенсивність застосування різних типів цих апаратів військовими, що часто спричиняє багато жертв та руйнувань та завдає шкоди державі, що є важливою проблемою та потребує ретельного аналізу та розробки ефективних методів класифікації БПЛА аби запобігти випадковому збиванню

союзних та навпаки пропуску ворожих дронів.

Способи розрізнення БПЛА за їх типами та розмірами є важливою складовою для їх подальшої класифікації. Саме тому дослідження в даній галузі забезпечить наукове підґрунтя для подальших наукових робіт у сфері класифікації дронів, та дасть прямий практичний досвід у даній галузі, та тим самим буде мати прямий практичний вплив на повітряну оборону країни.

Окрім згаданого вище, класифікація БПЛА за їхніми характеристиками такими як форма, тип матиме значення для глибшого розвитку стратегії безпеки та оборони. Виявлення візуальних ознак, що впливають на класифікацію цих дронів сприятиме покращенню реагування на них, та допоможе у розробці ефективних методів протидії загрозам, що становлять бойові та розвідувальні БПЛА.

Об'єктом дослідження в даній роботі є типи безпілотних літальних апаратів та їхні візуальні характеристики такі як форма і колір. **Предметом дослідження** є методи та алгоритми класифікації, що використовуються на даний момент у подібного роду задачах.

Основна увага дослідження спрямована на виявлення та аналіз візуальних ознак БПЛА, що сприяють покращенню їх класифікації. Об'єкт і предмет дослідження відображають основні аспекти, які аналізуються у даній роботі з метою розробки системи класифікації безпілотних літальних апаратів та покращення їхньої класифікації у сфері безпеки та оборони, сприяючи ефективному вирішенню завдань у військовому контексті.

Висновки до розділу

У розділі “ВСТУП” даної роботи було проаналізовано проблему класифікації БПЛА на сучасному етапі у сфері безпеки та повітряної оборони, аби зрозуміти потребу у розробці методу їх класифікації на основі візуальних ознак. Окрім цього була визначена мета дослідження, що складається з аналізу ознак, наявних на даний момент у відкритому доступі зображень та відео, про існуючі типи БПЛА з подальшою класифікацією до певних класів на основі цих даних. Також у даному розділі було проаналізовано актуальність даної проблематики на даний момент, а також окреслено об'єкт та предмет дослідження у подальших розділах.

РОЗДІЛ 1. АНАЛІЗ ЛІТЕРАТУРНИХ ДЖЕРЕЛ

Для класифікації об'єктів на зображенні та у відеопотоці існує багато різноманітних методик заснованих на роботі машинного та глибокого навчання. Серед яких автори статті “A Survey on Image Classification Techniques” [1] (2021) виділяють наступні сучасні широко популярні методи:

1. Згорткові нейронні мережі
2. Метод опорних векторів
3. Трансферне навчання
4. К найближчих сусідів
5. Алгоритм Random Forest

Так згорткові нейронні мережі виділяють ознаки з вхідного зображення шляхом застосування фільтрів та процесу згортки після чого на основі зменшеного за кількістю ознак зображення використовують лінійні шари для подальшої класифікації певною функцією класифікації. Перевагою CNN є те, що вони можуть застосовуватися на зображеннях будь-якої розмірності та мають необмежену можливість багатокласової класифікації, а також дають хорошу точність, проте потребують великої кількості даних для навчання.

У свою чергу метод опорних векторів для об'єктів на зображенні буде границю рішень, яка найкраще розділяє дані одного класу від іншого, після чого знаходить на основі отриманих ознак з зображення опорні вектори, що грають роль у тому, де буде відбуватися розділення класів гіперплощиною. Серед переваг SVM є їхня мала потреба в даних для навчання та велика точність класифікації, проте метод опорних векторів працює лише для бінарної класифікації об'єктів, через що у задачах багатокласової класифікації він не підходить, або вимагає глибокого втручання в сам процес навчання і зміни моделі.

Говорячи ж про трансферне навчання, то воно реалізує процес навчання існуючої моделі на подібних за семантичним значенням даних, для покращення швидкості і точності навчання на задачі подібного типу. Перевагою трансферного навчання є швидке навчання та кращі результати, ніж у моделей, які навчають з нуля, проте трансферне навчання має недолік у вигляді втрати гнучкості і потреби сумісності даних, оскільки вона використовує попередньо навчені ваги, що були зафіксовані у попередньому навчанні моделі.

Алгоритм К найближчих сусідів є не параметричним, так званим лінівним алгоритмом класифікації. Даний алгоритм часто використовується не лише для задач класифікації об'єктів на зображенні но і у задачах регресії. Алгоритм класифікує об'єкти до певного класу на зображенні на основі значень відстаней для отриманих ще не класифікованих ознак та вже класифікованих К найближчих об'єктів. Перевагою даного алгоритму є насамперед відсутність потреби в навчанні, оскільки він не запам'ятовує жодних значень чи параметрів, а лише класифікує за вже наявними класами і ознаками, також серед переваг даного алгоритму часто виділяють його доволі хорошу точність. Проте, даний алгоритм має і ряд недоліків таких як велика чутливість до зашумленості вхідного зображення та потреба у низькій вимірності вхідних даних, оскільки при збільшенні вимірів алгоритм втрачає свою ефективність.

І тепер розглянемо алгоритм Random Forest, який є ансамблевим, і складається з певної кількості дерев рішень кожне з яких навчається на своїй випадковій частині вхідних ознак зображення, після чого в залежності від типу задачі для якого використовувався Random Forest, регресійної чи класифікації відбувається об'єднання результатів дерев шляхом голосування за клас, до якого було класифіковано більшість ознак. Серед переваг даного алгоритму виділяють його боротьбу з перенавчанням, та високу точність класифікації, а також здатність враховувати важливість різних ознак зображення. Недоліком виділяють можливість втрати контексту при обробці зображення та вимагає налаштування великого числа гіперпараметрів.

Тому підсумовуючи запропоновані авторами статі підходи до класифікації об'єктів на зображенні можна представити у вигляді таблиці:

Таблиця 1

Методика	Переваги	Недоліки
Згорткові нейронні мережі	Універсальний алгоритм, що здатен завжди давати доволі хороші результати	Потреба у великому числі тренувальних даних для хорошої точності класифікації
Метод опорних векторів	Мала потреба в даних для навчання	Працює лише із бінарною класифікацією
Трансферне навчання	Швидке і точне навчання	Недостатньо гнучкий при навчанні і чутливий до даних
К найближчих сусідів	Простий у імплементації та не потребує попереднього навчання моделі. Має хорошу точність на добре попередньо підготовлених даних.	Чутливий до шумів на зображенні. Погіршується ефективність при збільшенні вимірності даних.
Random Forest	Висока точність, та боротьба з перенавчанням. Здатність враховувати важливість ознак.	Втрата контексту при обробці зображення, потреба у налаштуванні великого числа гіперпараметрів.

Серед запропонованих методів класифікації для даного дослідження був обраний метод глибокого навчання на основі CNN мережі, що зумовлено балансом співвідношення точності і швидкості класифікації під час роботи моделі, а також обраною проблематикою, оскільки класифікація БПЛА за типом та формою є задачею багатокласової класифікації, а також у зв'язку з тим, що зазвичай зображення з камер, які фіксують політ БПЛА є багатоканальним, а отже багато вимірним, через що такі методи як метод опорних векторів, К найближчих сусідів та Random Forest не підходять, через їхні головні недоліки наведені в таблиці 1. Проте, можливе застосування

трансферного навчання для згорткової нейронної мережі, шляхом завантаження однієї претренованої моделі з числа існуючих популярних на даний момент. Так серед моделей CNN існує багато різноманітних навчених моделей, що дають різну точність і швидкість класифікації в залежності від вхідних даних. Саме тому спробуємо детально проаналізувати популярні на даний момент моделі глибокого навчання CNN для класифікації об'єктів на зображеннях та відео, та порівняти їх.

У своїй статті “Pre-trained CNNs: Evaluating Emergency Vehicle Image Classification” [2] (2023) автори досліджують та порівнюють архітектури та точність і швидкість популярних у сфері класифікації об'єктів на зображеннях претренованих моделей VGG19, VGG16, MobileNetV3Large, MobileNetV3Small, MobileNetV2, MobileNetV1 на прикладі класифікації автомобілів екстрених служб. Зокрема в результаті порівняння часу і точності класифікації даних моделей отримані наступні результати:

Таблиця 2

Перетренована модель	Точність Accuracy	Втрати Loss	Час навчання
VGG19	87,01	0,4864	80,58
VGG16	89,01	0,4472	69,00
MobileNetV3Large	58,44	0,8772	36,99
MobileNetV3Small	38,96	1,0491	33,59
MobileNetV2	90,91	0,2766	34,30
MobileNetV1	92,21	0,2688	32,47

Так автори вказують MobileNetV1 як найбільш збалансованою по значеннях точності класифікації та часу навчання серед запропонованих моделей.

Оглядаючи статтю “Comparison Study on Convolution Neural Network (CNN) Techniques for Image Classification”[3] (2020) автори порівнюють претреновані моделі ResNet-50, AlexNet, GoogleNet. Зокрема в статті наводиться загальна кількість шарів та нейронів у кожній з цих трьох моделей. Так AlexNet містить в собі 60 мільйонів параметрів та 650,000 нейронів у 5 згорткових та 3 зв'язних шарів. ResNet-50 в свою чергу містить в загальному 25,557,032 параметрів. GoogleNet ж в свою чергу має найбільшу архітектуру із запропонованих і містить величезну кількість параметрів. Так при порівнянні даних моделей із різними значеннями learning rate та оптимізаторів автори порівнюють час та точність класифікації квітів для 5ти класів. В результаті отримано наступні результати:

Таблиця 3

	Learning Rate			
Модель	0,001	0,0001	0,0005	0,00005
AlexNet	99,50	98,40	99,54	97,36
GoogleNet	99,74	96,84	99,82	94,64
ResNet-50	99,42	98,42	99,58	97,30

Таблиця 4

Модель	Число епох навчання із наченням Learning Rate 0,00005		
	10	20	30
AlexNet	94,88	98,50	97,44
GoogleNet	94,92	97,56	97,56
ResNet-50	94,02	96,16	98,54

За результатами порівняння співвідношення часу навчання та точності автори виділяють найкращою по показниках ResNet-50.

Із аналізу наведених раніше статей, ми дізналися, що станом на момент проведення порівняння популярних претренованих моделей по співвідношенню точності та швидкості класифікації на зображенні найкращими на даний момент є ResNet-50 та MobileNetV1.

У статті “Combining MobileNetV1 and Depthwise Separable convolution bottleneck with Expansion for classifying the freshness of fish eyes” [4] (2022) автори порівнюють MobileNetV1, MobileNetV2, ResNet50, DenseNet 121, VGG16, Nasnet Mobile, MB-BE(111) моделі для класифікації якості риби на певні класи свіжості. В результаті дослідження автори заявляють, що найкращу точність досягає ResNet50 серед усіх заявлених моделей, так автори наводять таблицю результатів класифікації:

Таблиця 5

Модель	Train Acc	Test Acc
MobileNetV1	53,87	57,97
MobileNetV2	54,29	55,35
ResNet50	84,86	78,47
DenseNet 121	35,96	43,05
VGG16	34,26	41,00
Nasnet Mobile	35,77	40,89
MB-BE(111)	56,75	63,21

Отже, із результатів отриманих авторами для усіх наведених моделей, бачимо, що найкраще себе показав ResNet-50, MobileNetV1, AlexNet, GoogleNet.

У подальшому дослідженні будемо використовувати та порівнювати точність цих претренованих моделей наведених авторами статей, для проблематики класифікації БПЛА за формою та типами, аби виявити, яка з моделей є найбільш підходящою для обраної проблематики

Висновок до розділу

У даному розділі необхідно було провести дослідження поточного стану обраної проблеми, з аналізом сучасних літературних джерел авторів, які досліджують сучасні методи розв’язання подібного типу задач. Розглядаючи дану проблематику та роботи дослідників, важливим було виокремити основні алгоритми та методи з найбільшим числом переваг і найменшим числом недоліків, що використовуються для класифікації об’єктів на зображенні. А

також на основі сучасних наукових статей для обраного алгоритму дослідити в якому стані на даний момент він перебуває.

РОЗДІЛ 2. АНАЛІЗ МАТЕРІАЛІВ ТА МЕТОДІВ

2.1 Опис набору даних

Для класифікації БПЛА за формою та типом необхідний достатньо великий за розмірами набір даних, аби навчання та тестування моделі мали велику точність. Саме тому для дослідження були взяті 4 різних набори даних з платформи Kaggle з фотографіями у форматі .jpg.

“Fixed Wing UAV - Plane” [5] – це набір даних, що містить 3571 фотографію безпілотних літальних апаратів з типом монокрило, а також безпілотних літаків, більшість з яких є легкомоторними.

“Fixed Wing UAV Dataset” [6] – набір даних, що містить 554 зображення причому з яких оригінальними є 319, а решта це горизонтальні перетворення існуючих зображень.

“Drone Dataset (UAV)” [7] – набір даних, містить 1359 перемішаних фотографій БПЛА з обертальними крилами.

“UAV Detection Dataset” [8] – набір даних, що містить 1962 фотографії дронів з обертальним крилом.

Усі наведені набори даних у зв’язку з тим, що фотографії в них були призначені для задачі виявлення, а не класифікації БПЛА, містили в собі не погруповані за формою і типом фото. Тому, на основі даних з цих наборів був сформований власний набір, що містить 5420 фотографій [9], через те що частина фото були непідходящими для задач класифікації. Так він складається з чотирьох класів фото дронів, зокрема це БПЛА:

1. З монокрилом
2. Квадрокоптери
3. Гексакоптери
4. Октокоптери

Окрім цього наведений набір даних для навчання і тестування моделі буде розділений у співвідношенні 7:3 під час подальшого дослідження.

Постановка завдання

Для обраного методу трансферного навчання CNN мережі на основі претренованої моделі у розділі аналізу літературних джерел, було наведено різні популярні претреновані моделі для задач класифікації об’єктів на зображенні, такі як ResNet-50, MobileNetV1, VGG16, VGG19 та інші, за останні 5 років.

Так аналіз робіт дослідників показав, що певні моделі показували кращі результати, а певні гірші. Найкращими претренованими моделями виявилися ResNet-50, MobileNetV1, AlexNet, GoogleNet. Тому аби наочно удостовіритися в цьому, і обрати модель, яка буде давати найкращі результати класифікації БПЛА за типами та формою, використаємо усі ці чотири моделі та порівняємо їх результати на обраному наборі даних із подальшими оцінками та порівняльними характеристиками результатів. Для того аби реалізувати це завдання виконаємо наступні кроки:

1. Завантаження та попередня обробка зображень, які будуть подаватися на вхід кожній з моделей, в залежності від їхньої внутрішньої архітектури.

2. Завантаження претренованих моделей та значень їхніх ваг.
3. Тренування та тестування моделей з різними значеннями Learning Rate
4. Оцінка отриманих результатів для кожної з моделей через метрики Accuracy, Recall та порівняння функції втрат на тренувальних та тестувальних даних
5. Аналіз отриманих результатів та вибір на їх основі найбільш оптимальної моделі.

2.2 Попередній огляд ResNet-50

На діаграмі нижче представлена внутрішня архітектура мережі ResNet-50 та шари, що в ній використовуються:

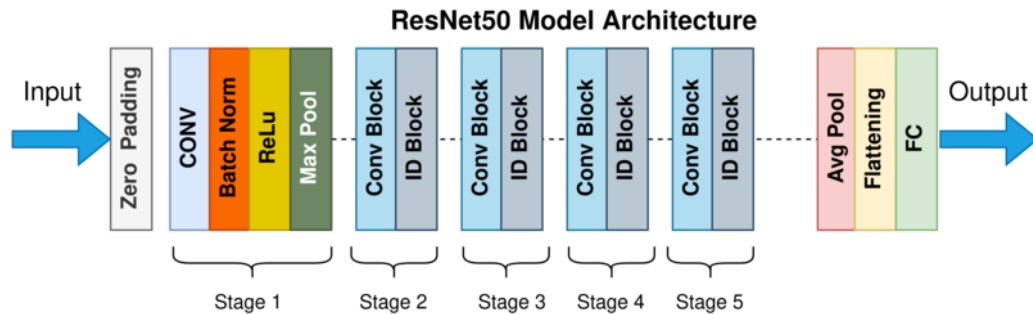


Рис 2.1 Внутрішня архітектура ResNet-50 [10]

Так із діаграми можемо бачити, що безпосередня внутрішня будова ResNet-50 складається із 5 шарів. Перший з яких застосовує фільтр матрицю розмірності 7x7, після чого для отриманого процесом згортки шару застосовується Batch нормалізація, в слід за чим застосовується функція активації ReLu, що усуває усі від'ємні значення на 0, після чого застосовується MaxPool шар, що максимізує значення ознак та відповідно передає їх у процес обробки 4-ма згортковими та ID блоками, архітектура кожного з яких наведена нижче:

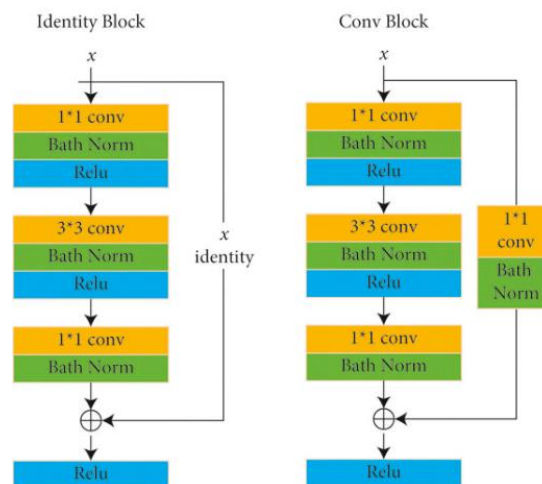


Рис 2.2 Будова блоків Conv та ID [11]

Відповідно під час роботи кожного з блоків Conv, застосовується згортка розмірності 1x1 та нормалізація з функцією активації ReLu, після якої виконується повторне зменшення ознак фільтром 3x3 з подальшою нормалізацією та активацією, та поданням на вхід фільтру 1x1. В результаті роботи згорткового блоку на вихід подається об'єднаний вхідний шар

пройдений фільтром 1x1, і шар що пройшов через 3 фільтри, після їх об'єднання вони подаються на вхід функції активації.

Аналогічний процес відбувається із ID блоком архітектури, що повторює все те саме, але за винятком того, що на виході об'єднується не змінений вхід та оброблений 3ма фільтрами шар, та також подаються на вхід функції активації.

Такі блоки Conv, ID повторюються 4 рази, після чого застосовується шар усереднення ознак, перетворення згорткового шару на набір ознак для лінійного, з подальшою класифікацією на основі функції SoftMax класифікації, яка математично задається наступним чином:

$$s(x_i) = \frac{e^{x_i}}{\sum_{j=1}^n e^{x_j}},$$

де $s(x_i)$ ймовірність того що даний об'єкт на зображенні належить до певно класу

2.3 Попередній огляд MobileNetV1

Розглянемо внутрішню будову MobileNetV1 та з яких шарів вона складається:

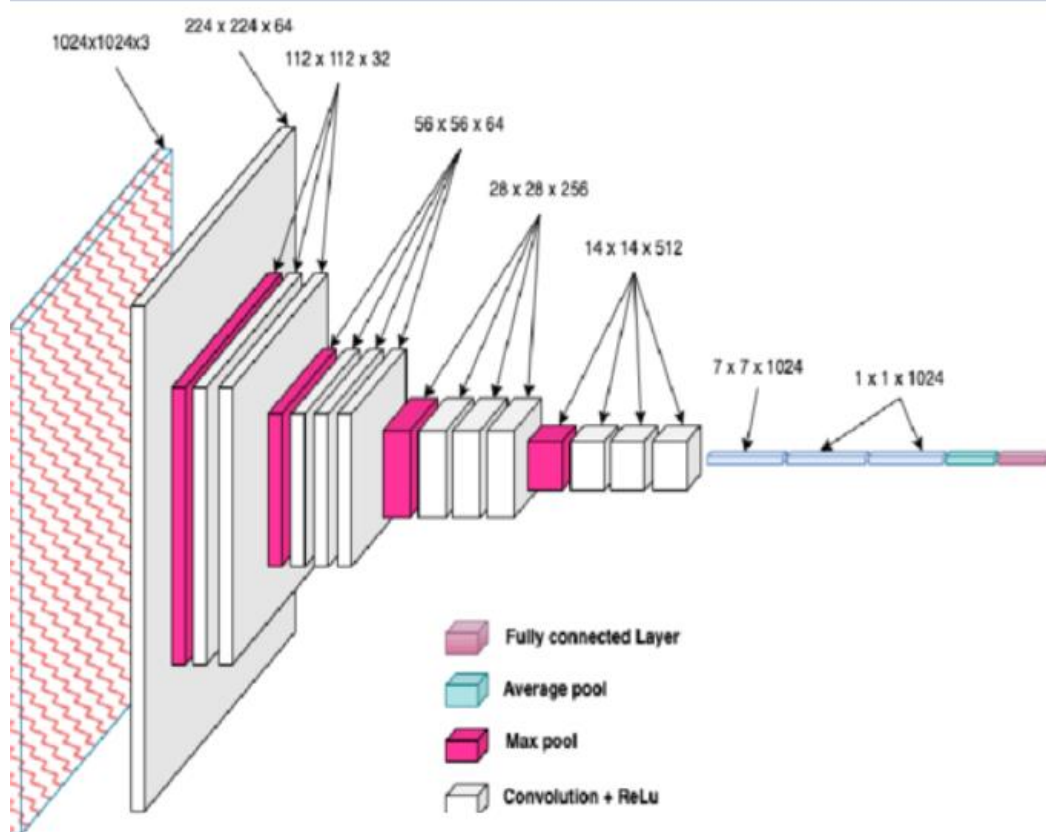


Рис 2.3 Загальна архітектура MobileNetV1 [12]

У випадку архітектури MobileNetV1, то вона містить у собі 4 групи згорткових та максимізуючих у поєднанні з функцією активації ReLu шарів. Так перша група складається з згорткового шару та функції активації, одразу після яких іде шар максимізації ознак, після чого ще два згорткових шари і їх функції активації. В другій, третій та четвертій групах відбувається аналогічне дублювання архітектури, де спочатку відбувається максимізація ознак з попереднього шару, після чого вхідне перетворене зображення 3 рази згорткові

шари після кожного з яких застосовується функція активації ReLu. Наприкінці роботи згорткових шарів відбувається усереднення ознак 3 рази, після чого згортковий шар перетворюється на список ознак, що передаються лінійному шару, де після його роботи застосовується функція SoftMax для виділення ймовірності належності до певного класу об'єкту.

2.4 Попередній огляд AlexNet

Як і раніше розглянемо перше архітектуру мережі AlexNet у вигляді діаграми:

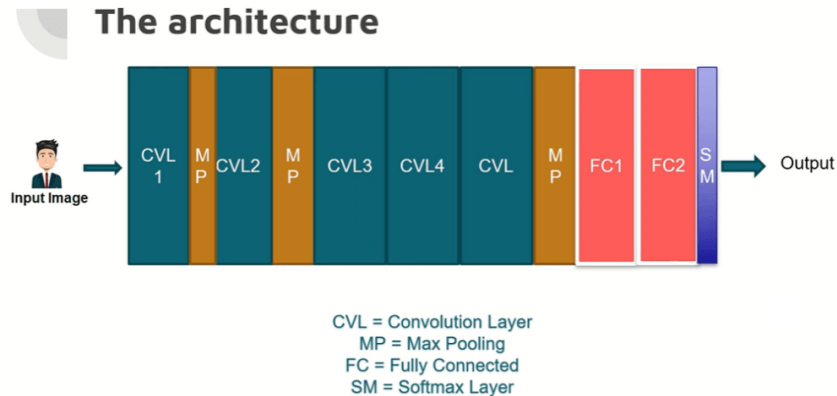


Рис 2.4 Внутрішня архітектура AlexNet [13]

Так розроблена ще в 2012 році по сей день активно використовується і є дуже популярною у сфері класифікації об'єктів на зображенні через простоту її архітектури у порівнянні зі всіма іншими наведеними раніше архітектурами, а також малою кількістю параметрів.

Загалом як показано на діаграмі, AlexNet складається із 5 згорткових шарів та 2ох лінійних шарів. Відповідно при передачі зображення на вхід моделі, спочатку для перших двох шарів застосовується згортка та максимізація ознак. Після чого на виході після максимізації ознак другого шару, вона проходить ще через 3 згорткових шари без додаткових перетворень. І наприкінці роботи згорткових шарів на неї в останній раз застосовується операція максимізації ознак, після чого шар перетворюється у набір ознак для двох лінійних шарів моделі, де наприкінці роботи виділення ознак, результат передається функції SoftMax для визначення ймовірності того чи об'єкт належить до певного класу.

2.5 Попередній огляд GoogleNet

Останньою і найбільш комплексною за кількістю різноманітних шарів є GoogleNet, нижче наведено діаграму, що відображає загальну її архітектуру:

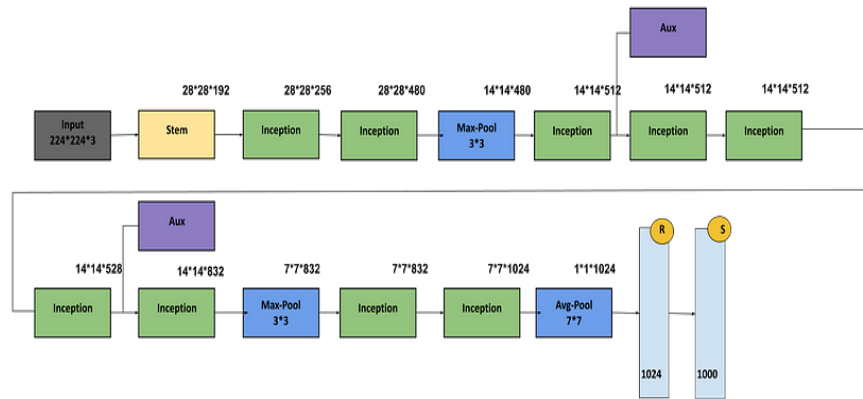


Рис 2.5 Загальна архітектура GoogleNet [14]

На відміну від попередніх 3ох архітектур GoogleNet має вимогу, щоб вхідне зображення було попередньо оброблене та зменшене до розмірності 224x224, після чого його вже можна подавати на вхід мережі. Узагальнено, архітектура мережі складається з 1го блоку під назвою Stem, та 9 блоків Inception, 2 шарів максимізації, 1го усереднення, та 2х шарів Aux для роботи підготовки виділення ознак, що ж д кількості лінійних шарів то їх 2, а вихід подається на функцію класифікації SoftMax. Загалом послідовність проходження зображення через усі шари наведена вище. А ми детальніше розглянемо принцип роботи блоків Stem, Inception, Aux. Так на діаграмі нижче наведено процес обробки Stem шару:

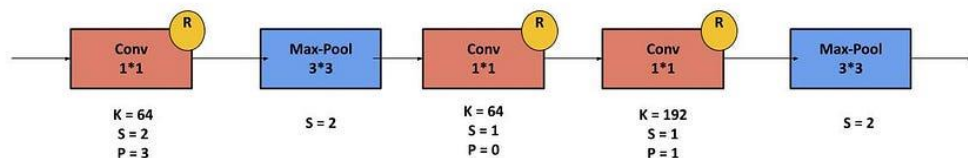


Рис 2.6 Архітектура шару Stem [15]

Відповідно для вхідного зображення застосовується згортка 1x1 з значеннями padding відступів від країв зображень padding=3, та значенням stride = 2. Після чого на отриманий вихід застосовується функція активації ReLu. Та вихід функції активації подається на шар максимізації розмірності 3x3, після чого під ряд застосовуються два згорткових шари та 2 функції активації, і на кінець вихід подається в шар максимізації MaxPool. Так після завершення роботи максимізації вихід подається для подальшої обробки блоками Inception, загальну архітектуру яких наведено нижче:

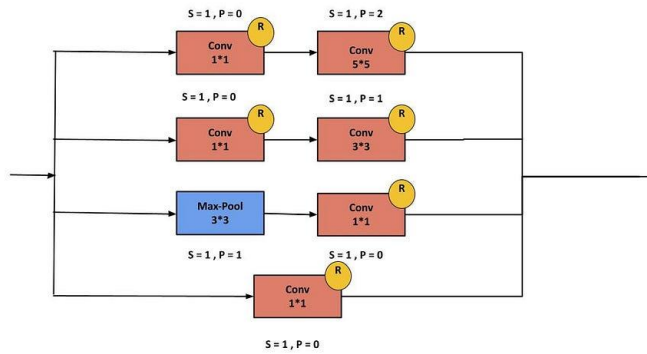


Рис 2.7 Архітектура шару Inception [16]

Загалом архітектура Inception шару складається із 4ох паралельних обробок вхідного шару, результати яких об'єднуються на виході. Так як показано на рисунку вище, вхід однаково обробляється двома згортковими шарами, де єдина відмінність це розміри фільтрів та значення padding. В свою чергу для інших двох паралельних шарів ситуація дещо інша, а саме 3й паралельний шар спочатку максимізує значення ознак, після чого передає їх на вхід операції згортки розмірності 1x1. І на кінець останній 4 паралельно працюючий шар застосовує лише одну єдину операцію згортки також розмірності 1x1.

Таким чином було розглянуто архітектуру блоку Inception. Також під час роботи також застосовується блок Aux загальна архітектура якого наведена нижче:

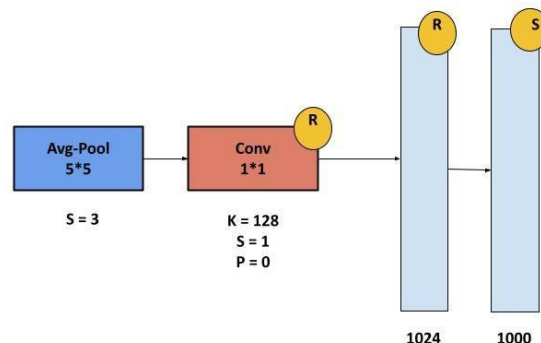


Рис 2.8 Архітектура шару Aux [17]

Відповідно на вхід застосовується усереднення ознак, проходження через ще один згортковий шар, після чого відбувається виділення ознак для проходження через лінійні шари, та класифікація використовуючи функцію SoftMax.

2.6 Підготовка даних

У зв'язку з тим, що певні обрані претреновані моделі такі як GoogleNet вимагають від вхідних даних певного попереднього опрацювання. То перед передачею зображень моделям для навчання та тестування потрібно провести невелику підготовку цих фотографій. Так перш за все потрібно, задати усім фотографіям однакову висоту і ширину, оскільки описані в розділі 2.1 завантажені і поділені по класах дані є різні за розмірами. Так спираючись на функціональні вимоги мережі GoogleNet, для її роботи необхідно, щоб зображення обов'язково було розмірності 224x224.

Окрім зміни висоти та ширини, корисним буде нормалізувати значення кольорів пікселів для каналів RGB, оскільки це допоможе стандартизувати значення кольорів пікселів усіх зображень, а також забезпечит стабільність та допоможе запобігати перенавчанню моделей під час опрацювання цих зображень. Здійснити нормалізацію і стандартизацію даних, можна якщо відняти від значення усіх пікселів середнє значення для кожного з каналів кольору, та розділити на значення стандартного відхилення кольору у значенні кольору кожного з каналів.

І на кінець, оскільки при програмній реалізації моделі в реальному світі працюють з типом даних тензор, то самі зображення потрібно перетворити на ці тензори, аби моделі могла на основі цих даних навчатися та робити тестування.

Висновки до розділу

В процесі аналізу матеріалів та методів, що будуть використовуватися у подальшому дослідженні, було обрано на основі результатів дослідників із попереднього розділу найкращі на даний момент моделі класифікації об'єктів на зображенні та проаналізовано їхню архітектуру. В результаті чого було виявлено вимоги до вхідних даних, які мають ці моделі та особливості їх внутрішньої архітектури. Також, було проаналізовано вхідний набір даних, що буде використовуватися для навчання моделей аби класифікувати БПЛА за формою та типом, а також необхідні підготування цих даних такі як нормалізація кольорів пікселів та стандартизація зображень, аби покращити якість навчання моделей.

РОЗДІЛ 3. ЕКСПЕРИМЕНТИ

Проведення експериментів для обраної тематики класифікації БПЛА за формою та типом, є важливою складовою процесу дослідження, оскільки дозволяє точно оцінити наскільки добре відпрацьовують обрані в розділі 2 претреновані моделі ResNet-50, MobileNetV1, AlexNet, GoogleNet, та зрозуміти, яка з них є найкращою у даній проблематиці.

3.1 Інструменти розробки

Для розробки системи класифікації БПЛА за типом та розміром було обрано мову програмування Python, оскільки це високорівнева мова, з легким і зрозумілим синтаксисом, а також широко застосовуваний інструмент у сфері комп'ютерного зору та обробки даних, на якому є багато різних бібліотек для роботи з нейронними мережами та машинним навчанням, зокрема і згорткові нейронні мережі, що є основним предметом дослідження в даній роботі. Окрім цього серед переваг Python є також широка його підтримка великою кількістю розробників по всьому світу.

Зокрема для розробки системи класифікації БПЛА були використані наступні бібліотеки:

- PyTorch - фреймворк для роботи з нейронними мережами, що містить в собі велику кількість претренованих моделей, зокрема і ті що досліджуються в даній роботі: ResNet-50, GoogleNet, AlexNet
- Transformers – фреймворк для роботи з моделями глибокого навчання, що містить велику кількість претренованих моделей, зокрема і модель MobileNetV1, якої немає у PyTorch
- Sklearn – бібліотека машинного навчання, що містить у собі різноманітні інструменти машинного навчання, що використовуються для оцінки точності.
- Numpy – бібліотека для роботи з матрицями та різного роду множинами та математичними функціями, що часто застосовують для наукових обчислень
- Pandas – бібліотека для обробки та аналізу даних, часто застосовується для зчитування та агрегації вхідних даних.
- Matplotlib – бібліотека для візуалізації даних

3.2 Завантаження та підготовка даних

Описаний в розділі матеріалів та методів пункт 2.6 про попередню підготовку та завантаження даних зображень БПЛА реалізуємо в даному пункті. Так отримавши список усіх назв директорій класів, де зберігаються відповідні фотографії БПЛА для кожного класу з набору даних “UAVs” завантажимо їх та попередньо змінимо їм розмірність та одразу нормалізуємо їх, аби в подальшому при перетворенні в тензори і розділенні на тренувальні та тестові дані, моделі змогли навчатися на вже підготовлених даних. В результаті завантаження даних, виведемо по 1 зображенню з кожного класу представленого в розділі 2.1:

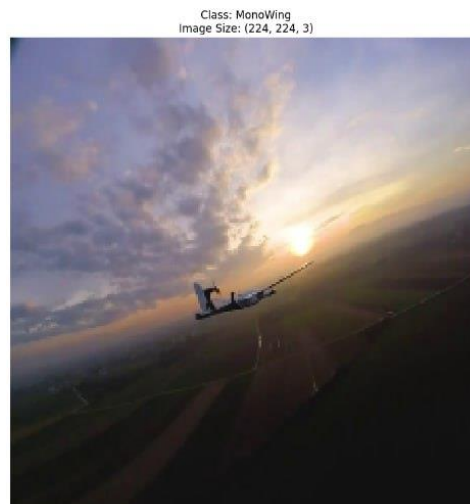


Рис 3.1, 3.2 Приклад зображень з класів Гексакоптери та Монокрило

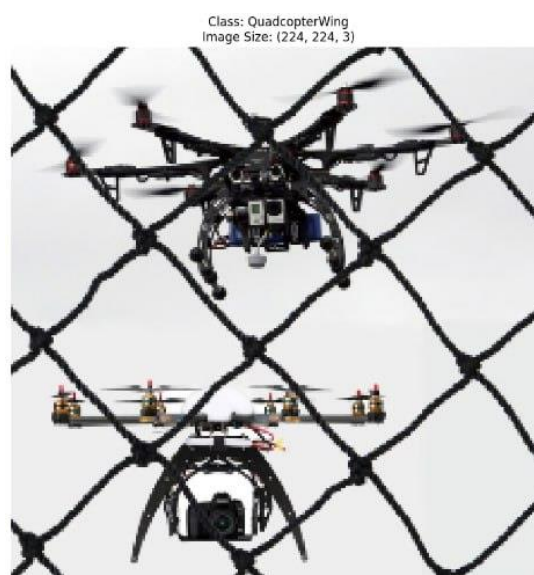
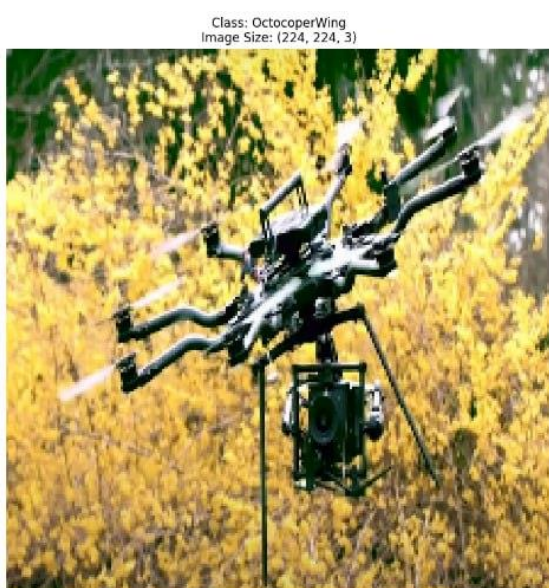


Рис 3.3, 3.4 Приклад зображень з класу Октокоптери та Квадрокоптери

Тепер ми маємо підготовлені для навчання усіх моделей дані, які будуть розділені у співвідношенні 7:3 та передані на навчання кожній з моделей.

3.2 Класифікація БПЛА через ResNet-50

Запустимо навчання моделі ResNet-50 з оптимізатором Adam та різними значеннями Learning Rate та подивимося, яку точність класифікації дає дана модель:

Таблиця 6

ResNet-50							
Кількість епох	Значення LR	Train Loss	Test Loss	Train Accuracy	Test Accuracy	Train Recall	Test Recall
5	0,001	0.0675	0.6740	97,66%	86,31%	85,71%	84,53%

10	0,001	0.0447	0.4785	97,98%	87,35%	88,60%	85,74%
20	0,001	0.0168	0.4886	98,39%	88,27%	92,44%	86,58%
5	0,0001	0.0098	0.4659	99,65%	90,77%	99,65%	90,76%
10	0,0001	0.0069	0.5061	99,70%	90,81%	99,70%	90,80%
20	0,0001	0.0062	0.5469	99,72%	90,84%	99,72%	90,83%
5	0,0005	0.1301	0.2803	92,49%	90,57%	92,49%	90,57%
10	0,0005	0.0691	0.3482	94,76%	91,01%	94,76%	91,00%
20	0,0005	0.0325	0.3057	96,49%	91,61%	96,49%	91,61%
5	0,00005	0.0058	0.2791	99,73%	94,58%	99,73%	94,57%
10	0,00005	0.0041	0.2965	99,75%	94,65%	99,75%	94,65%
20	0,00005	0.0035	0.2812	99,77%	94,66%	99,77%	94,65%

Із отриманих результатів навчання бачимо, що найкращі результати модель ResNet-50 дала з оптимізатором Adam та значенням learning rate = 0,00005 для 20 епох і мала точність класифікації на тестових даних рівну 94,66%, а на тренувальних 99,77%, що є хорошим показником.

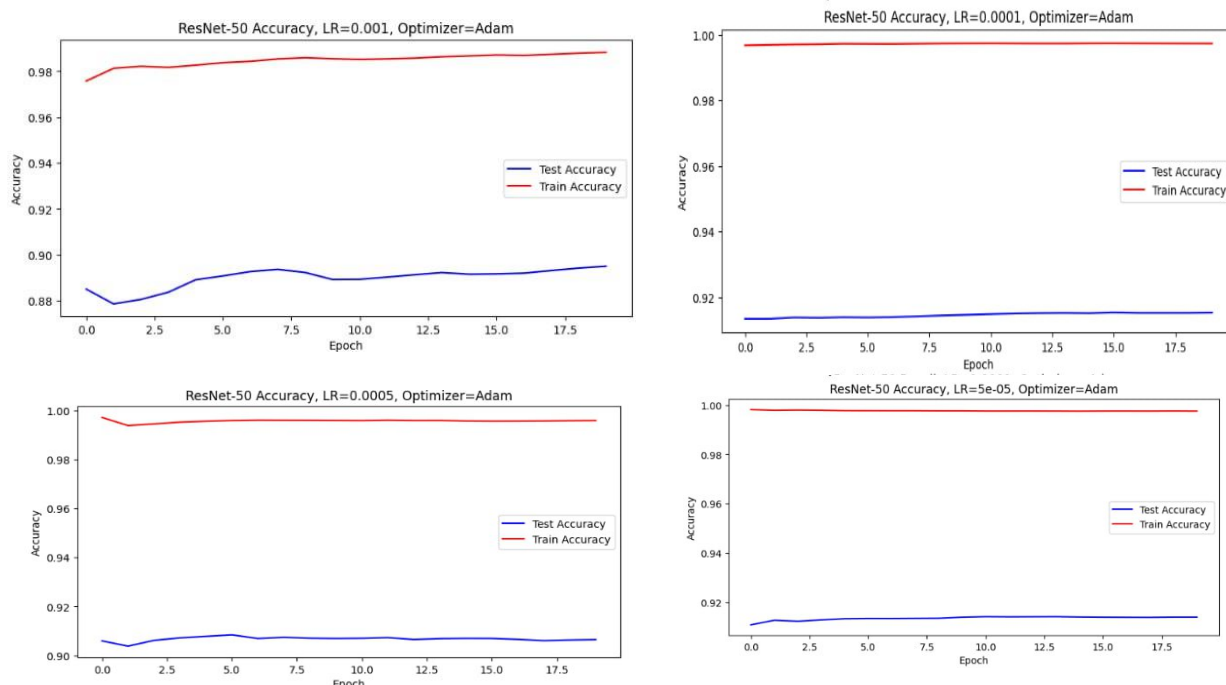


Рис 3.5-3.9 Графіки Accuracy для ResNet-50 в залежності від значення LR.

3.3 Класифікація БПЛА через MobileNetV1

Аналогічно до навчання моделі ResNet-50, запустимо модель MobileNetV1 з оптимізатором Adam та різними значеннями Learning Rate на навчання та подивимося, яку точність класифікації дає дана модель:

Таблиця 7

MobileNetV1							
Кількість епох	Значення LR	Train Loss	Test Loss	Train Accuracy	Test Accuracy	Train Recall	Test Recall
5	0,001	0.1563	0.2840	91,48%	91,56%	91,48%	91,55%
10	0,001	0.1063	0.2230	93,63%	92,08%	93,63%	92,08%
20	0,001	0.0580	0.3046	95,60%	92,20%	95,60%	92,20%

5	0,0001	0.0054	0.2937	99,60%	94,22%	99,60%	94,22%
10	0,0001	0.0042	0.3217	99,65%	94,25%	99,65%	94,25%
20	0,0001	0.0041	0.3667	99,70%	94,34%	99,70%	94,33%
5	0,0005	0.0227	0.4834	99,25%	93,07%	99,25%	93,06%
10	0,0005	0.0218	0.3420	99,21%	93,17%	99,21%	93,17%
20	0,0005	0.0044	0.3966	99,27%	93,17%	99,27%	93,17%
5	0,00005	0.0033	0.4278	99,76%	93,98%	99,76%	93,97%
10	0,00005	0.0031	0.4316	99,76%	94,01%	99,76%	94,01%
20	0,00005	0.0029	0.4633	99,75%	94,04%	99,75%	94,03%

Із результатів навчання бачим, що найкращі результати модель MobileNetV1 дала з оптимізатором Adam та значенням learning rate = 0,00005 для 20 епох і мала точність класифікації на тестових даних рівну 94,04%, і для тренувальних у свою чергу 99,75%.

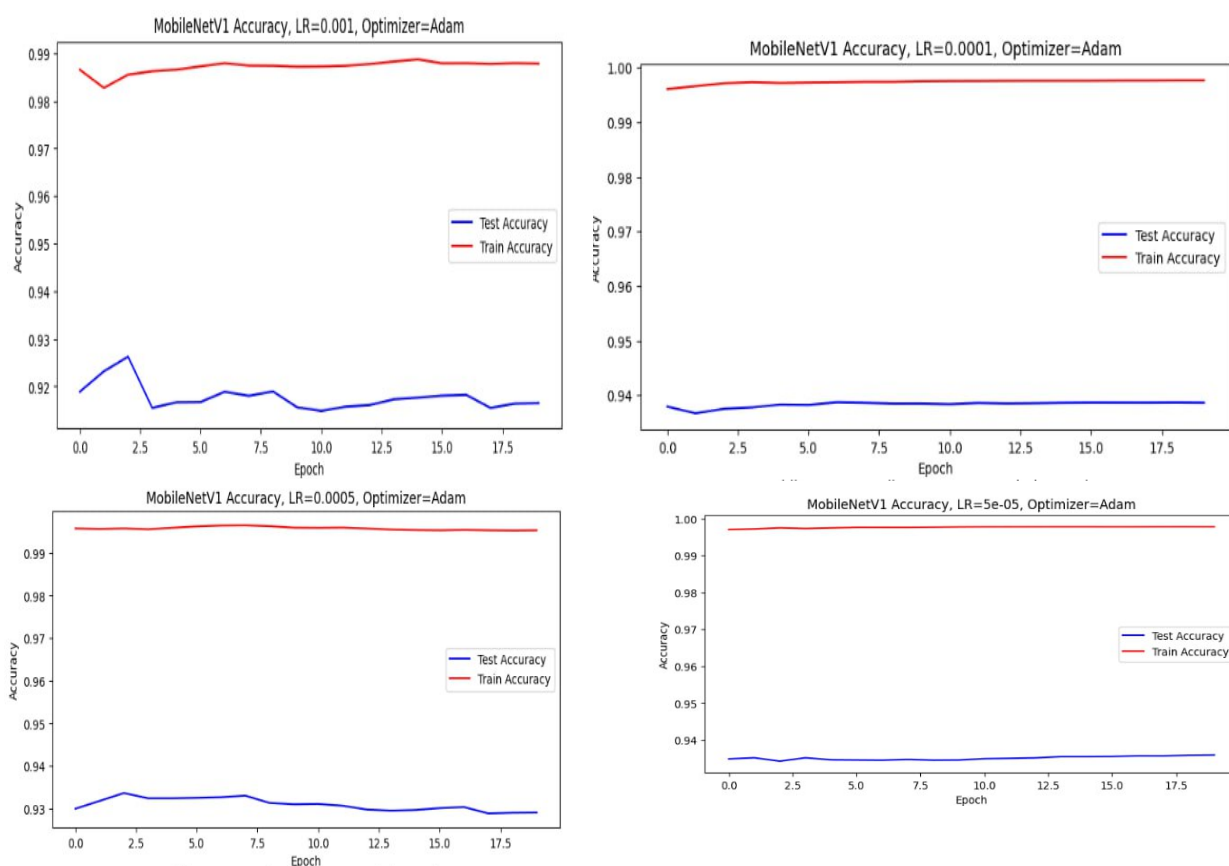


Рис 3.10-3.13 Графіки Accuracy для MobileNetV1 в залежності від значення LR.

3.4 Класифікація БПЛА через AlexNet

Навчимо модель AlexNet для класифікації даних БПЛА за формою і типами, застосувавши як оптимізатор Adam, і відповідно підбираючи різні значення Learning Rate навчання, та відповідно подивимось, яку точність класифікації дасть дана модель:

Таблиця 8

AlexNet							
Кількість епох	Значення LR	Train Loss	Test Loss	Train Accuracy	Test Accuracy	Train Recall	Test Recall
5	0,001	0.6088	0.6721	73,73%	76,85%	73,73%	76,85%
10	0,001	0.5243	0.5305	77,74%	78,22%	77,74%	78,22%
20	0,001	0.5147	0.4854	80,45%	80,05%	80,45%	80,04%
5	0,0001	0.2964	0.4167	89,34%	86,15%	89,34%	86,14%
10	0,0001	0.2252	0.5133	90,23%	86,96%	90,23%	86,95%
20	0,0001	0.1406	0.5999	91,56%	87,33%	91,56%	87,32%
5	0,0005	0.2830	0.4970	90,27%	86,05%	90,27%	86,04%
10	0,0005	0.2984	0.4535	90,79%	86,27%	90,79%	86,27%
20	0,0005	0.1850	0.6375	91,04%	86,66%	91,04%	86,65%
5	0,00005	0.0751	1.3277	96,27%	87,89%	96,62%	87,88%
10	0,00005	0.0527	2.2232	96,87%	87,91%	96,87%	87,91%
20	0,00005	0.0313	2.3052	97,62%	87,99 %	97,27%	87,99%

Із результатів класифікації для AlexNet, бачимо, що найкращу точність на тестових даних дали 20 епох, із значенням learning rate = 0,00005, і відповідно точність класифікації БПЛА на тренувальних даних склала 97,62%, на тестувальних 87,99%.

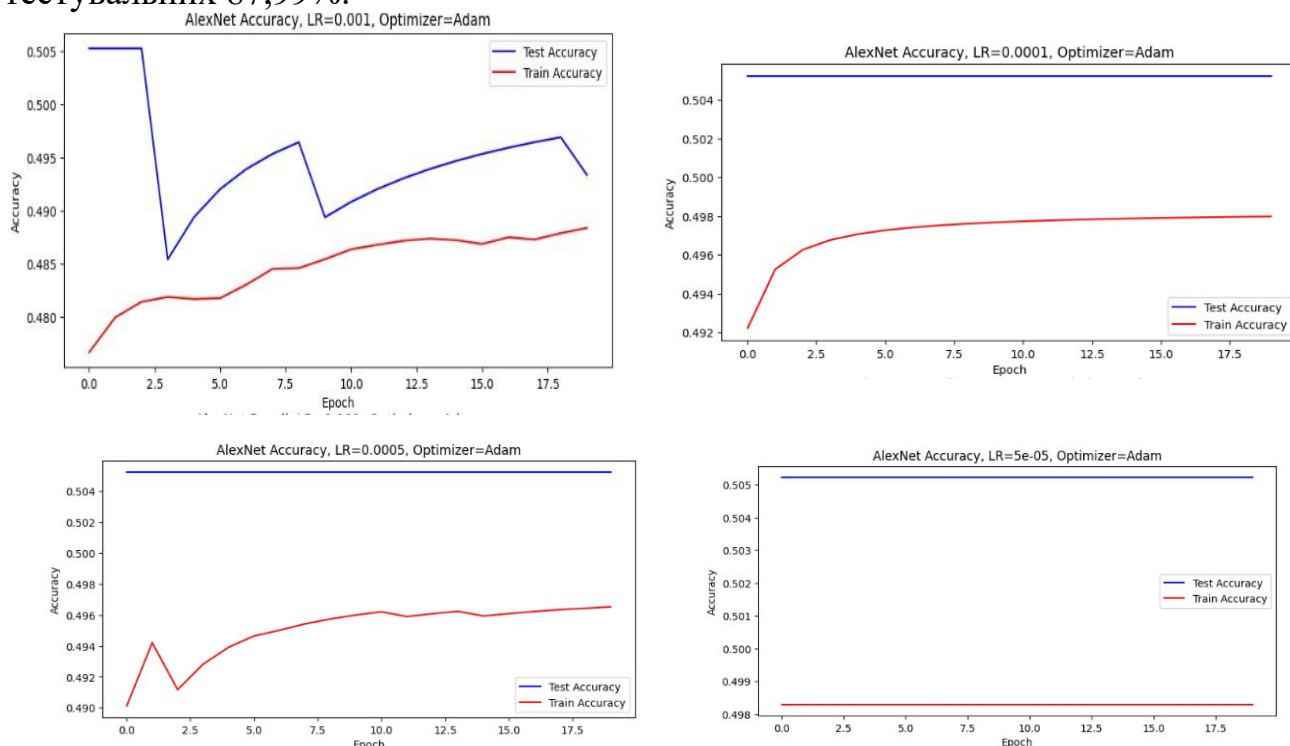


Рис 3.14-3.18 Графіки Ассурасу для AlexNet в залежності від значення LR.

3.5 Класифікація БПЛА через GoogleNet

Аналогічно як і зі всіма попередніми моделями провидимо навчання моделі на різній кількості епох, значенню Learning Rate, та подивимось наскільки хороші результати класифікації дасть GoogleNet:

Таблиця 9

GoogleNet							
Кількість епох	Значення LR	Train Loss	Test Loss	Train Accuracy	Test Accuracy	Train Recall	Test Recall
5	0,001	0.1209	0.2809	92,19%	89,71%	92,19%	89,71%
10	0,001	0.0687	0.4551	94,44%	89,60%	94,44%	89,60%
20	0,001	0.0461	0.3540	96,24%	89,68%	96,24%	89,68%
5	0,0001	0.0055	0.2679	99,68%	94,84%	99,68%	94,83;
10	0,0001	0.0054	0.2651	99,72%	94,96%	99,72%	94,96%
20	0,0001	0.0031	0.2952	99,75%	95,06%	99,75%	95,05%
5	0,0005	0.0205	0.3087	99,35%	93,20%	99,35%	93,20%
10	0,0005	0.0321	0.2803	99,26%	92,74%	99,26%	92,73%
20	0,0005	0.0099	0.2929	99,38%	93,31%	99,38%	93,30%
5	0,00005	0.0029	0.3082	99,78%	94,91%	99,78%	94,91%
10	0,00005	0.0023	0.3285	99,79%	94,96%	99,79%	94,96%
20	0,00005	0.0027	0.3391	99,80%	94,98%	99,80%	94,97%

Із результатів класифікації бачимо, що точність класифікації дронів для GoogleNet є найкращою для 20 епох, і значення learning rate = 0,0001 і відповідно складає 99,75% для тренувальних даних та 95,06 % для тестувальних.

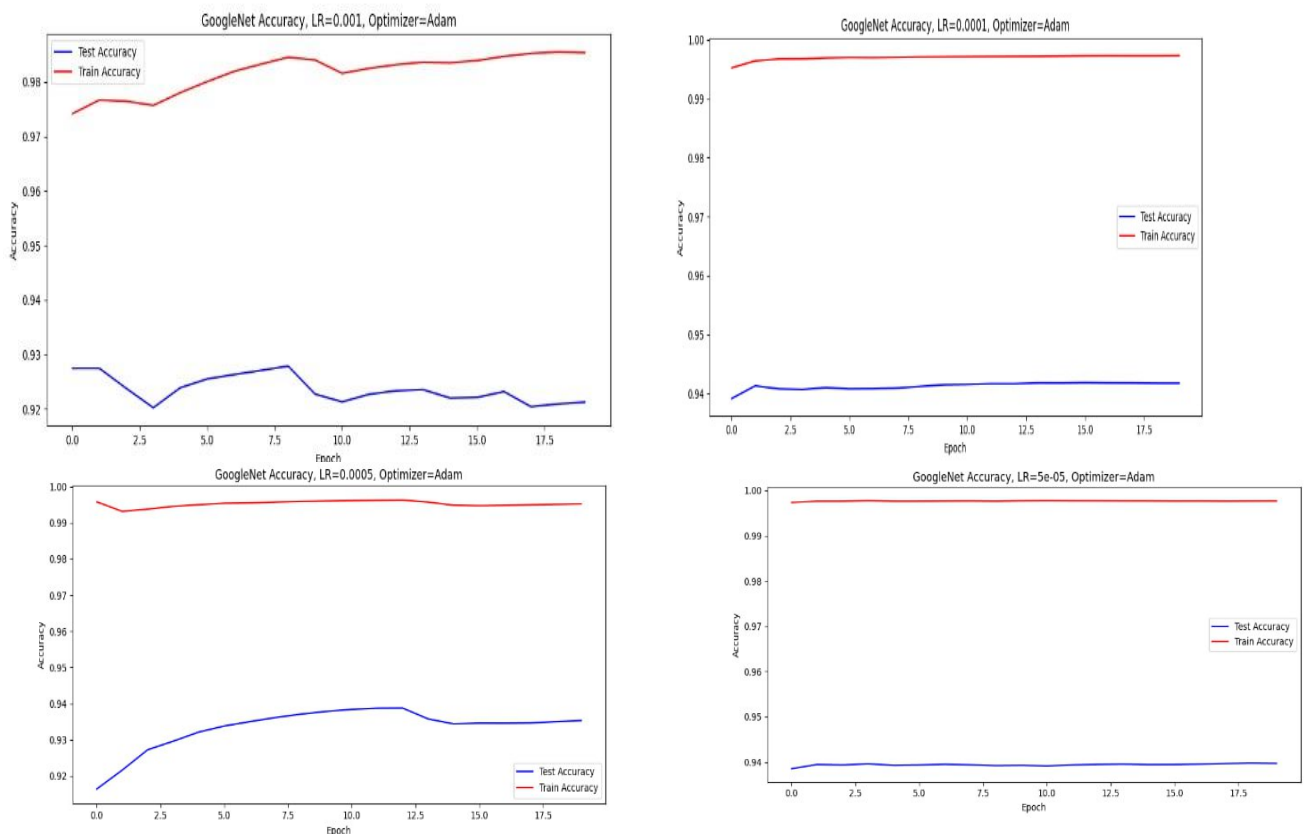


Рис 3.19-3.22 Графіки Ассурасу для GoogleNet в залежності від значення LR

3.6 Співставлення та порівняння результатів моделей

На основі усіх отриманих в пунктах 3.2-3.5 даних з експериментів про результати класифікації кожної з претренованих моделей, проведемо порівняльну характеристику усіх результатів, взявши найкращі з них для кожної моделі з конкретним значенням Learning Rate, та порівнявши їх і тим самим визначивши, яка модель з яким значенням Learning Rate є найкращою, і може бути використана для класифікації БПЛА за типами реальних системах.

Таблиця 10

Модель/LR	Train Accuracy	Test Accuracy	Train Recall	Test Recall
ResNet-50/0,001	98,39%	88,27%	92,44%	86,58%
ResNet-50/0,0001	99,72%	90,84%	99,72%	90,83%
ResNet-50/0,0005	96,49%	91,61%	96,49%	91,61%
ResNet-50/0,00005	99,77%	94,66%	99,77%	94,65%
MobileNetV1/0,001	95,60%	92,20%	95,60%	92,20%
MobileNetV1/0,0001	99,70%	94,34%	99,70%	94,33%
MobileNetV1/0,0005	99,27%	93,17%	99,27%	93,17%
MobileNetV1/0,00005	99,75%	94,04%	99,75%	94,03%
AlexNet/0,001	80,45%	80,05%	80,45%	80,04%
AlexNet/0,0001	91,56%	87,33%	91,56%	87,32%
AlexNet/0,0005	91,04%	86,66%	91,04%	86,65%
AlexNet/0,00005	97,62%	87,89%	97,62%	87,88%
GoogleNet/0,001	96,24%	89,68%	96,24%	89,68%
GoogleNet/0,0001	99,75%	95,06%	99,75%	95,05%
GoogleNet/0,0005	99,38%	93,31%	99,38%	93,30%
GoogleNet/0,00005	99,80%	94,98%	99,80%	94,97%

Отже, порівнявши усі результати, ми бачимо, що найкращу точність серед усіх запропонованих моделей із значеннями learning rate = 0,001, має MobileNetV1. В свою чергу для значень learning rate = 0,0001 найкращою моделлю для тестових даних є GoogleNet. Для значення learning rate = 0.0005 є відповідно GoogleNet. І на кінець, для значення learning rate = 0,00005 найкращі результати показує знову GoogleNet.

Звідси при окремому порівнянні даних точностей між собою і різними значеннями learning rate найкращою моделлю по значенню усіх метрик для тестових даних є як видно з таблиці 11, GoogleNet із значенням learning rate рівним 0,0001:

Таблиця 11

Модель	Train Accuracy	Test Accuracy	Train Recall	Test Recall
MobileNetV1/0,001	95,60%	92,20%	95,60%	92,20%
GoogleNet/0,0001	99,75%	95,06%	99,75%	95,05%
GoogleNet/0,0005	99,38%	93,31%	99,38%	93,30%
GoogleNet/0,00005	99,80%	94,98%	99,80%	94,97%

Отже, із результатів експериментів та порівняння точності класифікації із різними значеннями learning rate ми виявили, що найкращим поєднанням для

даного набору даних для класифікації БПЛА за формою та типом є GoogleNet із значенням $\text{learning rate} = 0,0001$.

Тепер аби наочно удостоверитися в отриманих результатах, спробуємо вивести фотографії із різними типами БПЛА на них, та подивитися до якого класу буде їх класифікувати найкраща обрана модель:

GoogleNet Real Class: MonoWing, Predicted Class: MonoWing



GoogleNet Real Class: MonoWing, Predicted Class: MonoWing



GoogleNet Real Class: QuadcopterWing, Predicted Class: QuadcopterWing



GoogleNet Real Class: QuadcopterWing, Predicted Class: QuadcopterWing

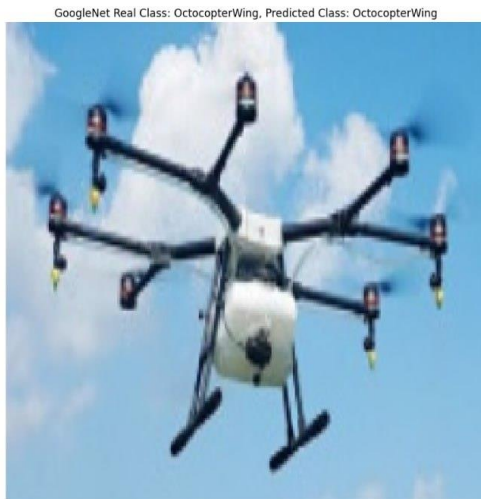


GoogleNet Real Class: HexacopterWing, Predicted Class: HexacopterWing



GoogleNet Real Class: HexacopterWing, Predicted Class: HexacopterWing





Із отриманих результатів для реальних зображень, про існування яких модель нічого не знала, бачим що результати класифікації підтверджують, що модель добре навчена, а отже GoogleNet дійсно є найкращою претренованою моделлю серед усіх наведених раніше для даної проблематики.

Висновки до розділу

У процесі роботи із претренованими моделями ResNet-50, MobileNetV1, AlexNet, GoogleNet, було проведено ряд експериментів із різними значеннями learning rate оптимізатора Adam, аби визначити, яка модель дає найкращі результати класифікації для проблематики класифікації БПЛА за формою та типом, в результаті чого було виявлено, що найкращим з точки зору класифікації є CNN мережа GoogleNet із значенням learning rate рівним 0,0001. Вона давала найкращі результати для метрик accuracy та recall, і досягала 95,06% точності класифікації об'єктів на тестових даних.

Окрім цього, було проведено перевірку моделі та вивід проміжних результатів її роботи, що показало точність її роботи на реальному прикладі.

РОЗДІЛ 4. ОБГОВОРЕННЯ РЕЗУЛЬТАТІВ ЕКСПЕРИМЕНТІВ

У розділі 3 було описано роботу різних претренованих моделей таких як ResNet-50, MobileNetV1, AlexNet та GoogleNet для задачі класифікації БПЛА за формою та типом до певного класу. Навчання моделей та їх порівняння для різних значень параметра learning rate оптимізатора Adam, було отримано їхні точності класифікації, на основі метрик accuracy та recall, а також значень loss. В результаті найкращою моделлю для даної проблематики виявилася GoogleNet із значенням learning rate рівним 0,0001, що дала точність accuracy на тренувальних даних рівною 99,75%, та на тестувальних 95,06%. В свою чергу метрика recall для даної моделі дала 99,75% для тренувальних та 95,05% для тестувальних даних, що є дуже хорошим результатом класифікації.

Що ж до інших моделей таких, як ResNet-50, MobileNetV1, AlexNet, то вони дали дещо гірші результати за GoogleNet, так найгіршою моделлю класифікації виявилася AlexNet із значенням learning rate 0,001, яка давала 80,45% accuracy для тренувальних даних, та 80,05% accuracy для тестувальних і відповідно 80,45% recall для тренувальних та 80,04% для тестувальних даних.

В свою чергу альтернативою по якості класифікації даних після GoogleNet із значенням learning rate рівним 0,0001, може виступати модель MobileNetV1 із значенням learning rate 0,001, яка дала точність accuracy на тренувальних даних 95,60% та 92,20% на тестувальних, і в свою чергу recall для тренувальних і тестувальних даних рівний 95,60% та 92,20%.

Загалом, для можливого подальшого аналізу даної проблематики і підбору найкращої моделі серед наведених чи пошуку іншої не наведеної в дані роботі моделі, можна також застосувати підбір найбільш оптимального оптимізатора, а також функції втрат, оскільки в даній роботі для вимірювання втрат під час навчання використовувалася CrossEntropyLoss функція.

Додатково, важливо розглянути можливість адаптації даних моделей для роботи з іншими типами БПЛА та розміром зображень цих дронів. Можливо, що деякі моделі будуть ефективнішими для великих зображень, тоді як інші можуть краще справлятися з малими зображеннями, так та ж сама GoogleNet добре працює лише із зображеннями 224x224 пікселі, проте на зображеннях більшої розмірності потенційно інші моделі могли б дати кращі результати.

У подальших дослідженнях також можна розглядати можливість ансамблювання моделей для покращення точності класифікації. Ансамблювання дозволить об'єднати декілька моделей для отримання більш точного та стійкого результату.

Загалом, результати дослідження свідчать про потенціал подальшого розвитку та оптимізації моделей для класифікації БПЛА, і дослідникам рекомендується продовжити роботу у цьому напрямку, враховуючи вищезгадані аспекти та можливі напрямки покращення.

Висновки до розділу

У розділі 4, було проведено аналіз та інтерпритацію отриманих раніше результатів та наведено можливі варіанти подальшого дослідження даної проблематики класифікації БПЛА за типами та формою. Окрім цього під час обговорення було наведено найкращий та найгірший із можливих поєднань моделей.

Так, за результатами експериментів, найкращою моделлю виявилася GoogleNet з параметром learning rate 0,0001, що показала високу точність на тренувальних і тестувальних даних. Також було проведено порівняння інших моделей, таких як ResNet-50, MobileNetV1 і AlexNet. Вони показали дещо гірші результати порівняно з GoogleNet. І у свою чергу AlexNet була найменш ефективною моделлю в цьому контексті.

Для подальшого аналізу і покращення результатів рекомендується розглянути наступні аспекти:

- Підбір оптимального оптимізатора та функції втрат для покращення результатів моделі.
- Адаптація моделей для роботи з іншими типами БПЛА та розмірами зображень.
- Розгляд можливості ансамблювання обраних моделей для отримання ще більш точного результату класифікації.

Ці напрямки можуть допомогти подальшому розвитку і оптимізації моделей для класифікації БПЛА. Рекомендується продовжити дослідження у цьому напрямку з урахуванням вищезгаданих аспектів та можливих шляхів покращення результатів.

РОЗДІЛ 5. ВИСНОВКИ

У ході курсової роботи було проведено дослідження класифікації БПЛА за формою та типом, аби дати змогу швидко та ефективно реагувати на ворожі та союзні БПЛА і випадково їх збити чи не збити. Так в розділі “ВСТУП” було чітко сформовано мету дослідження, та проведено детальний аналіз обраної проблематики і відповідно визначено об’єкт та предмет дослідження.

У розділі 1 проведено аналіз літературних джерел за останні 5 років аби виявити які підходять та методи використовуються для класифікації об’єктів на зображенні, та які з них є актуальними для обраної проблематики. Окрім цього в розділі 1, було на основі обраного методу згорткових нейронних мереж додатково проаналізовано популярні та широко використовувані претреновані моделі CNN, і відповідно обрано декілька з них, із найбільш великим значенням точності класифікації, як основу для дослідження.

У розділі 2 було проведено огляд набору даних, який використовувався для подальшого навчання даних моделей, а також спосіб попередньої обробки і підготовки даних для найкращого навчання обраних моделей. Також розглянуто внутрішню архітектуру обраних моделей ResNet-50, MobileNetV1, AlexNet, GoogleNet, із детальним аналізом послідовності їх виконання, та роботу внутрішніх блоків при огляді архітектури мережі. Окрім цього, під час огляду вище згаданого набору даних, було обрано основні метрики для оцінки точності моделі такі як, Accuracy, Recall, Loss. У свою чергу в подальших експериментах вони використовувалися як критерій оцінки якості навчання моделей.

У розділі 3 було проведено ряд експериментів із застосуванням обраних моделей, та підбором різних значень learning rate для оптимізатора Adam, що використовувався при тренуванні і оцінці роботи моделей. За результатами експериментів, було визначено найбільш ефективну модель для обраної проблематики, і потенціал її подальшого застосування у реальних умовах. Окрім цього також було запущено найкращу модель на даних про які вона нічого не знала під час навчання і валідації, для перевірки якості класифікації обраної моделі, якою була GoogleNet із значенням learning rate 0.0001, що показало що модель дійсно справляється із поставленою задачею класифікації БПЛА за формою та типом.

Відповідно в розділі 4, було інтерпретовано результати дослідження аби узагальнити отримані результати. Окрім цього також було запропоновано варіанти ще більшого покращення отриманих результатів за рахунок ансамблювання отриманих моделей чи підбору найкращого оптимізатора під час подальших досліджень даної проблематики.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

- [1] K. Sanghvi, A. Aralkar, S. Sanghvi, I. Saha “A Survey on Image Classification Techniques”
https://www.researchgate.net/publication/348531979_A_Survey_on_Image_Classification_Techniques
- [2] A. O. Alaoui, O. E. Bahi, M. R. Fethi, O. Farhaoui, A. E. Allaoui, Y. Farhaoui “Pre-trained CNNs: Evaluating Emergency Vehicle Image Classification”
<https://dm.saludcyt.ar/index.php/dm/article/view/153/303>
- [3] S. M. Zainorzuli, S. A. Abdullah, H. Z. Abidin, F. A. Ruslan “Comparison Study on Convolution Neural Network (CNN) Techniques for Image Classification”
<https://ir.uitm.edu.my/id/eprint/63165/>
- [4] E. Prasetyo, R. Purbaningtyas, R. D. Adityo, N. Suciati, C. Faticah “Combining MobileNetV1 and Depthwise Separable convolution bottleneck with Expansion for classifying the freshness of fish eyes”
<https://www.sciencedirect.com/science/article/pii/S2214317322000026>
- [5] Набір даних “Fixed Wing UAV - Plane”
<https://www.kaggle.com/datasets/melihcan09/fixed-wing-uav-plane>
- [6] Набір даних “Fixed Wing UAV Dataset”
<https://www.kaggle.com/datasets/nyahmet/fixed-wing-uav-dataset>
- [7] Набір даних “Drone Dataset (UAV)”
<https://www.kaggle.com/datasets/dasmehdixtr/drone-dataset-uav>
- [8] Набір даних “UAV Detection Dataset”
<https://www.kaggle.com/datasets/nelyg8002000/uav-detection-dataset-images>
- [9] Набір даних, що використовувався для навчання і тестування моделі
https://drive.google.com/drive/folders/1VnyM8yOHZDY-Y-KRmc4i_QzPY9BkTgg?usp=drive_link
- [10] Зображення архітектури моделі ResNet-50 із статті
<https://medium.com/@nitishkundu1993/exploring-resnet50-an-in-depth-look-at-the-model-architecture-and-code-implementation-d8d8fa67e46f>
- [11] Зображення архітектури блоку Conv та ID із статті
https://www.researchgate.net/figure/ResNet50-Identity-Block-and-Conv-Block_fig2_363894256
- [12] Зображення архітектури моделі MobileNetV1 із статті
https://www.researchgate.net/figure/MobileNet-V1-architecture_fig3_376752661
- [13] Зображення архітектури моделі AlexNet із статті
<https://www.nomidl.com/deep-learning/introduction-to-alexnet-architecture/>
- [14] Зображення загальної архітектури моделі GoogleNet із статті
<https://medium.com/@AnasBrital98/googlenet-cnn-architecture-explained-inception-v1-225ae02513fd>
- [15] Зображення архітектури блоку Stem із статті
<https://medium.com/@AnasBrital98/googlenet-cnn-architecture-explained-inception-v1-225ae02513fd>
- [16] Зображення архітектури блоку Inception із статті
<https://medium.com/@AnasBrital98/googlenet-cnn-architecture-explained-inception-v1-225ae02513fd>

- [17] Зображення архітектури блоку Aux із статті
<https://medium.com/@AnasBrital98/googlenet-cnn-architecture-explained-inception-v1-225ae02513fd>
- [18] Код програмної реалізації, що використовувався для виконання курсової роботи
https://github.com/OrlykM/MachineLearningCourseWork/blob/master/UA Vs_classification.ipynb

ДОДАТОК А

Графіки функції втрат Loss усіх моделей ResNet-50, MobileNetV1, AlexNet, GoogleNet із різними значеннями параметра learning rate та оптимізатором Adam:

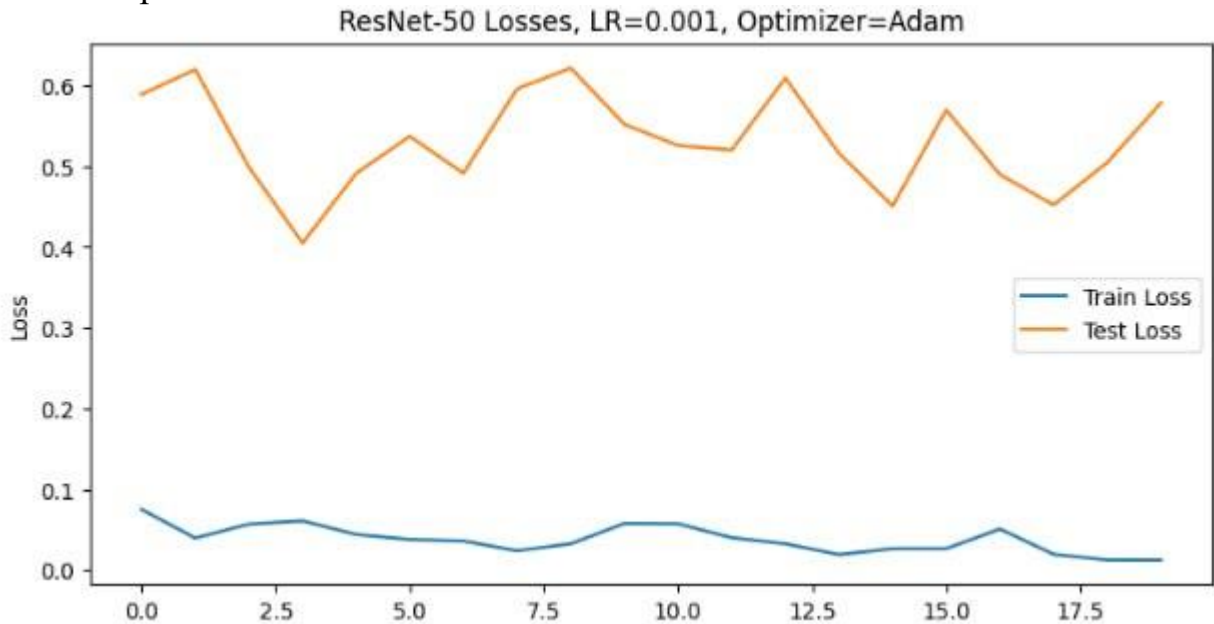


Рис А.1 Графік функції втрат ResNet-50 із значенням learning rate 0.001

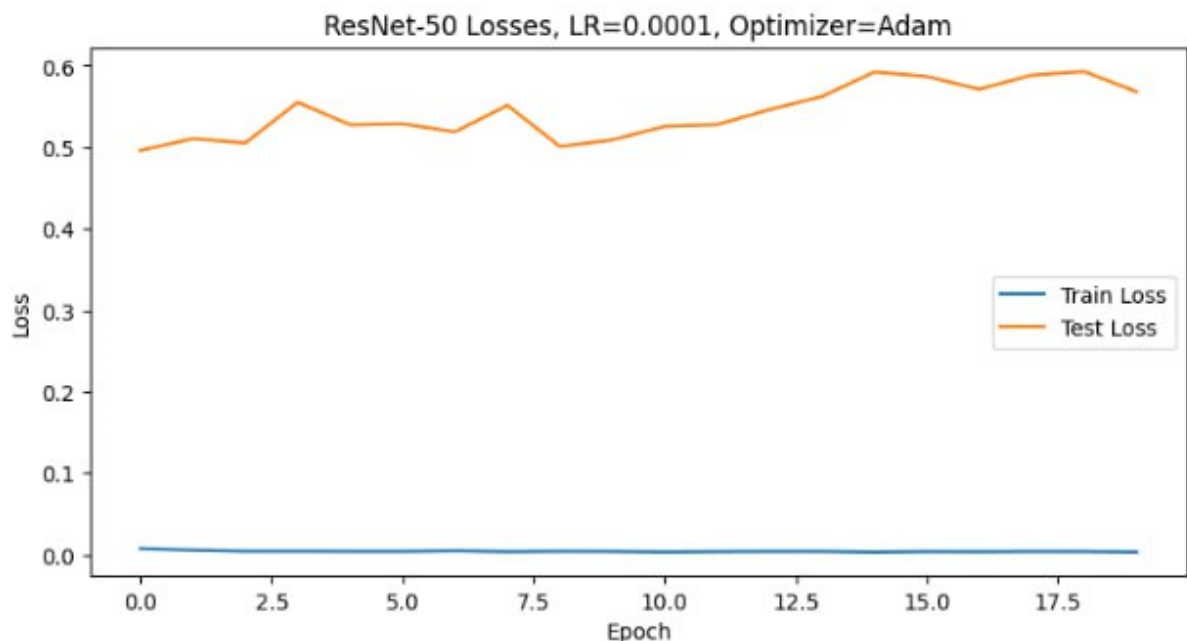


Рис А.2 Графік функції втрат ResNet-50 із значенням learning rate 0.0001

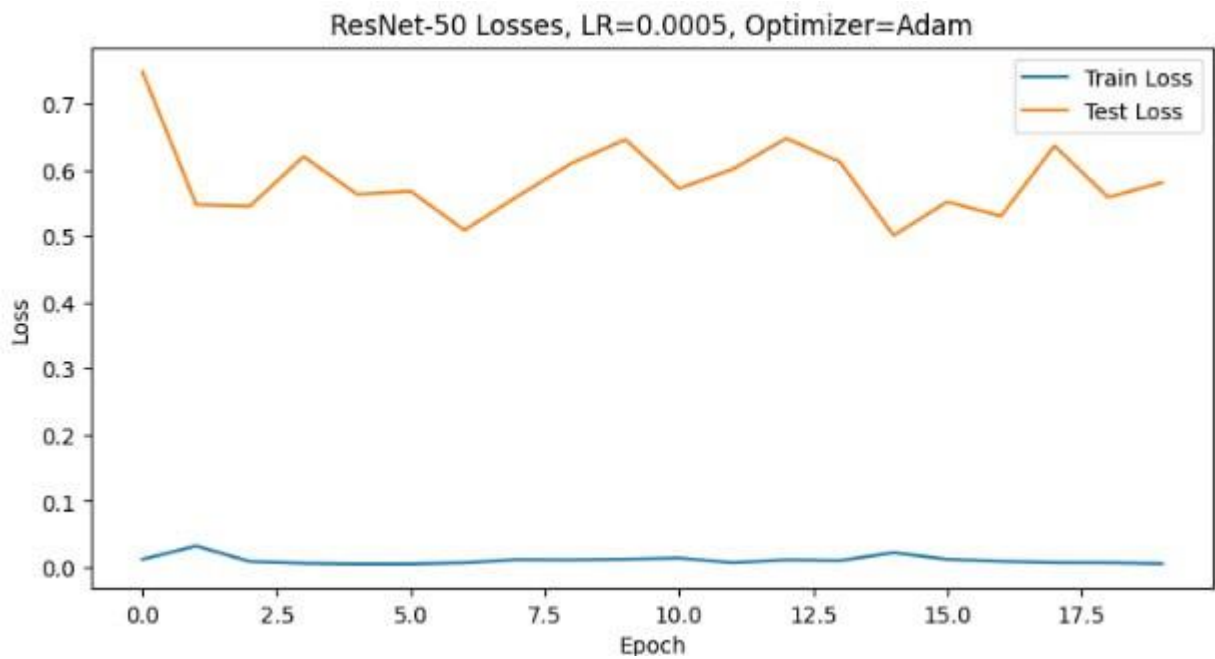


Рис А.3 Графік функції втрат ResNet-50 із значенням learning rate 0.0005

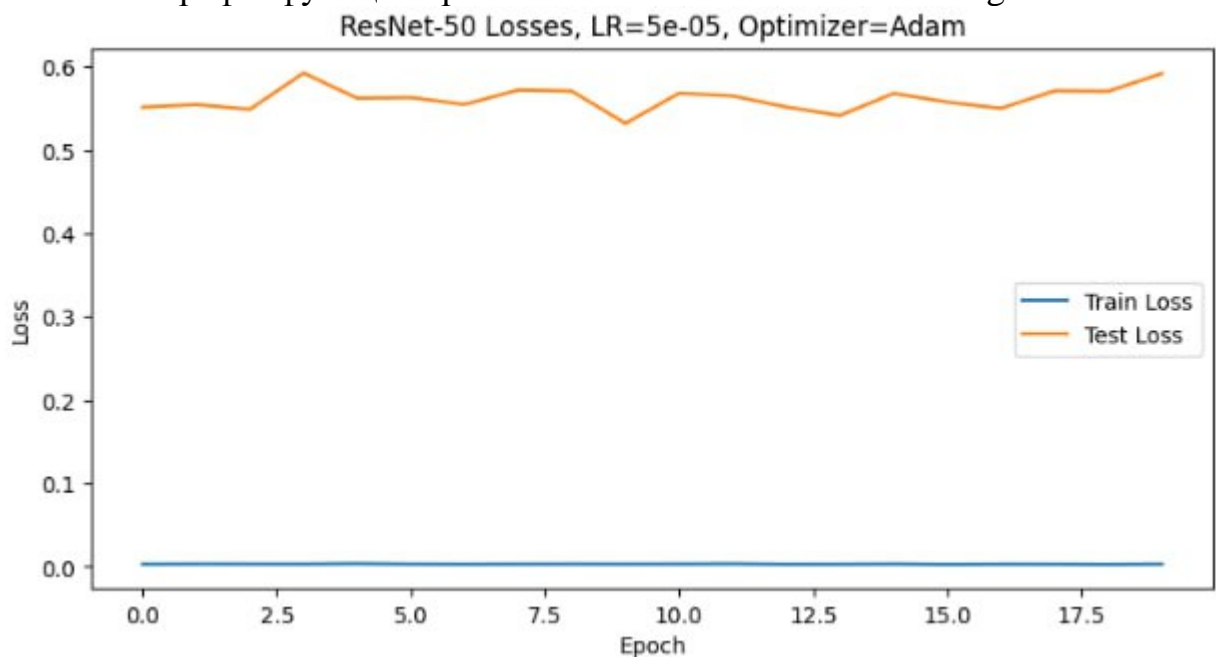


Рис А.4 Графік функції втрат ResNet-50 із значенням learning rate 0.00005

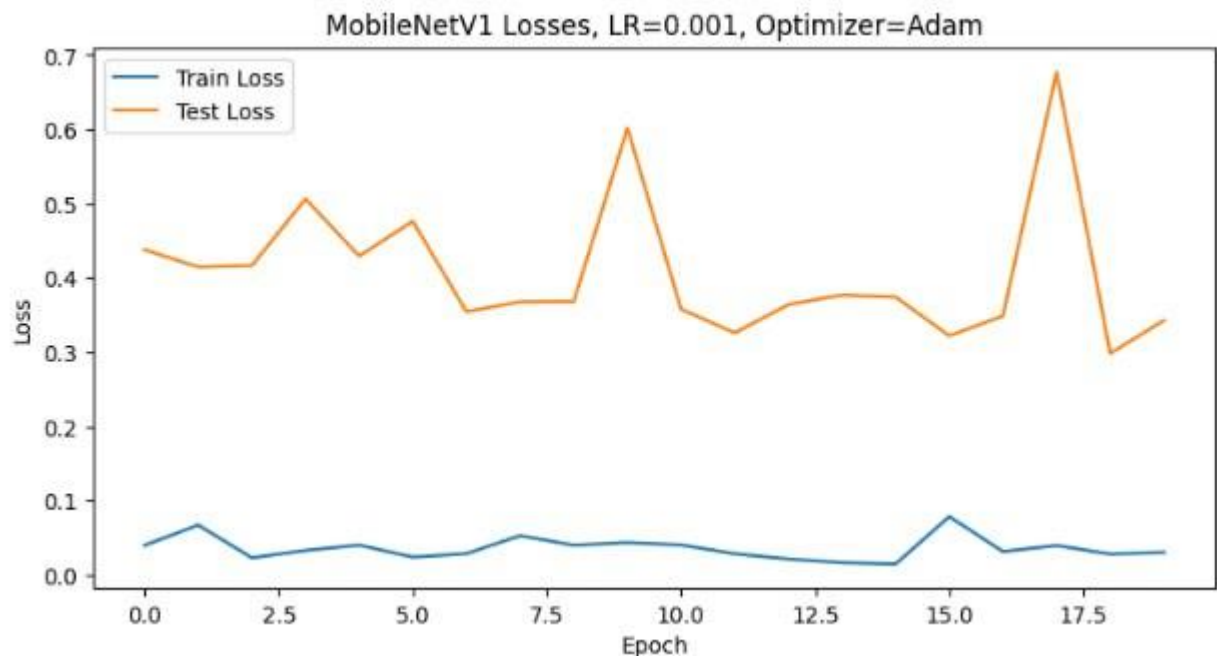


Рис А.5 Графік функції втрат MobileNetV1 із значенням learning rate 0.001

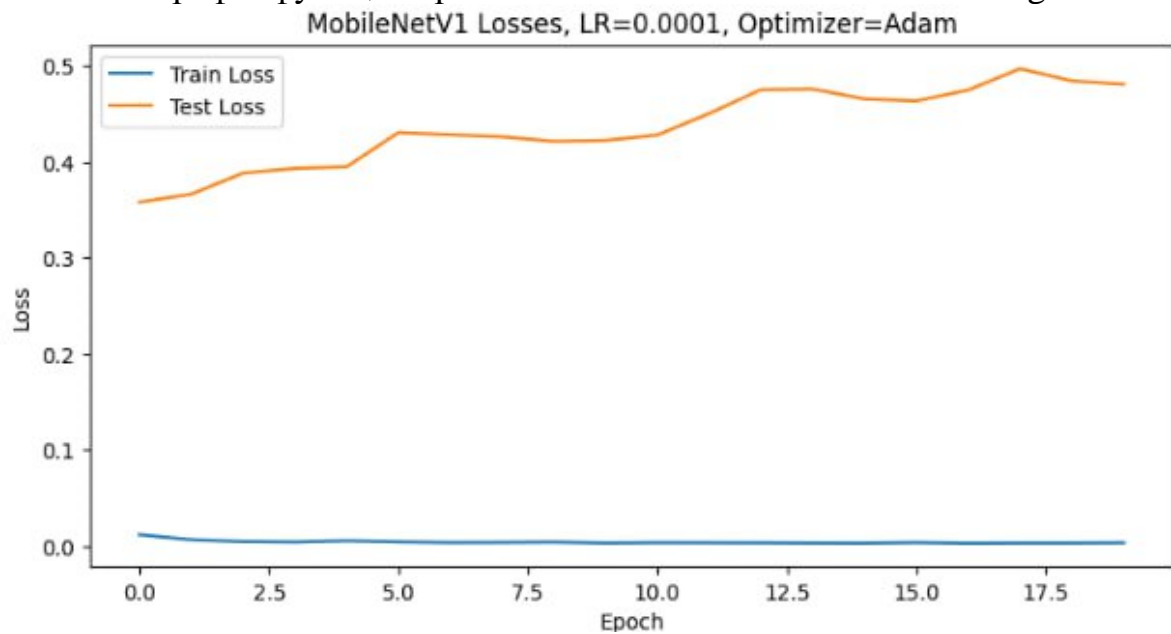


Рис А.6 Графік функції втрат MobileNetV1 із значенням learning rate 0.0001

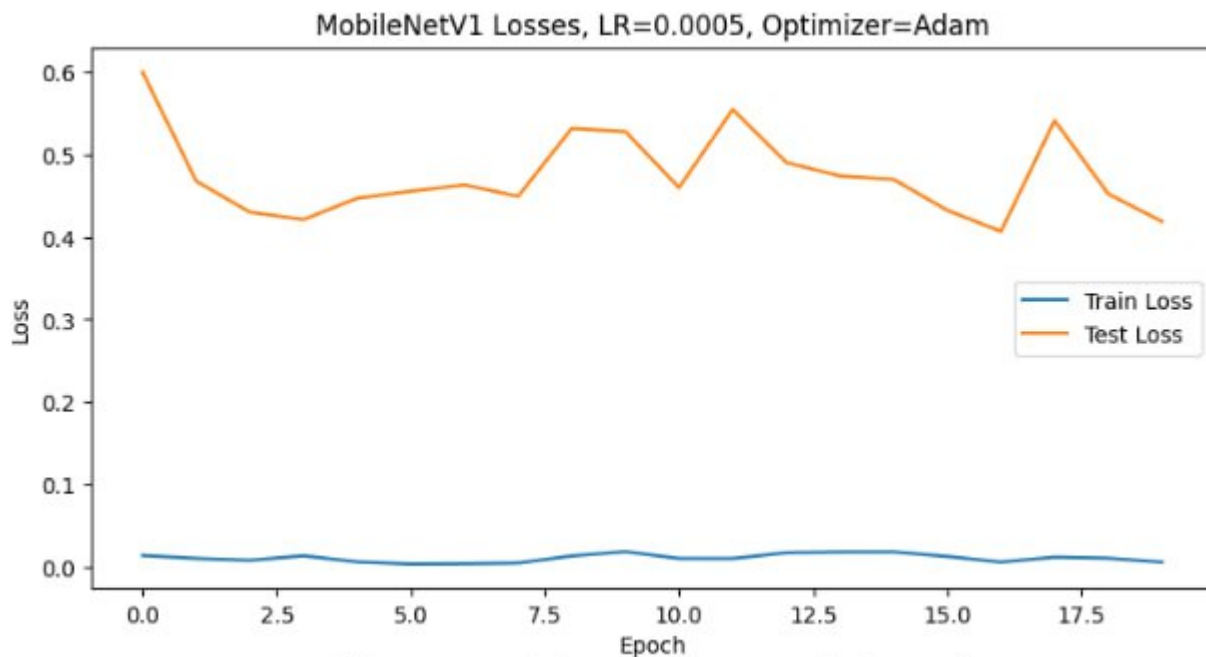


Рис А.7 Графік функції втрат MobileNetV1 із значенням learning rate 0.0005

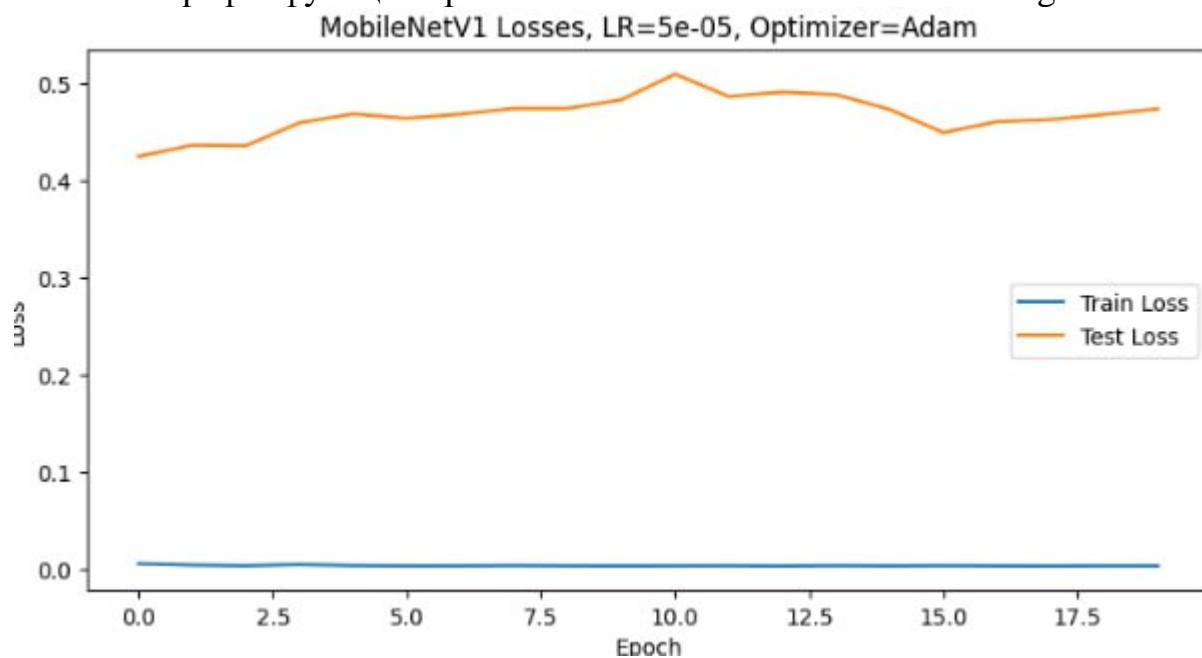


Рис А.8 Графік функції втрат MobileNetV1 із значенням learning rate 0.00005

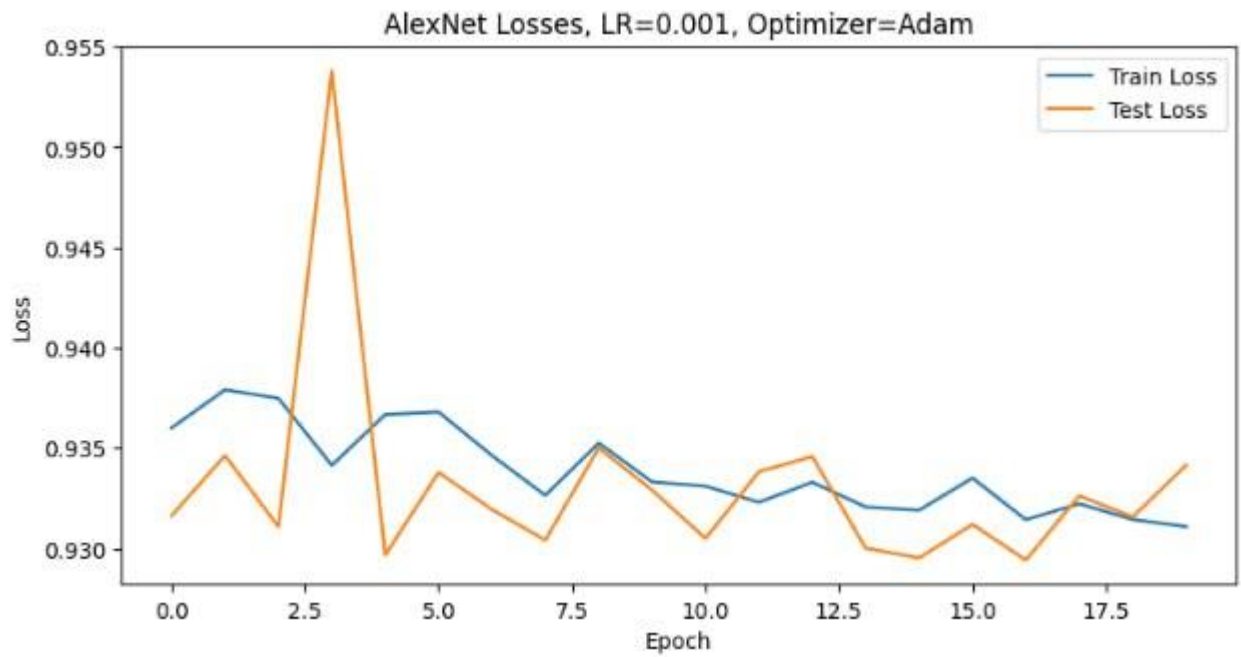


Рис А.9 Графік функції втрат AlexNet із значенням learning rate 0.001

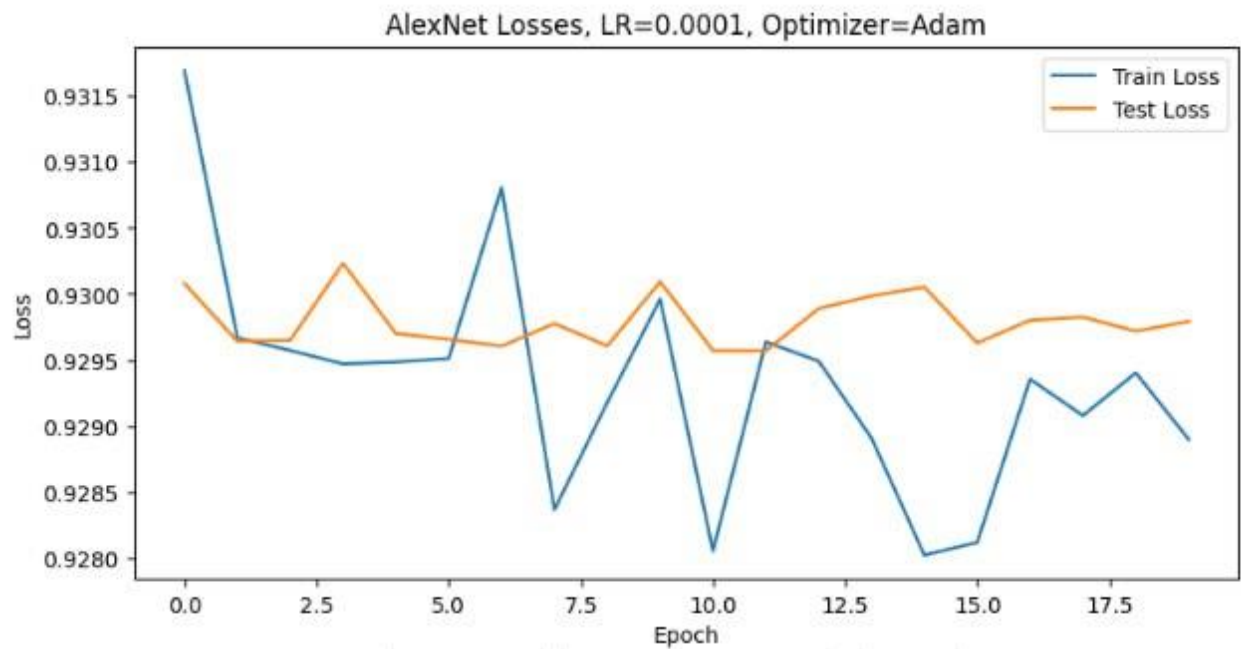


Рис А.10 Графік функції втрат AlexNet із значенням learning rate 0.0001

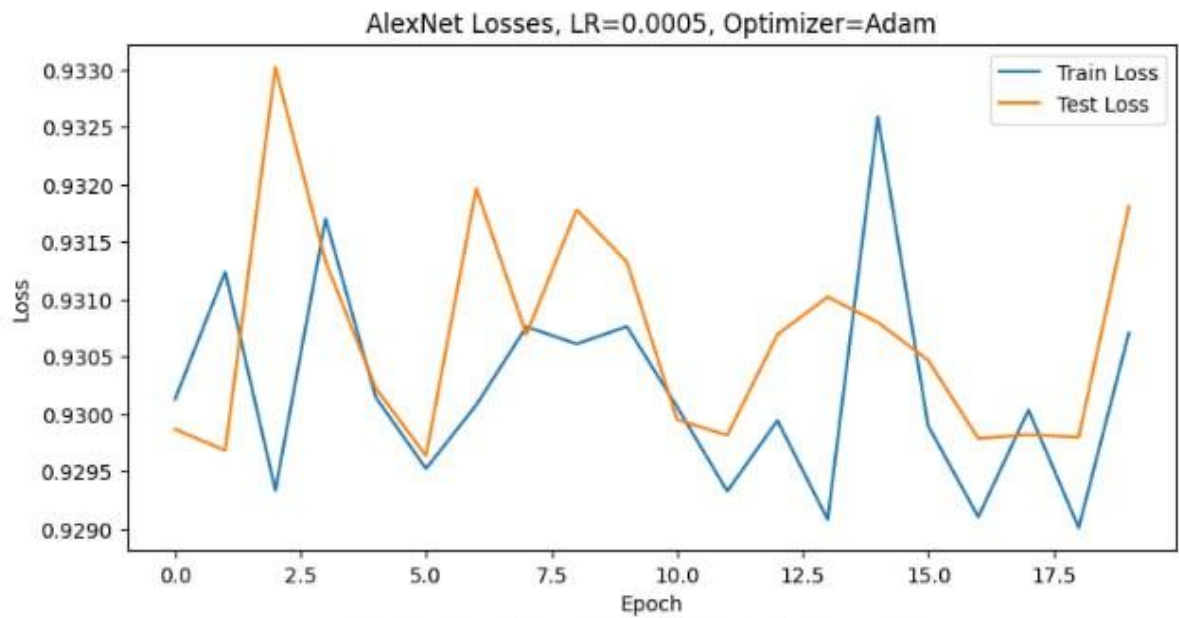


Рис А.11 Графік функції втрат AlexNet із значенням learning rate 0.0005

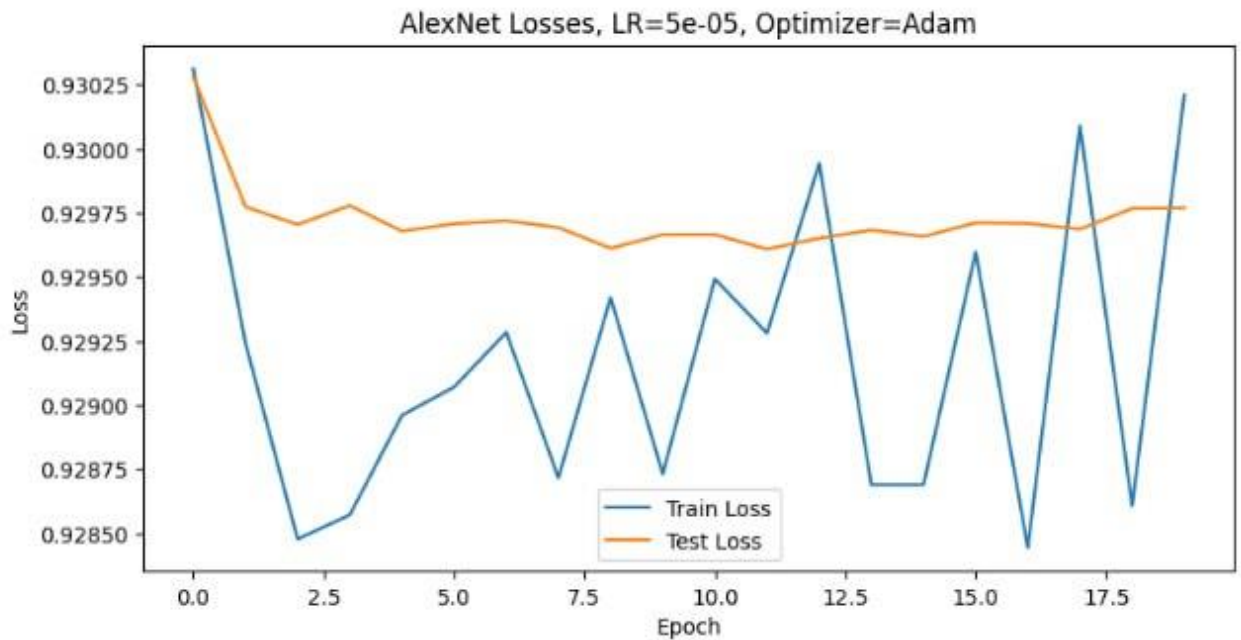


Рис А.12 Графік функції втрат AlexNet із значенням learning rate 0.00005

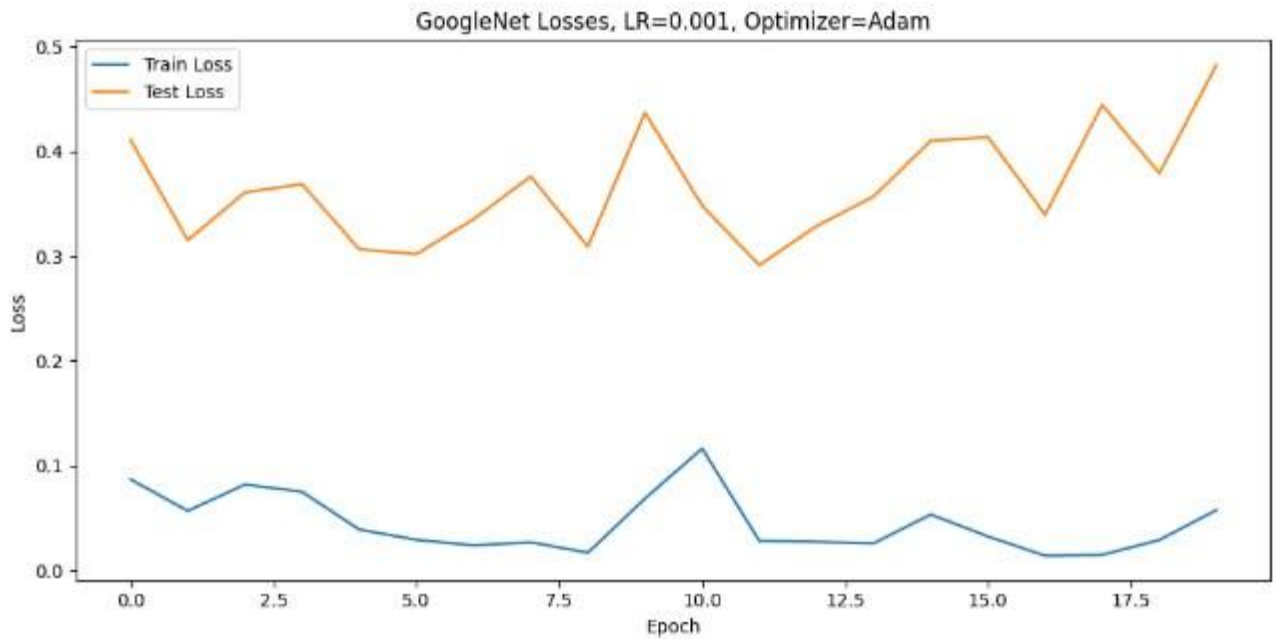


Рис А.13 Графік функції втрат GoogleNet із значенням learning rate 0.001

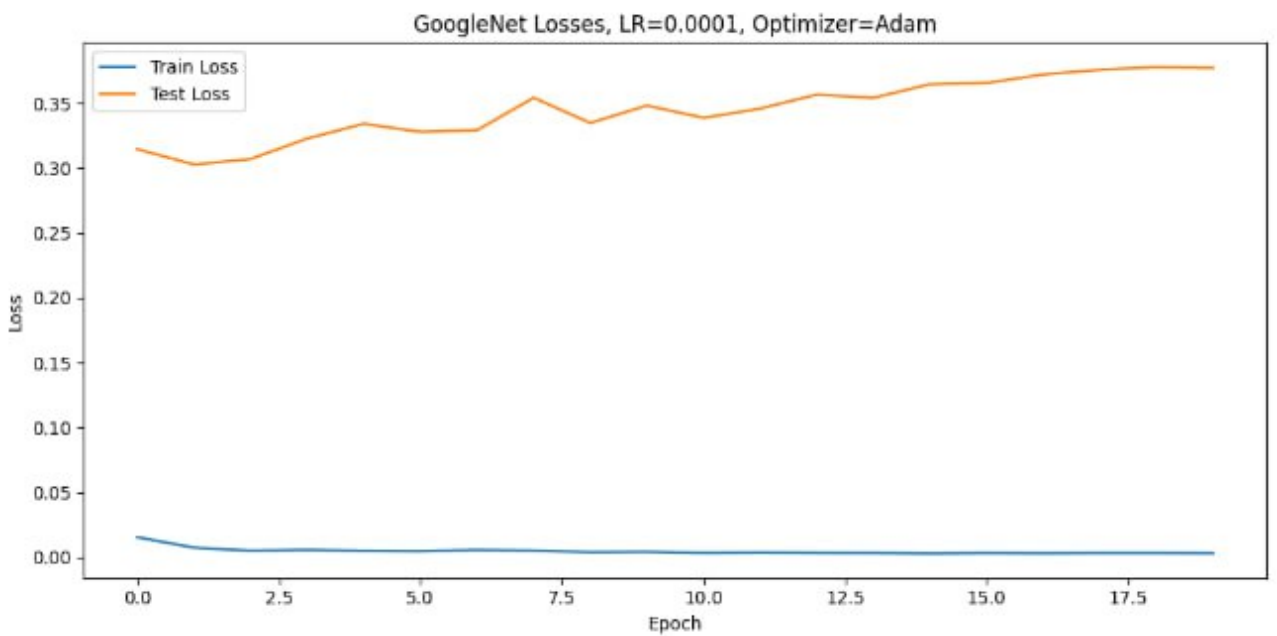


Рис А.14 Графік функції втрат GoogleNet із значенням learning rate 0.0001

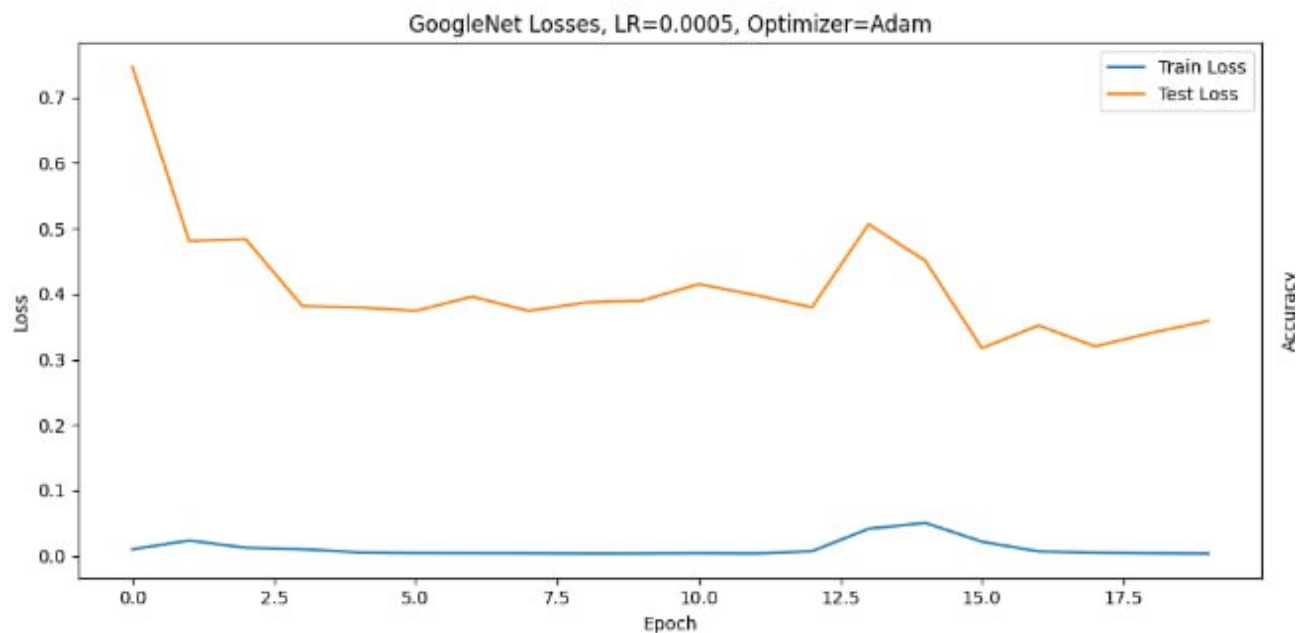


Рис А.15 Графік функції втрат GoogleNet із значенням learning rate 0.0005

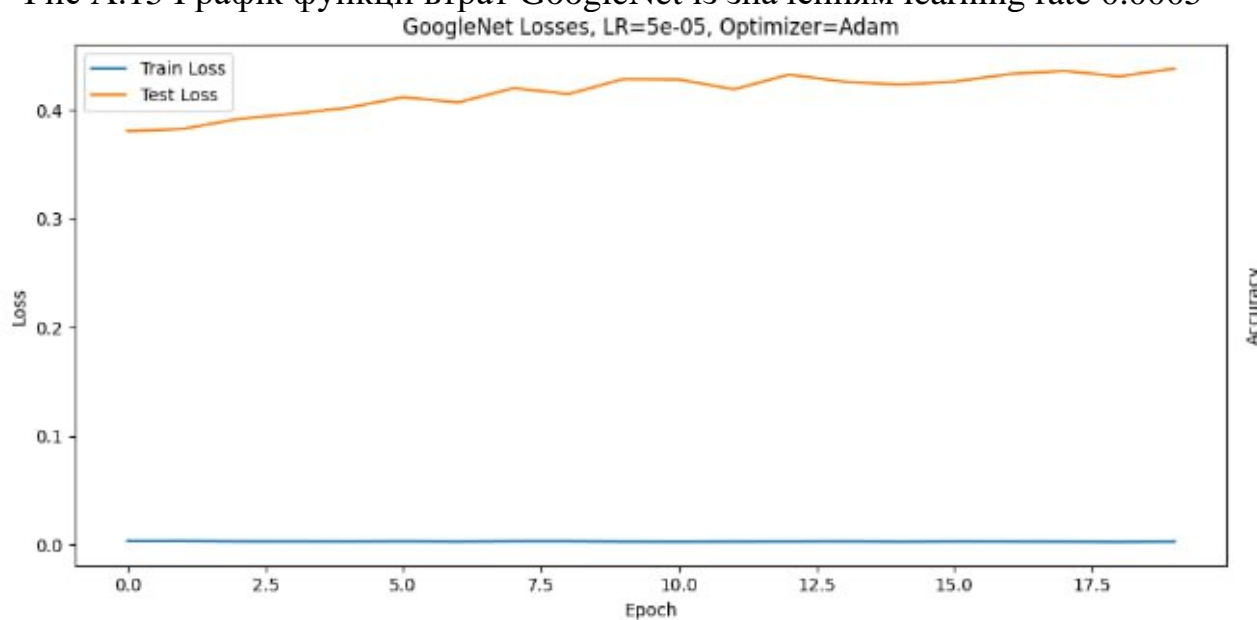


Рис А.16 Графік функції втрат GoogleNet із значенням learning rate 0.00005

ДОДАТОК Б

Графіки метрики Recall усіх моделей ResNet-50, MobileNetV1, AlexNet, GoogleNet із різними значеннями параметра learning rate та оптимізатором Adam:

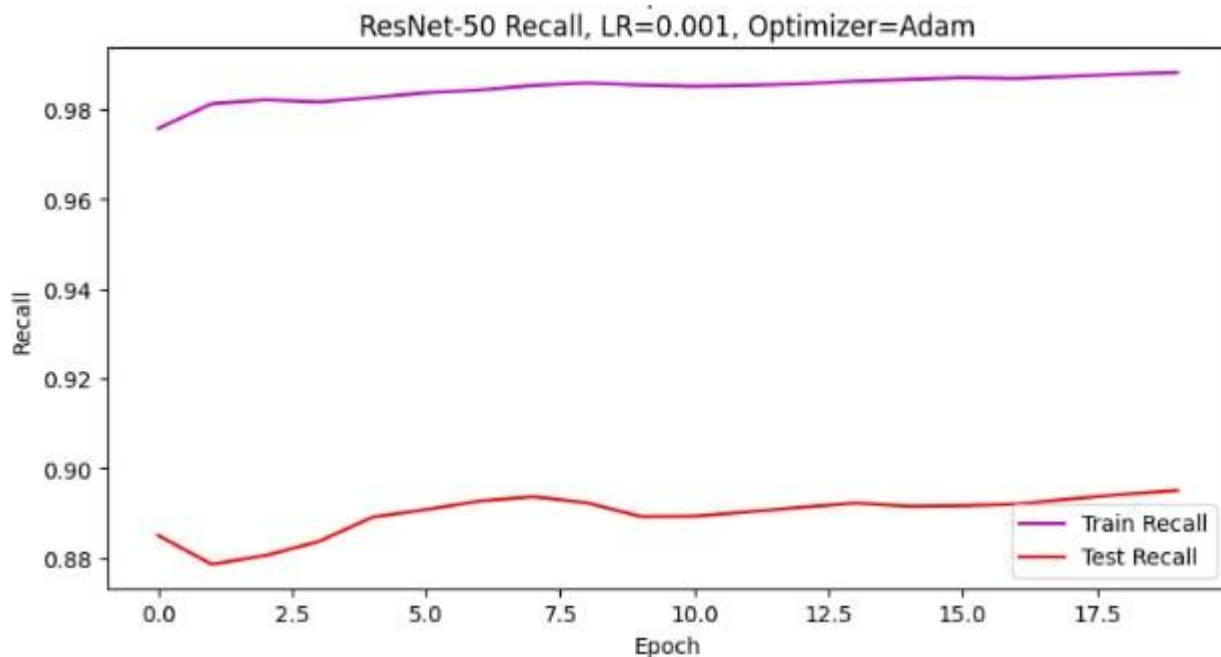


Рис Б.1 Графік метрики Recall ResNet-50 із значенням learning rate 0.001

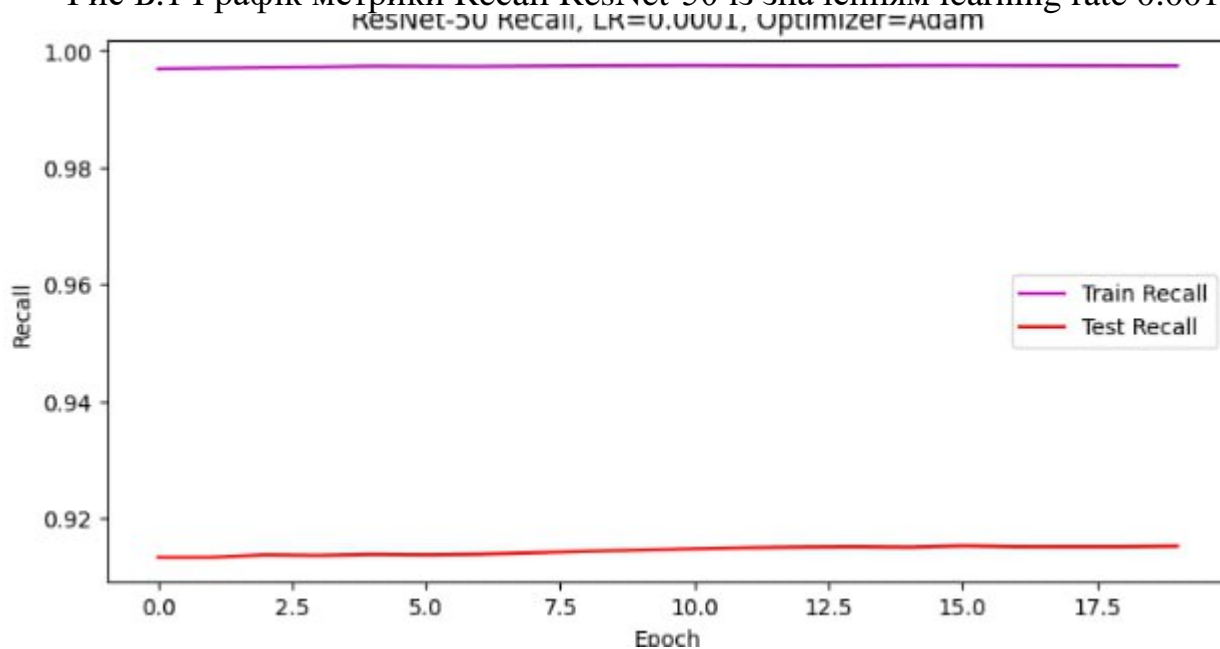


Рис Б.2 Графік метрики Recall ResNet-50 із значенням learning rate 0.0001

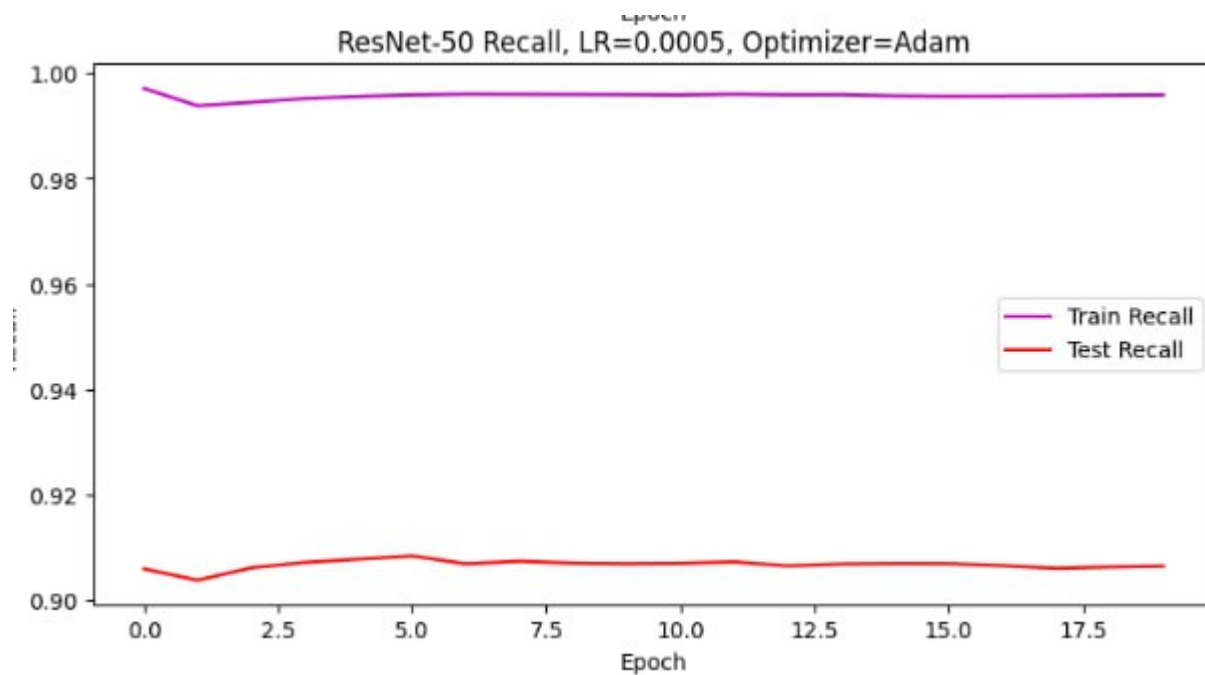


Рис Б.3 Графік метрики Recall ResNet-50 із значенням learning rate 0.0005

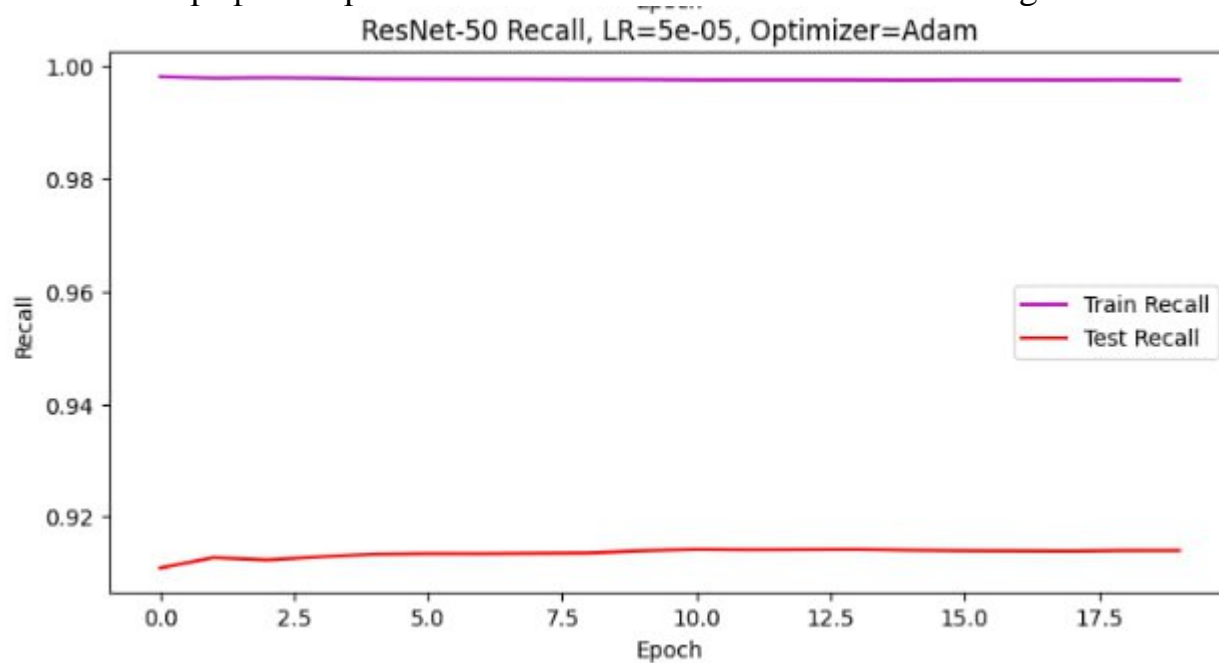


Рис Б.4 Графік метрики Recall ResNet-50 із значенням learning rate 0.00005

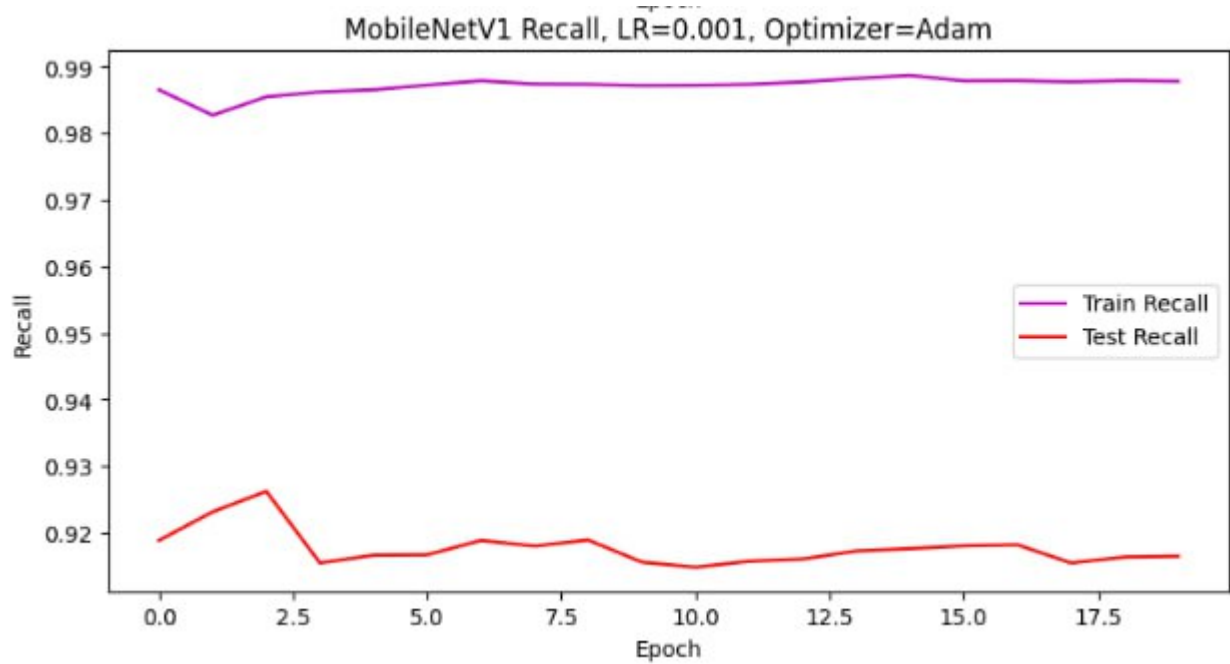


Рис Б.5 Графік метрики Recall MobileNetV1 із значенням learning rate 0.001

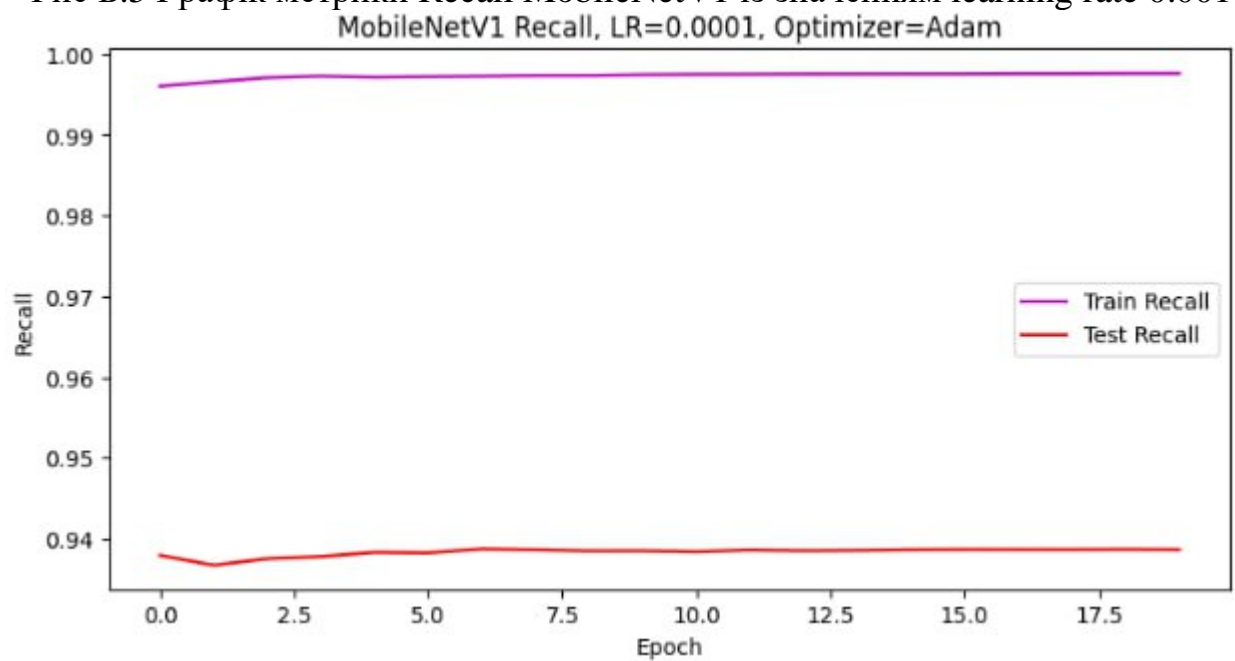


Рис Б.6 Графік метрики Recall MobileNetV1 із значенням learning rate 0.0001

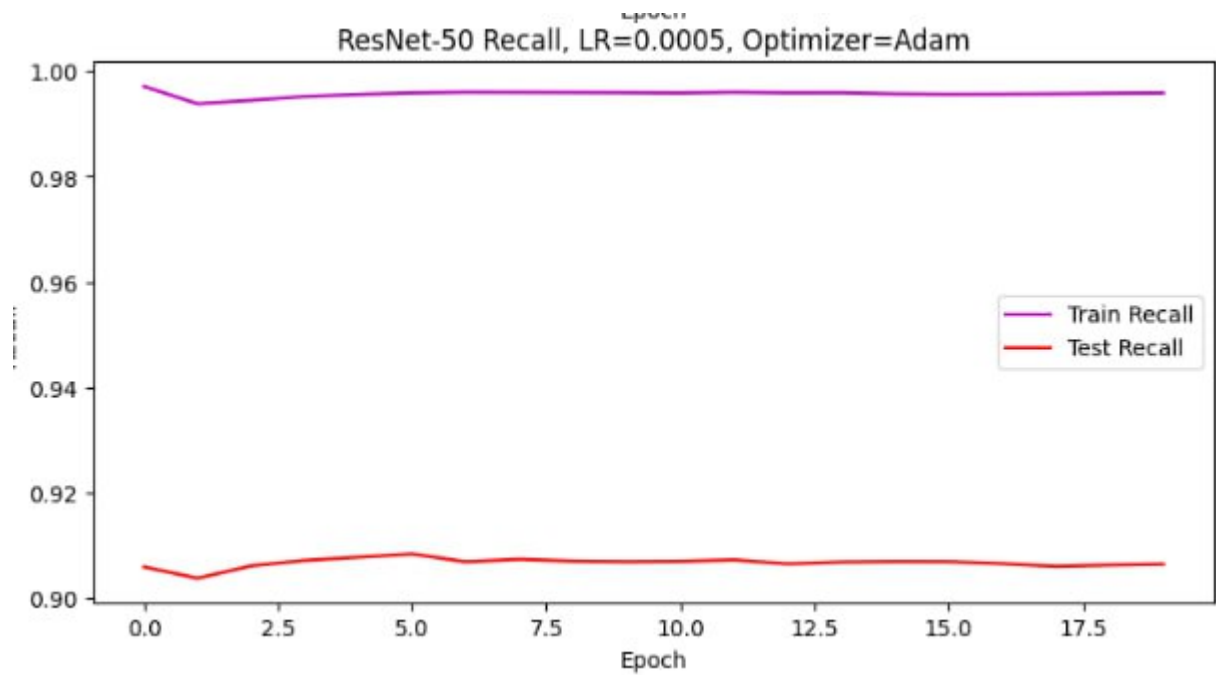


Рис Б.7 Графік метрики Recall MobileNetV1 із значенням learning rate 0.0005

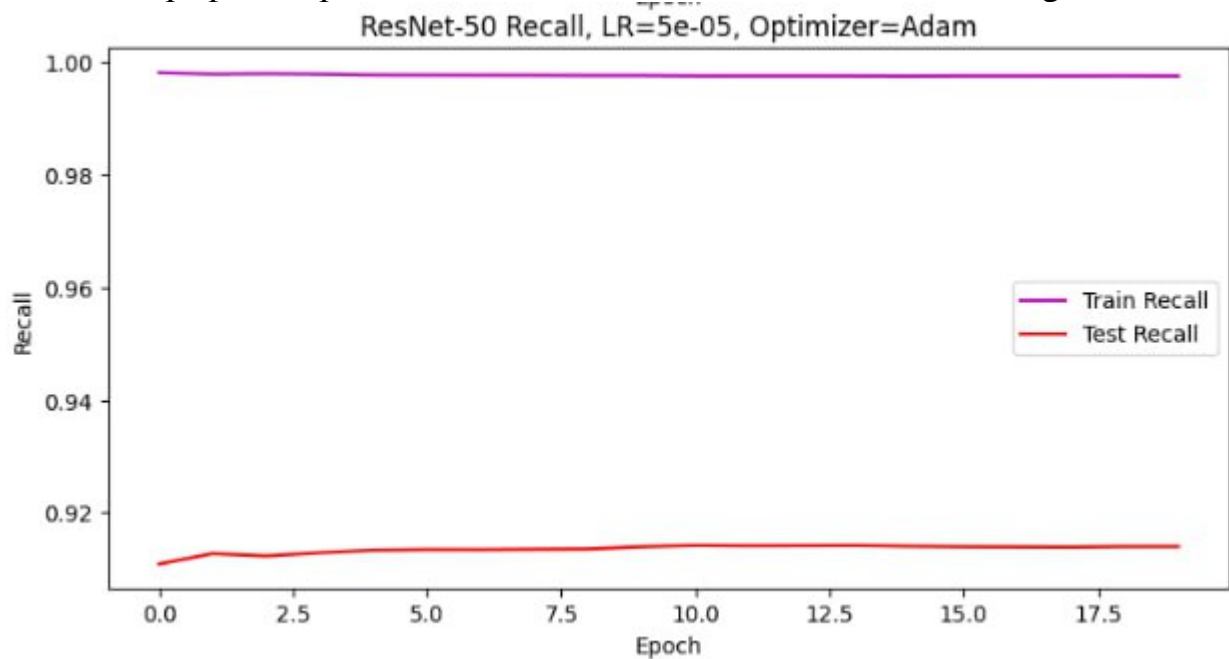


Рис Б.8 Графік метрики Recall MobileNetV1 із значенням learning rate 0.00005

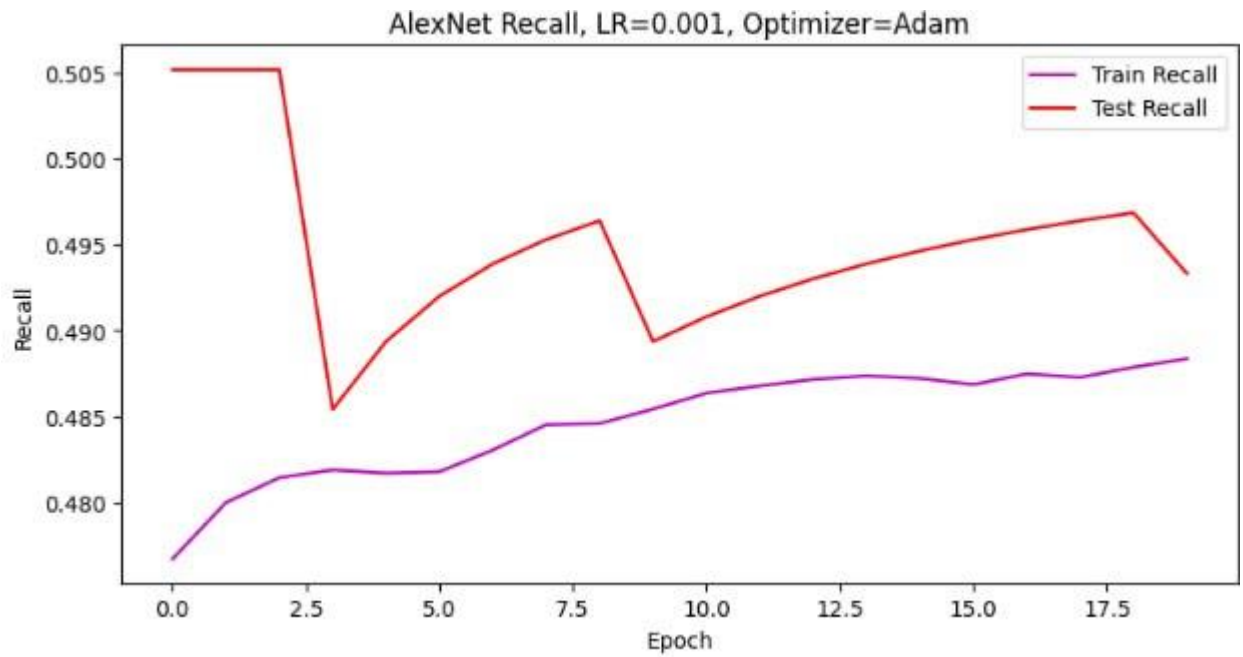


Рис Б.9 Графік метрики Recall AlexNet із значенням learning rate 0.001

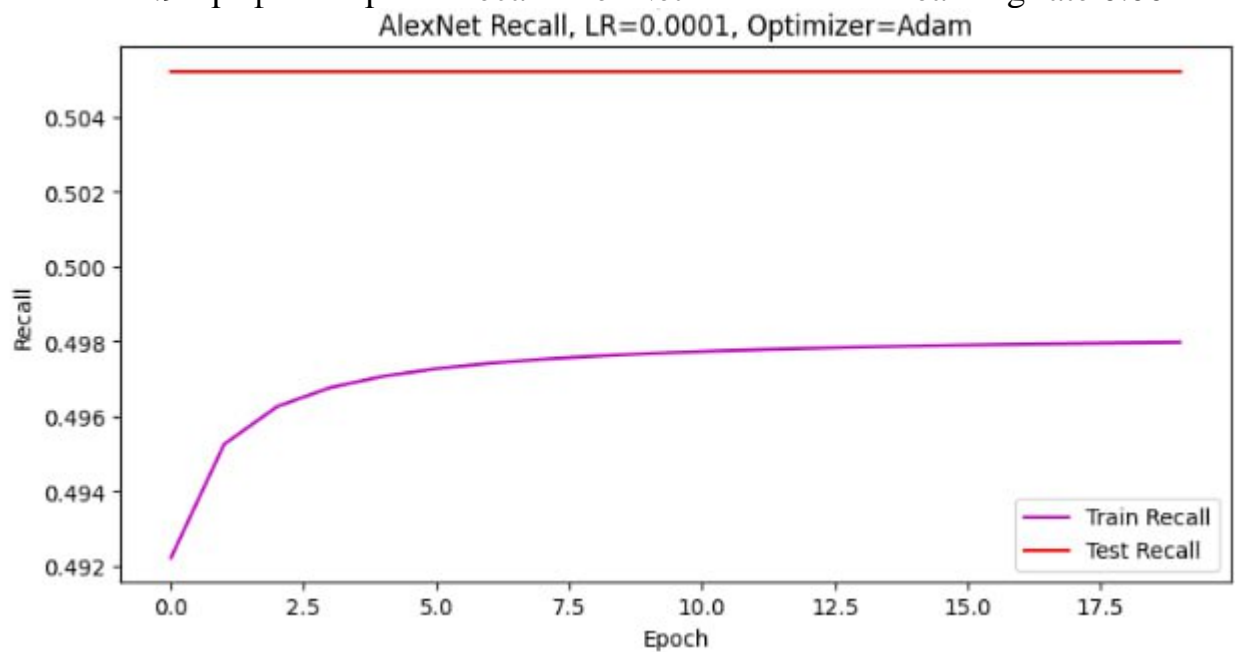


Рис Б.10 Графік метрики Recall AlexNet із значенням learning rate 0.0001

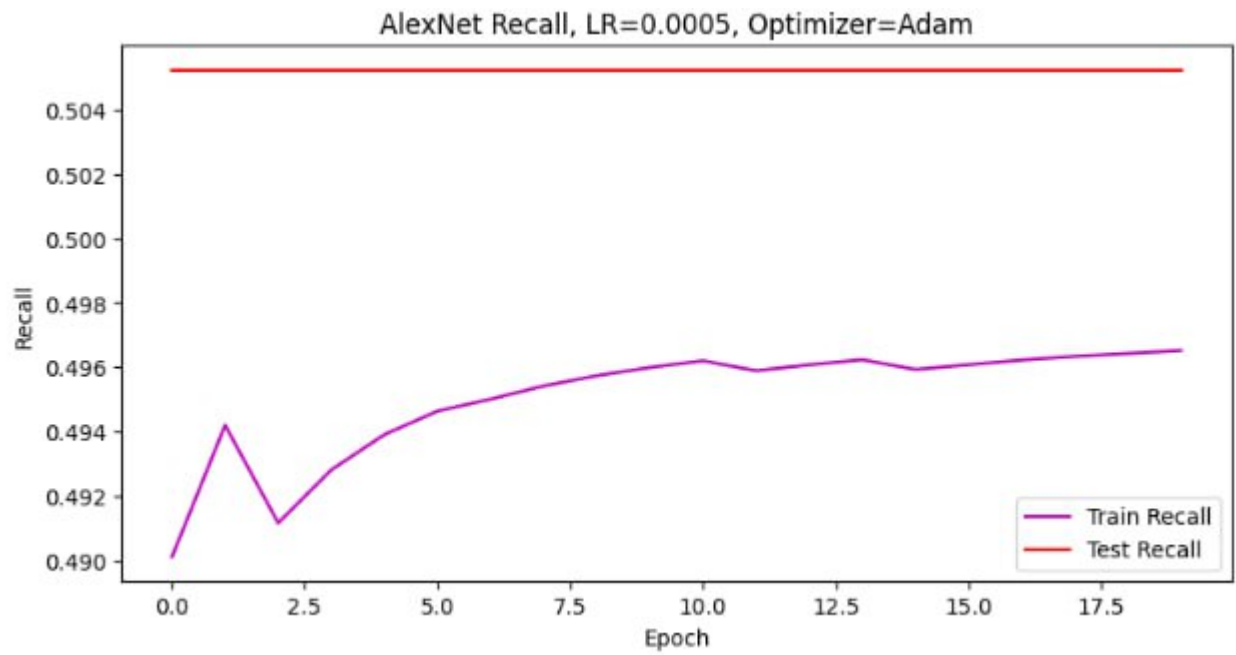


Рис Б.11 Графік метрики Recall AlexNet із значенням learning rate 0.0005

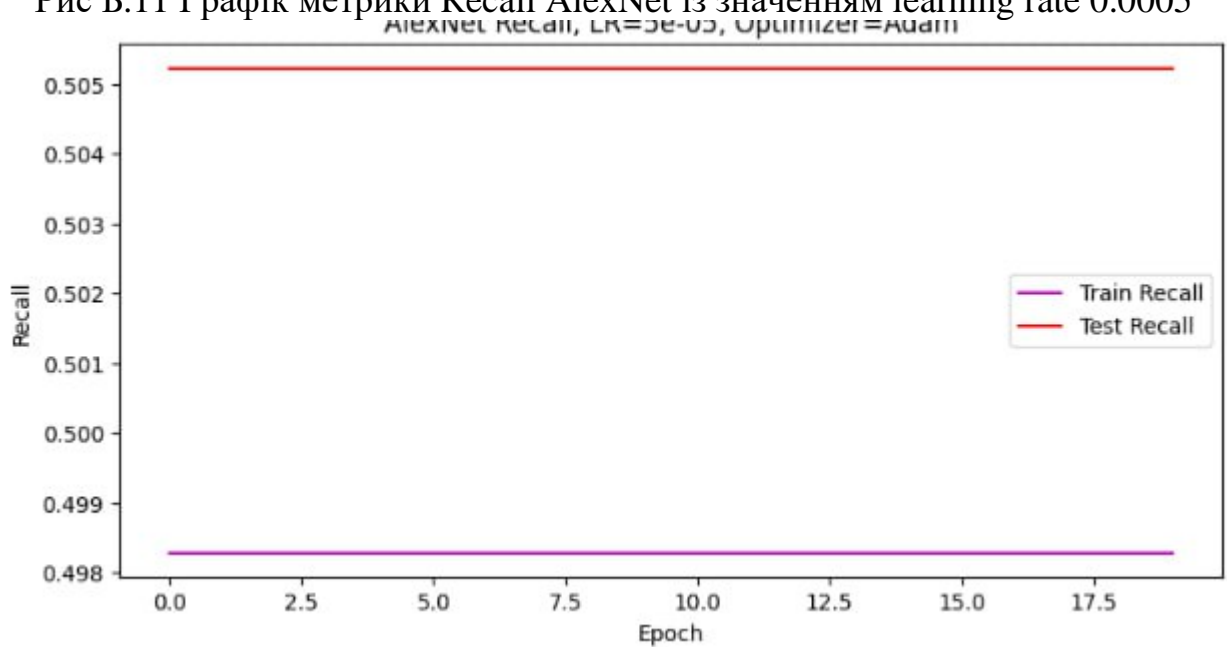


Рис Б.12 Графік метрики Recall AlexNet із значенням learning rate 0.00005

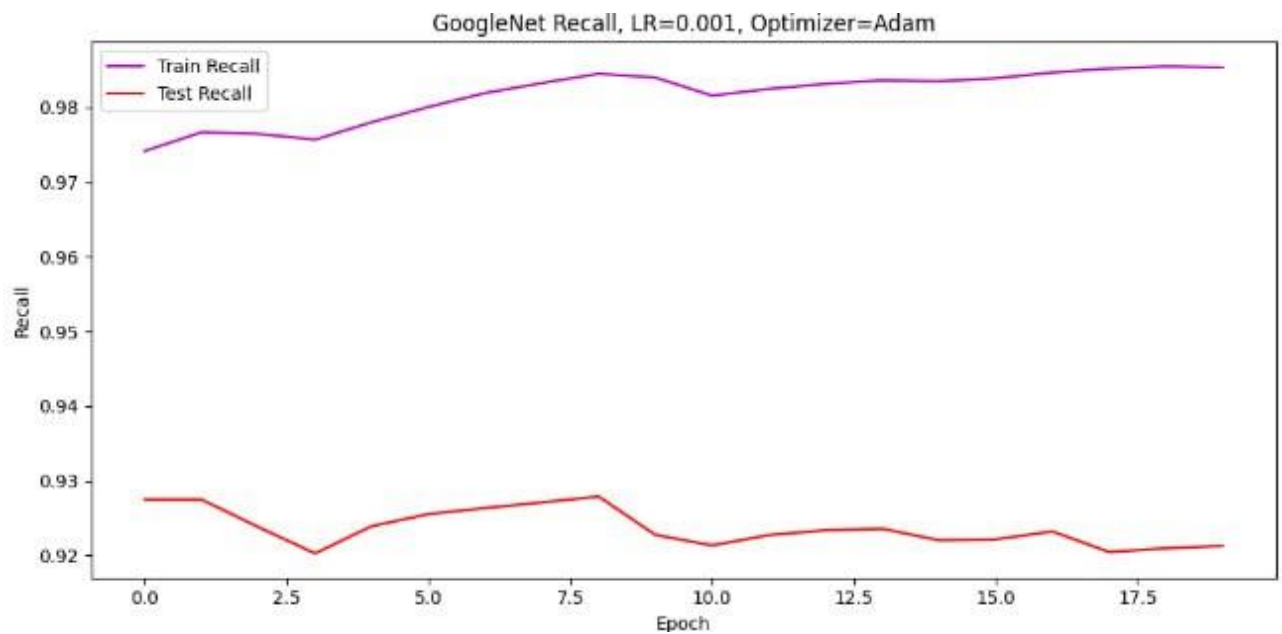


Рис Б.13 Графік метрики Recall GoogleNet із значенням learning rate 0.001

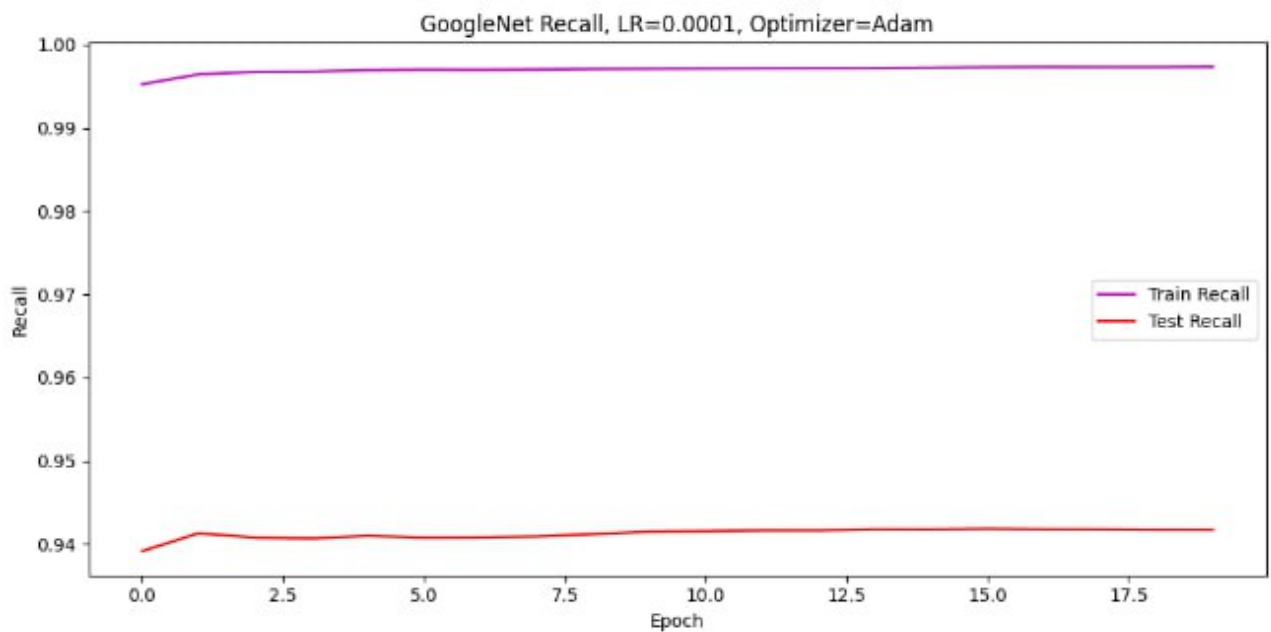


Рис Б.14 Графік метрики Recall GoogleNet із значенням learning rate 0.0001

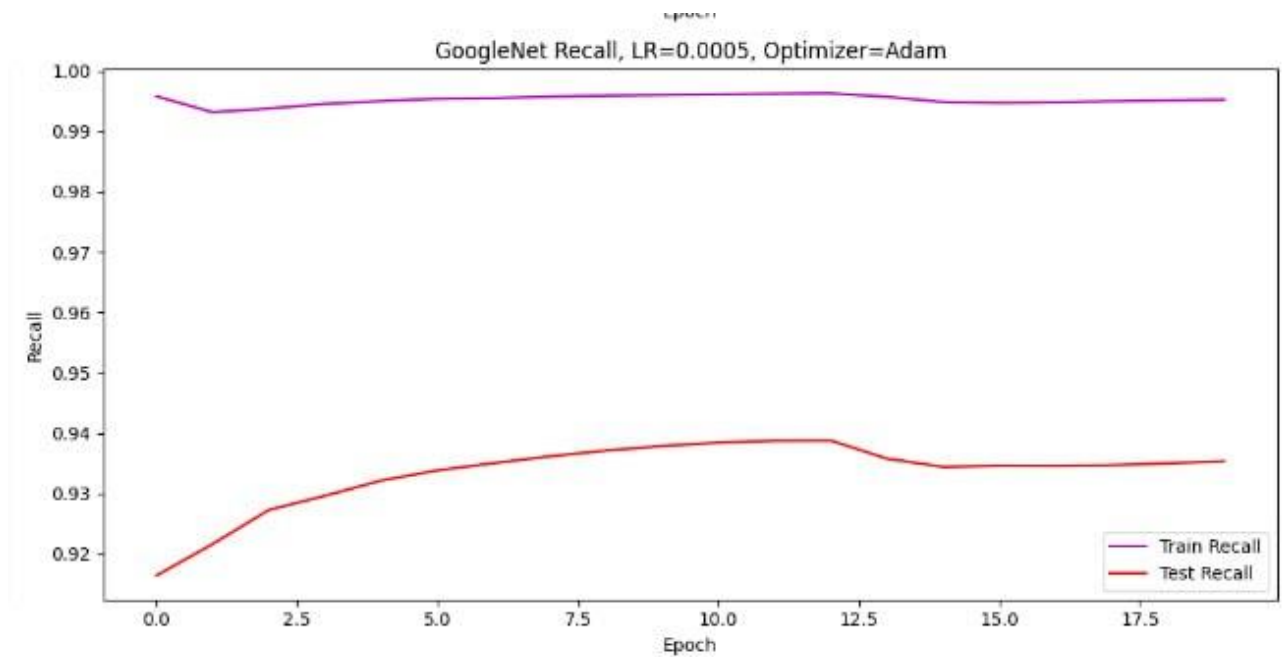


Рис Б.15 Графік метрики Recall GoogleNet із значенням learning rate 0.0005

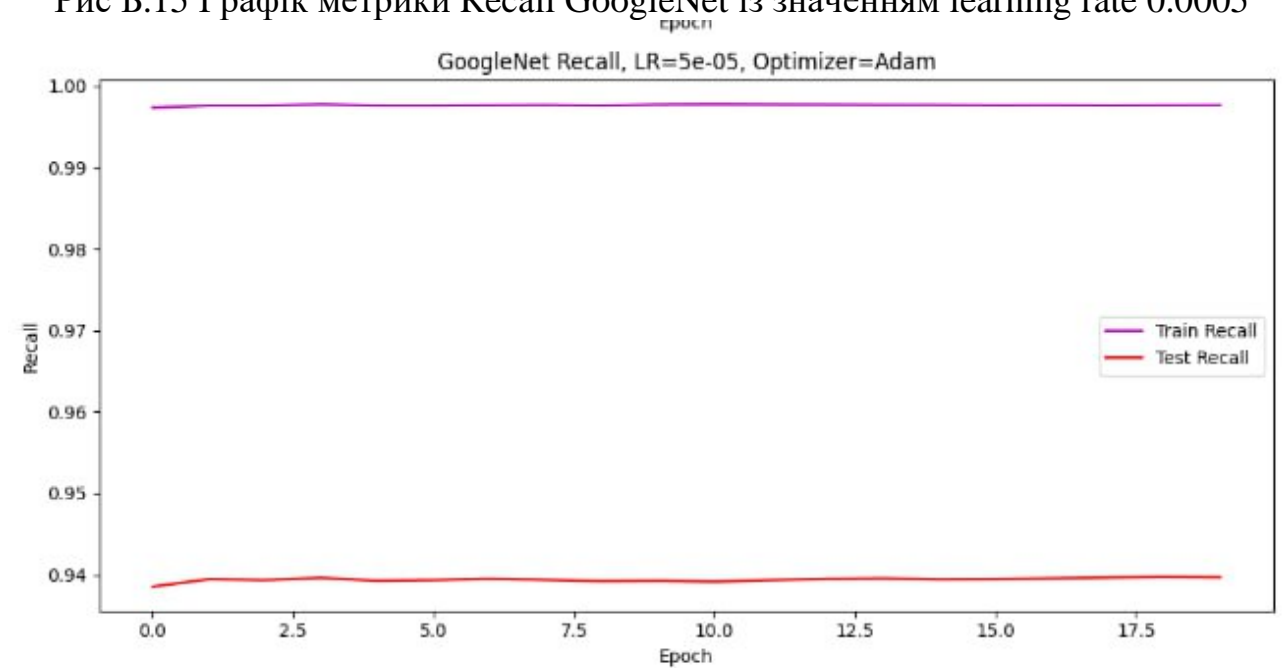


Рис Б.16 Графік метрики Recall GoogleNet із значенням learning rate 0.00005

ДОДАТОК В

Графіки метрики Precision усіх моделей ResNet-50, MobileNetV1, AlexNet, GoogleNet із різними значеннями параметра learning rate та оптимізатором Adam:

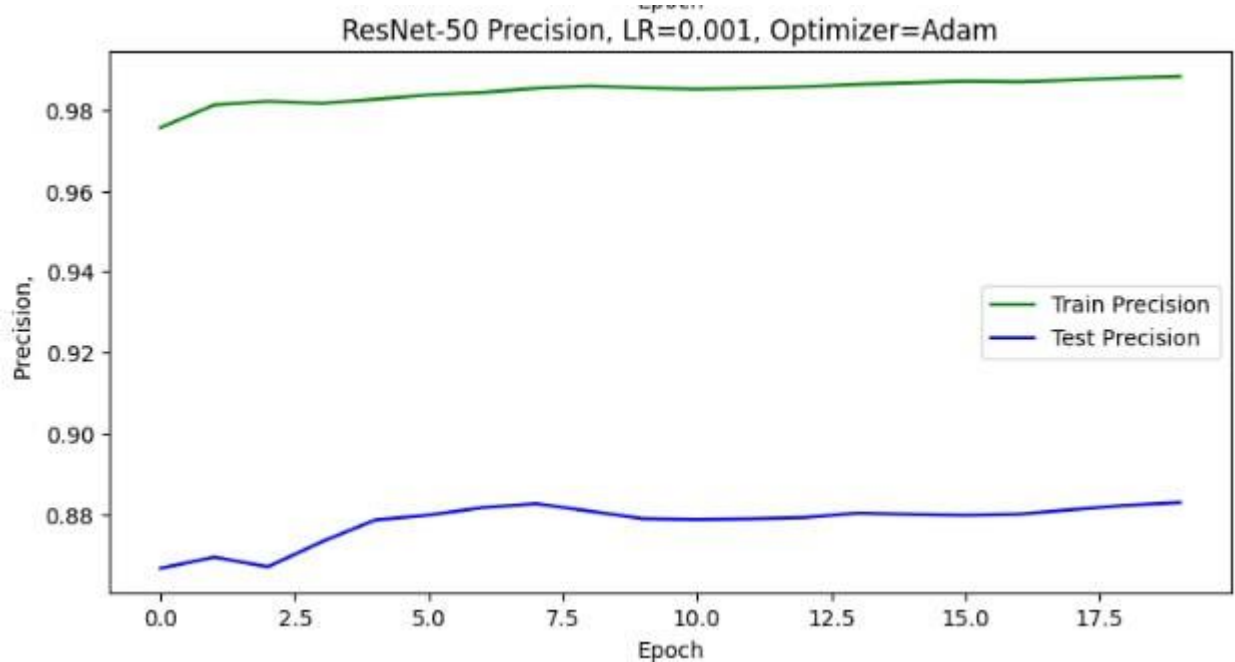


Рис В.1 Графік метрики Recall ResNet-50 із значенням learning rate 0.001

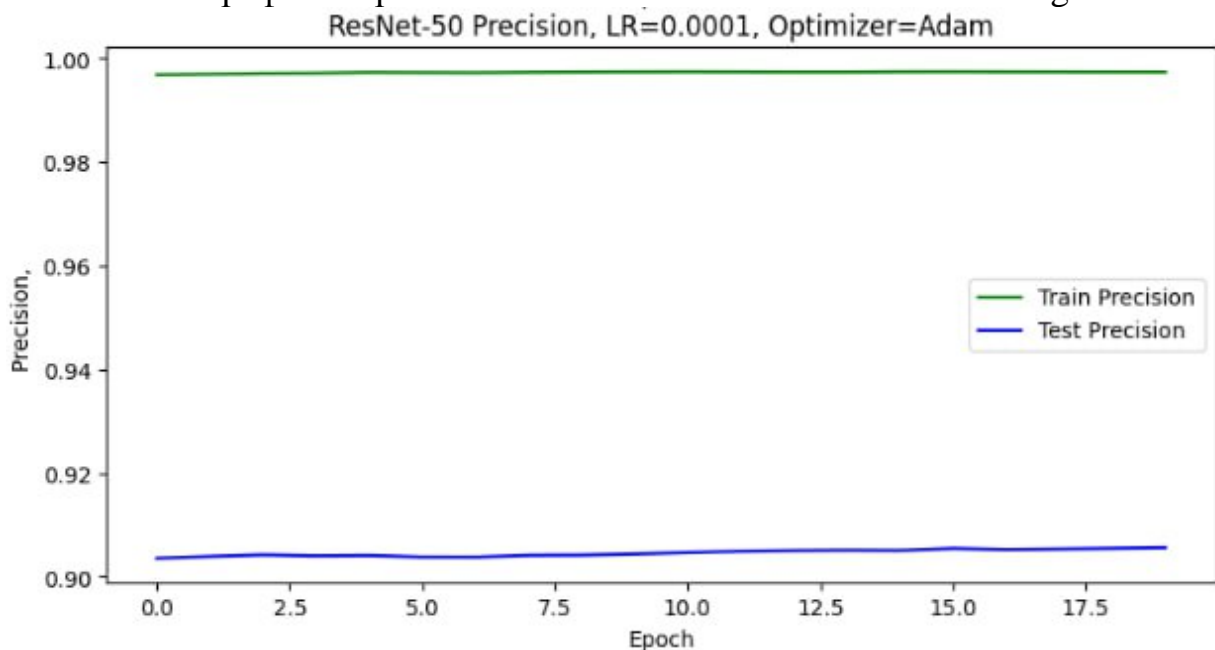


Рис В.2 Графік метрики Recall ResNet-50 із значенням learning rate 0.0001

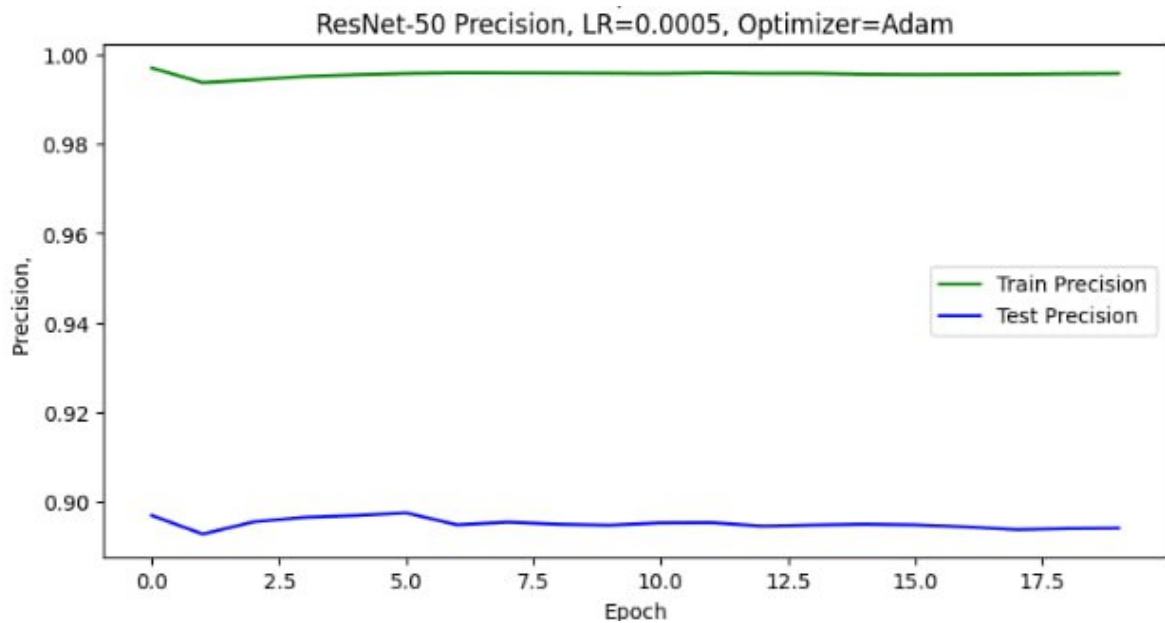


Рис В.3 Графік метрики Recall ResNet-50 із значенням learning rate 0.0005

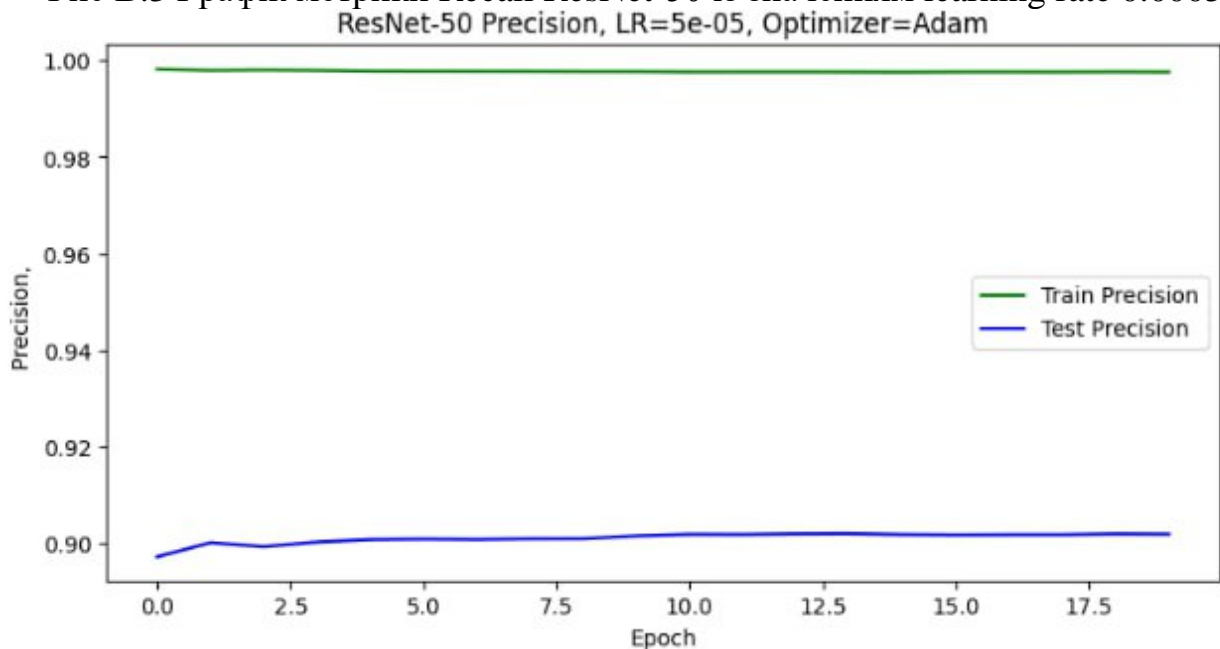


Рис В.4 Графік метрики Recall ResNet-50 із значенням learning rate 0.00005

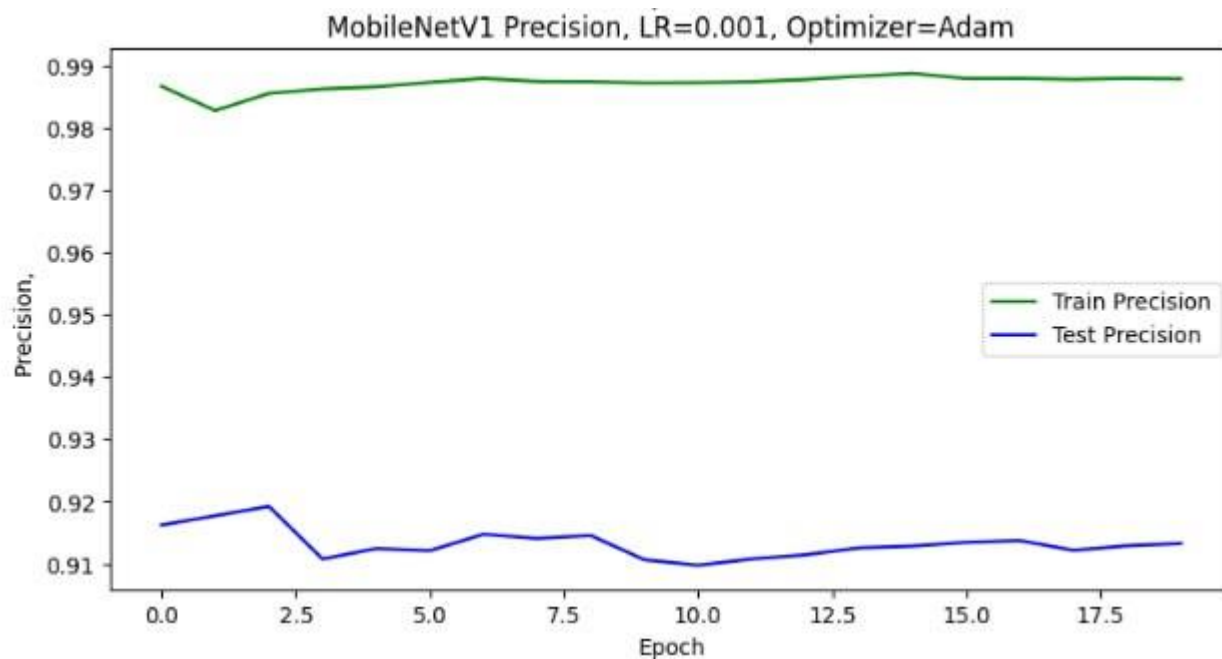


Рис В.5 Графік метрики Recall MobileNetV1 із значенням learning rate 0.001

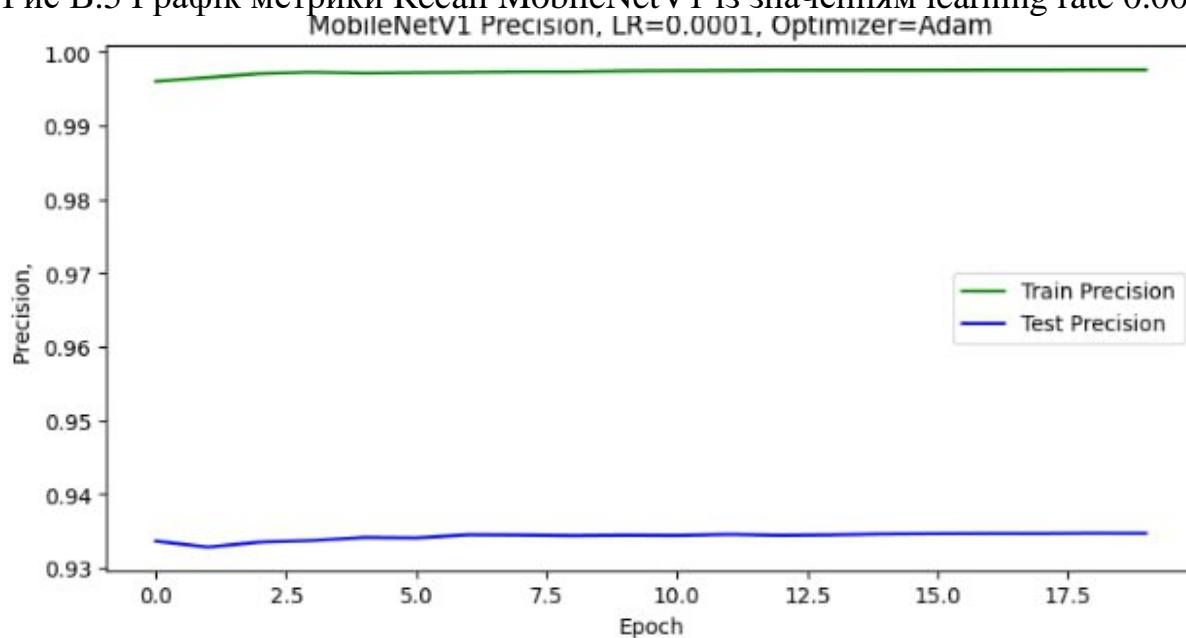


Рис В.6 Графік метрики Recall MobileNetV1 із значенням learning rate 0.0001

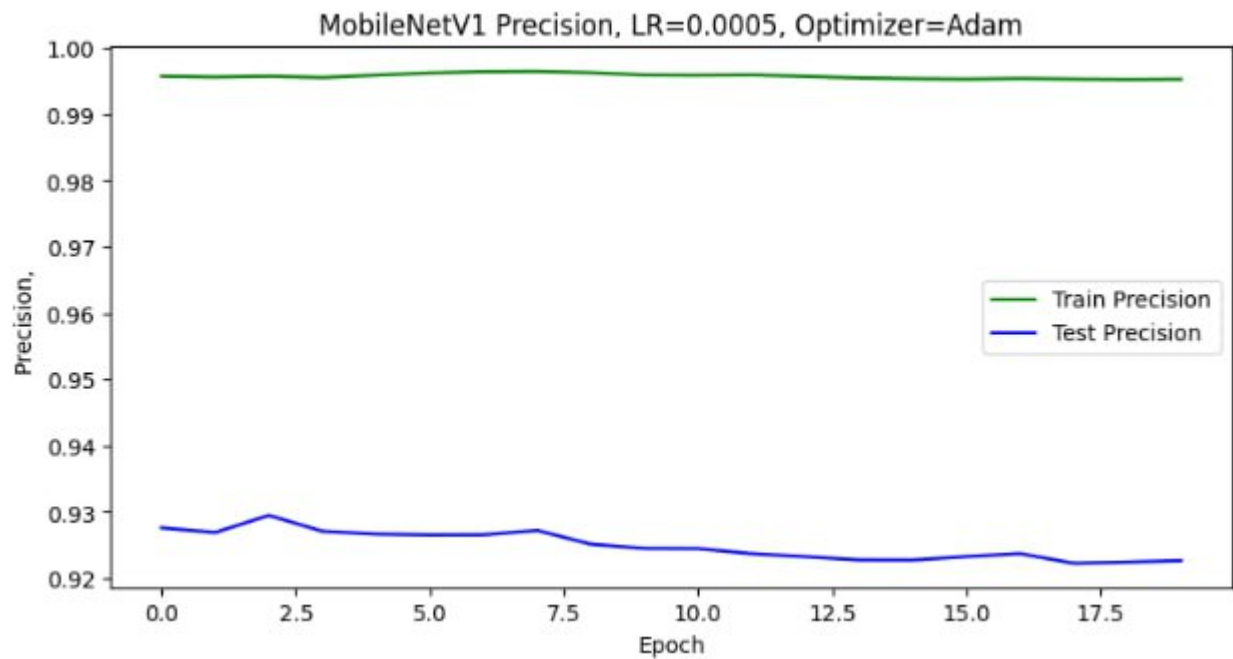


Рис В.7 Графік метрики Recall MobileNetV1 із значенням learning rate 0.0005

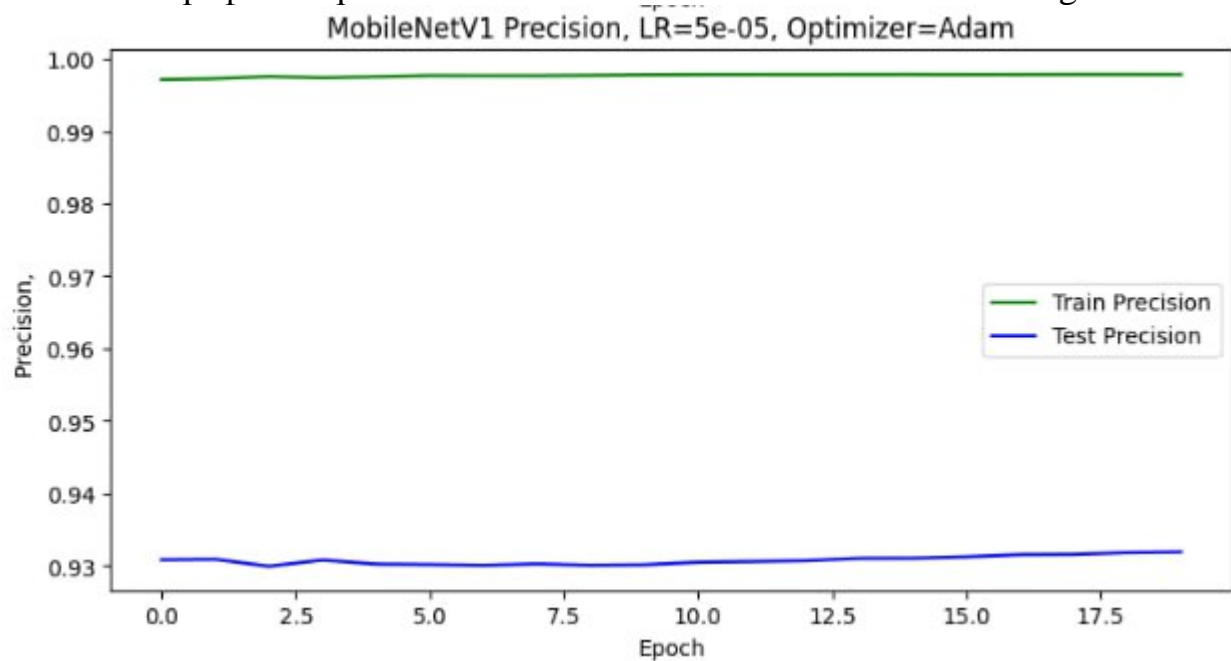


Рис В.8 Графік метрики Recall MobileNetV1 із значенням learning rate 0.00005

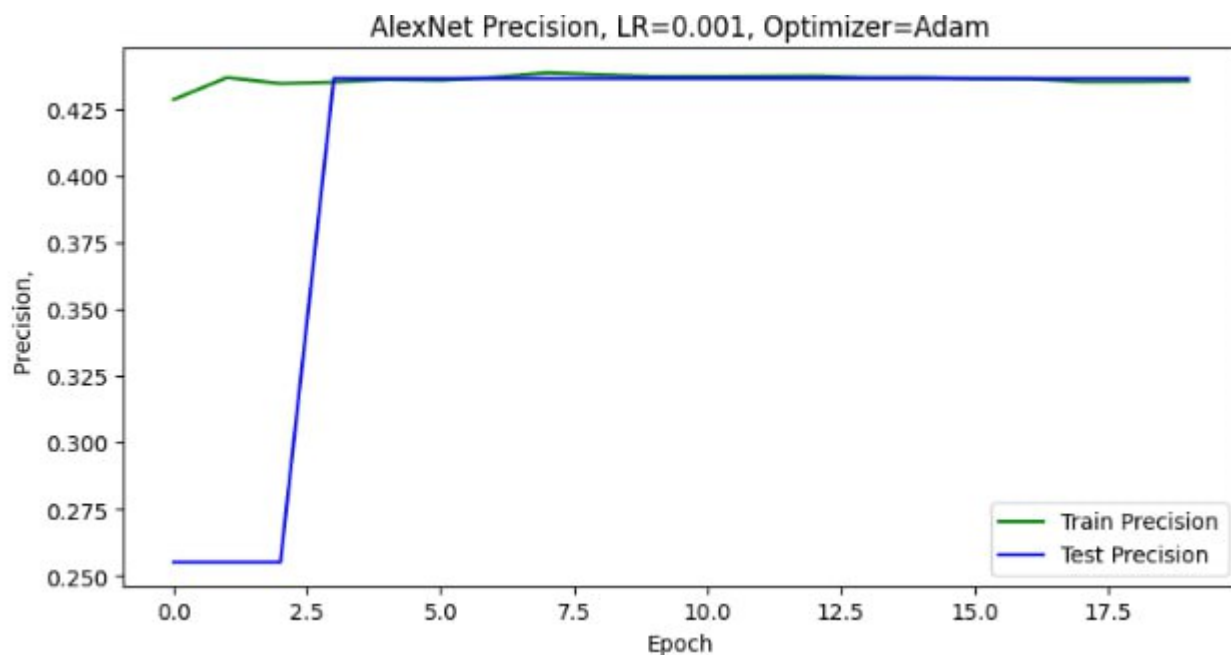


Рис В.9 Графік метрики Recall AlexNet із значенням learning rate 0.001

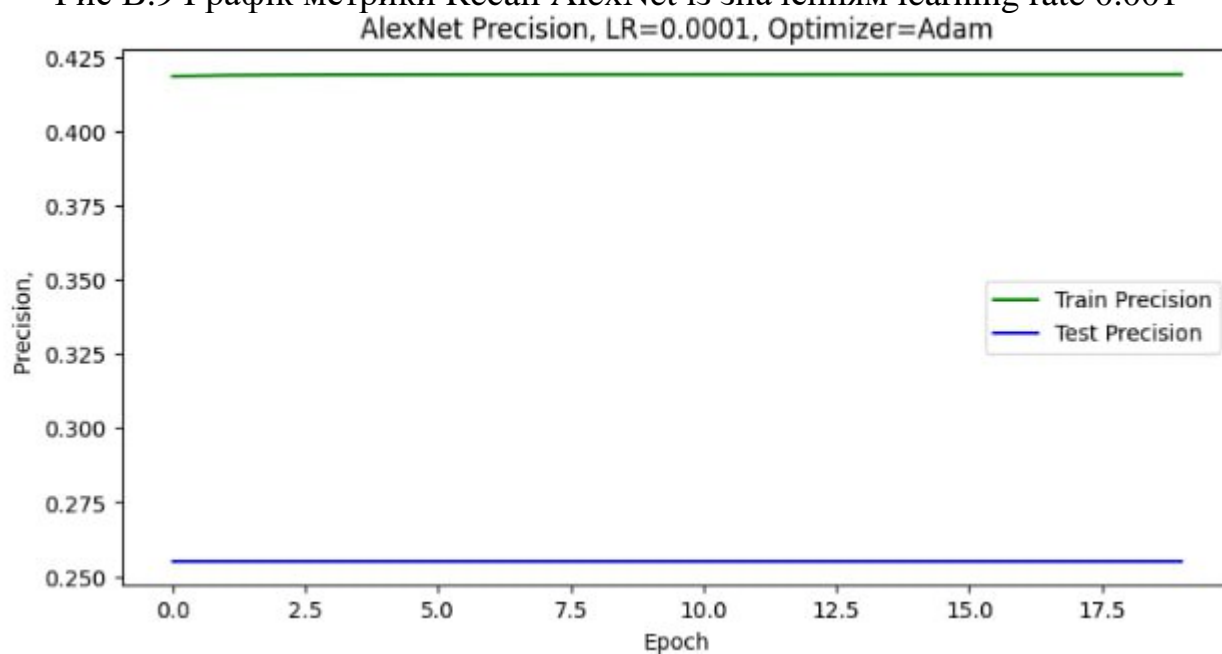


Рис В.10 Графік метрики Recall AlexNet із значенням learning rate 0.0001

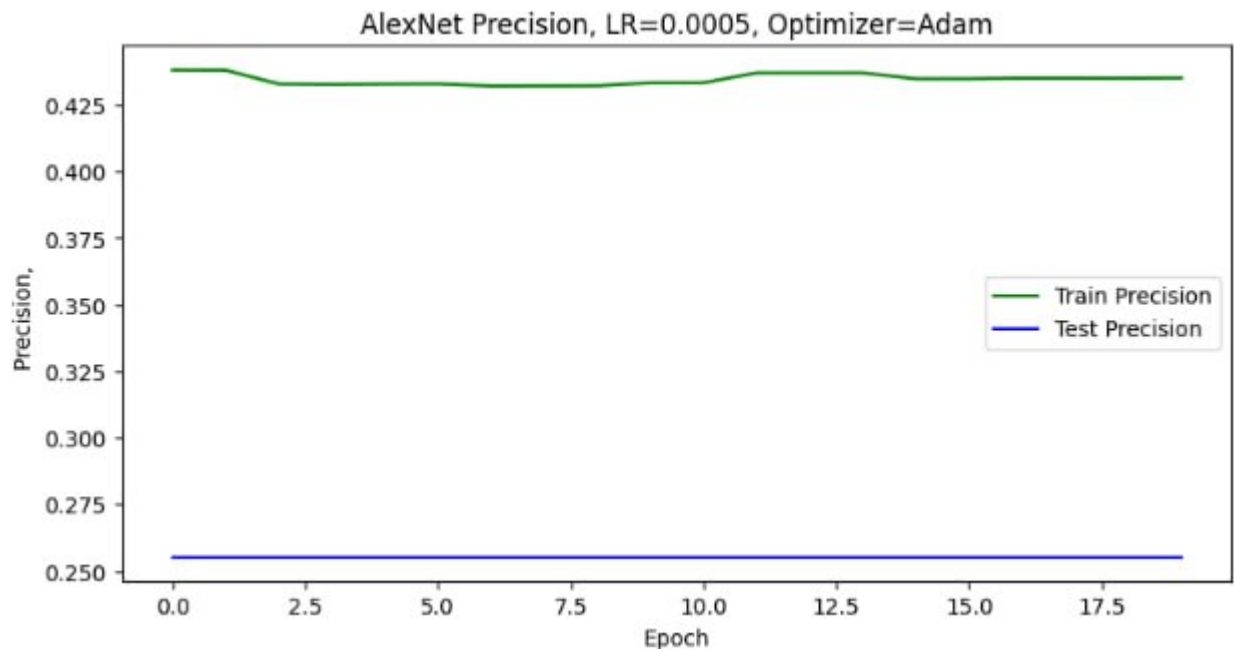


Рис В.11 Графік метрики Recall AlexNet із значенням learning rate 0.0005

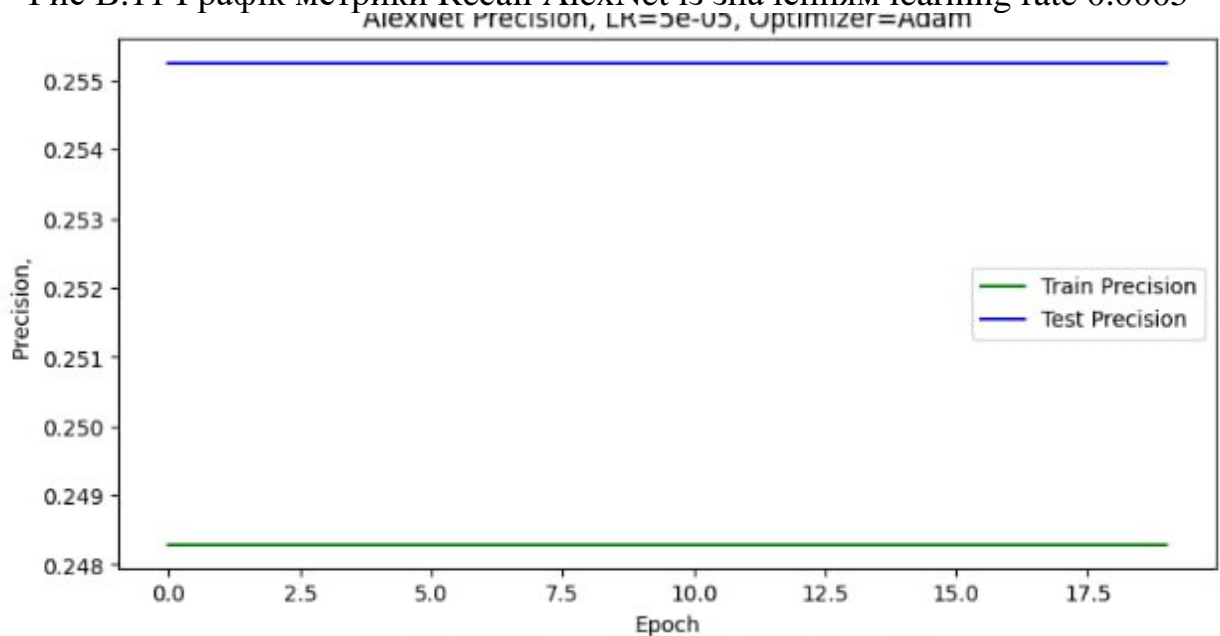


Рис В.12 Графік метрики Recall AlexNet із значенням learning rate 0.00005

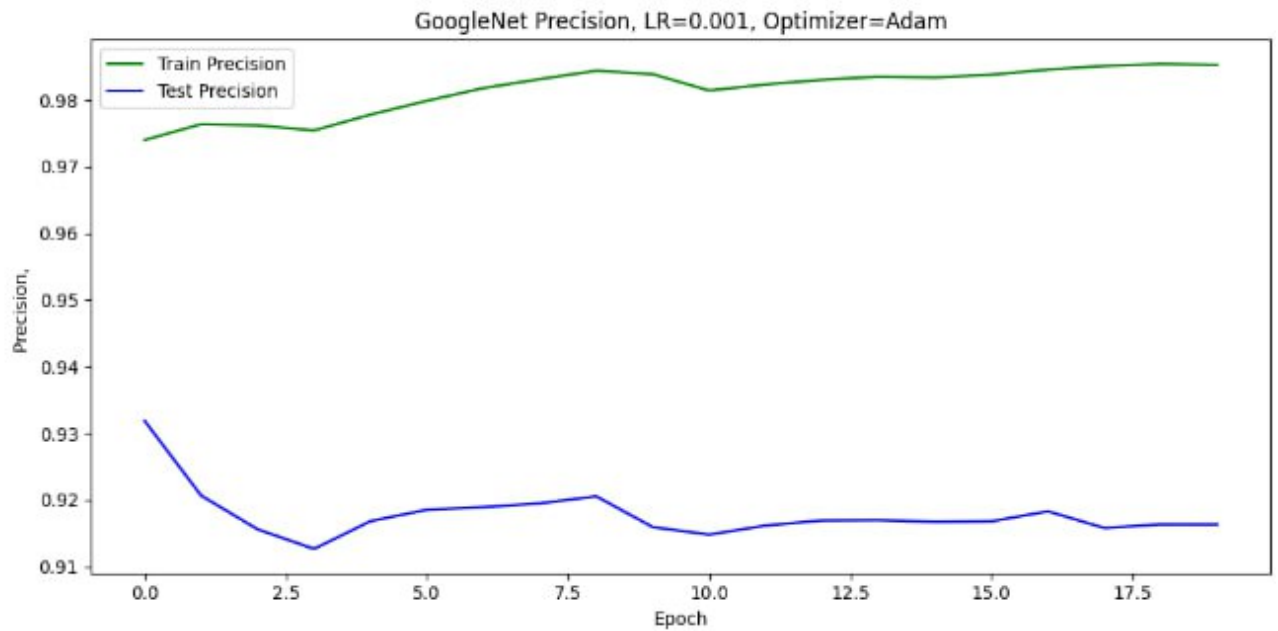


Рис В.13 Графік метрики Recall GoogleNet із значенням learning rate 0.001

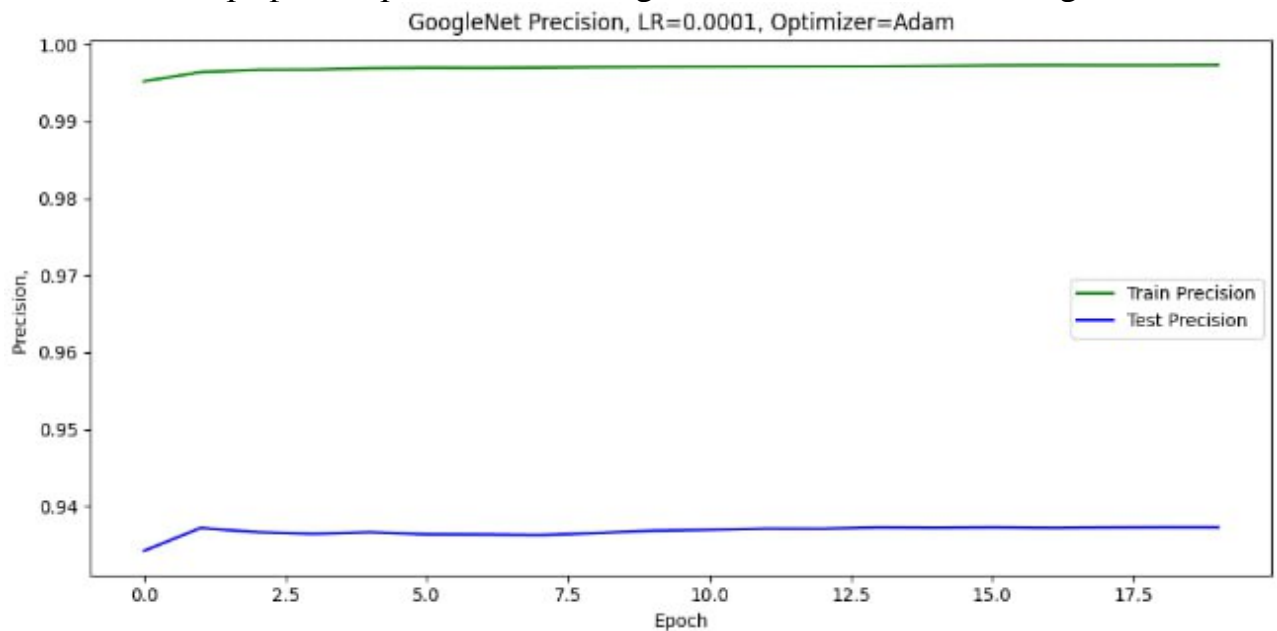


Рис В.14 Графік метрики Recall GoogleNet із значенням learning rate 0.0001

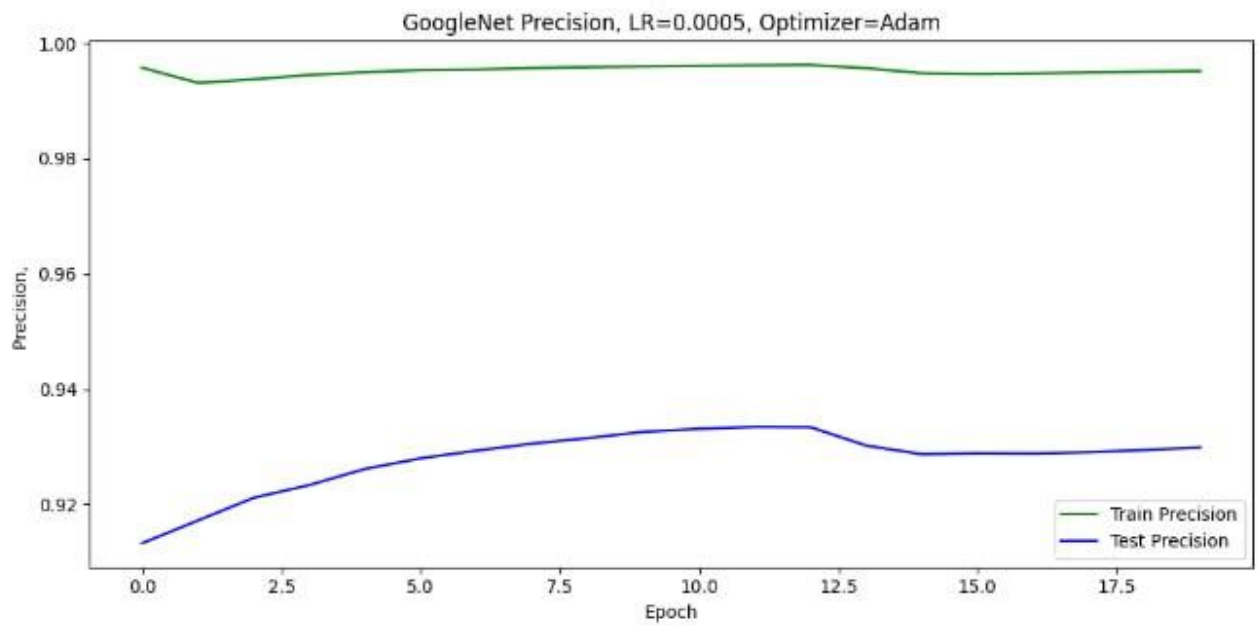


Рис В.15 Графік метрики Recall GoogleNet із значенням learning rate 0.0005

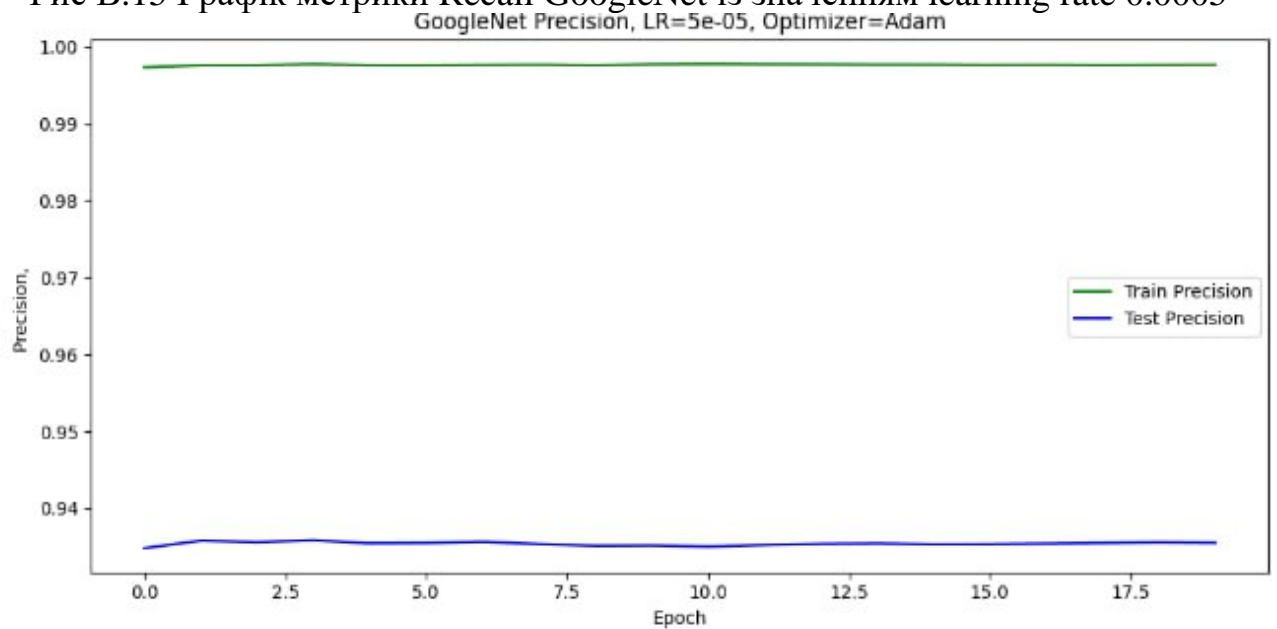


Рис В.16 Графік метрики Recall GoogleNet із значенням learning rate 0.00005

