

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМ. ІГОРЯ СІКОРСЬКОГО”
Факультет електроніки
Кафедра електронної інженерії

"На правах рукопису"

УДК 004.89

«До захисту допущено»
Завідувач кафедри

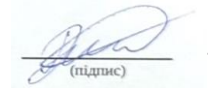
_____ В.І. Тимофєєв
“ ___ ” _____ 20__ р.

Магістерська дисертація

зі спеціальності 153 мікро- та наносистемна техніка

на тему Розпізнавання діабетичної ретинопатії за аналізом зображень сітківки ока
методами машинного навчання

Виконав: студент 2 курсу, групи ДМ-21мп
Дмитришин Владислав Вікторович
(прізвище, ім'я, по батькові)



(підпис)

Науковий керівник доцент каф. ЕІ, к. т. н., доцент, Іванько Катерина Олегівна
(посада, вчене звання, науковий ступінь, прізвище та ініціали)



Рецензент доцент каф. МЕ, к.т.н., доцент, Коваль Вікторія Михайлівна
(посада, вчене звання, науковий ступінь, прізвище та ініціали)



Засвідчую, що у цій магістерській дисертації немає
запозичень з праць інших авторів без відповідних
посилань.

Студент



(підпис)

Київ - 2024 року

Національний технічний університет України
“Київський політехнічний інститут ім. Ігоря Сікорського”

Факультет електроніки

Кафедра електронної інженерії

Рівень вищої освіти – другий (магістерський) за освітньо-професійною програмою

Спеціальність 153 мікро- та наносистемна техніка

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ В.І. Тимофєєв
“ ___ ” _____ 20__ р.

З А В Д А Н Н Я
НА МАГІСТЕРСЬКУ ДИСЕРТАЦІЮ СТУДЕНТУ

Дмитришин Владислав Вікторович

(прізвище, ім'я, по батькові)

1. Тема дисертації Розпізнавання діабетичної ретинопатії за аналізом зображень сітківки ока методами машинного навчання

Науковий керівник Іванько Катерина Олегівна, к. т. н., доцент

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від “09” листопада 2023 року № 5218-с.

2. Строк подання студентом дисертації: 05 січня 2024 року

3. Об'єкт дослідження: зображення очного дна людини.

4. Вихідні дані: Цифрові зображення очного дна пацієнтів з діабетичною ретинопатією різних стадій та здорових людей

5. Перелік питань, які потрібно розробити:

1. Діабетична ретинопатія. Класифікація та діагностика.

2. Основні методи дослідження сітківки ока. Отримання зображень.

3. Попередня обробка зображень.

4. Машинне навчання.

5. Задача розпізнавання діабетичної ретинопатії за аналізом зображень сітківки ока.

6. Перелік графічного (ілюстративного) матеріалу: зображення стадій діабетичної ретинопатії, зображення приладів реєстрації очного дна та принципу їх роботи, архітектура згорткових нейронних мереж(ResNet, VGG, EfficientNet), гістограми розподілу даних у базі даних та у тренувальній і тестовій вибірках, приклади

випадкових зображень з тренувальної вибірки із застосуванням різних методів обробки зображень, графіки точності та втрат для тестової та навчальної вибірок після застосування різних методів обробки зображень

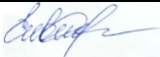
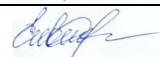
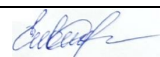
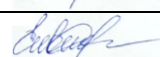
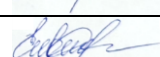
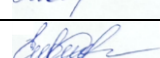
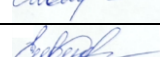
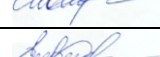
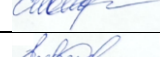
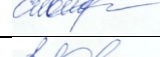
7. Орієнтовний перелік публікацій:

8. Консультанти розділів дисертації

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв

9. Дата видачі завдання: 27 жовтня 2023 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів виконання магістерської дисертації	Строк виконання етапів роботи	Примітка
	Складання плану дипломної роботи	30.10.2023	
	Огляд літературних джерел на тему магістерської дисертації	5.11.2023	
	Виконання розділу 1	12.11.2023	
	Виконання розділу 2	19.11.2023	
	Виконання розділу 3	26.11.2023	
	Виконання розділу 4	3.12.2023	
	Виконання розділу 5	17.12.2023	
	Оформлення пояснювальної записки	31.12.2023	
	Подання роботи на нормоконтроль	5.01.2024	
	Захист магістерської дисертації	16.01.2024	

Студент


(підпис)

Дмитришин В.В.

(прізвище та ініціали)

Науковий керівник роботи


(підпис)

Іванько К.О

(прізвище та ініціали)

РЕФЕРАТ

Магістерська дисертація: 104 сторінок, 5 розділів, 44 рисунки, 71 джерело, 2 додатки (6 сторінок).

ДІАБЕТИЧНА РЕТИНОПАТІЯ, ЦУКРОВИЙ ДІАБЕТ, МАШИННЕ НАВЧАННЯ, ШТУЧНИЙ ІНТЕЛЕКТ, ЗОБРАЖЕННЯ СІТКІВКИ ОКА, ФУНДУС-КАМЕРА, ЗГОРТКОВІ НЕЙРОННІ МЕРЕЖІ, МЕТОДИ ОБРОБКИ ЗОБРАЖЕНЬ.

У роботі розглянуто задачу класифікації зображень методами машинного навчання та перевірка впливу на точність класифікації основних методів обробки біомедичних зображень. Для навчання використовується промаркована лікарями база даних із зображеннями сітківки ока, яка знаходиться у відкритому доступі.

Метою даної роботи є розробка моделі згорткової нейронної мережі для діагностики діабетичної ретинопатії.

Предметом роботи зображення очного дна людини в нормі та з наявністю ознак діабетичної ретинопатії на різних стадіях хвороби.

Перший розділ пояснює природу виникнення та протікання хвороби, акцентується увага на частоті виникнення хвороби серед населення, методах діагностики та факторах, що призводять до прогресу хвороби.

У другому розділі розглядається різна апаратура для діагностики сітківки, а також аналізується прогрес в сфері оптики та електроніки для діагностики хвороб сітківки.

Третій розділ присвячено методам обробки зображень та їх математичному опису.

У четвертому розділі розглянуто засади побудови нейронних мереж та розглянуто їх основні структури

П'ятий розділ присвячений практичній реалізації методу для вирішення поставленої задачі, порівнянню результатів з використанням різних методів попередньої попередньої обробки зображень очного дна.

ABSTRACT

Master's thesis: 104 pages, 5 chapters, 44 figures, 71 sources, 1 appendix (6 pages).

DIABETIC RETINOPATHY, DIABETES, MACHINE LEARNING, ARTIFICIAL INTELLIGENCE, RETINA IMAGING, FUNDUS CAMERA, CONVOLUTION NEURAL NETWORKS, IMAGE PROCESSING METHODS.

The paper examines the problem of image classification using machine learning methods and checks the impact on classification accuracy of basic methods of processing biomedical images. For training, a database marked by doctors with images of the retina, which is publicly available, is used.

The purpose of this work is to develop a convolutional neural network model for diagnosing diabetic retinopathy.

The subject of the work is the image of the fundus of a person in normal conditions and with the presence of signs of diabetic retinopathy at various stages of the disease.

The first chapter explains the nature of the occurrence and course of the disease, focusing on the frequency of occurrence of the disease among the population, methods of diagnosis and factors that lead to the progress of the disease.

The second chapter examines various retinal diagnostic equipment and analyzes progress in the field of optics and electronics for the diagnosis of retinal diseases.

The third chapter is devoted to image processing methods and their mathematical description.

The fourth chapter examines the principles of building neural networks and examines their main structures

The fifth chapter is devoted to the practical implementation of the method for solving the given problem, the comparison of results using different methods of preliminary preprocessing of images of the fundus.

ЗМІСТ

ВСТУП.....	10
1. ДІАБЕТИЧНА РЕТИНОПАТІЯ. КЛАСИФІКАЦІЯ ТА ДІАГНОСТИКА	13
1.1 Стадії діабетичної ретинопатії	15
1.2 Висновки до розділу 1	20
2. ОСНОВНІ МЕТОДИ ДОСЛІДЖЕННЯ СІТКІВКИ ОКА. ОТРИМАННЯ ЗОБРАЖЕНЬ	21
2.1 Оптична когерентна томографія	21
2.2 Флюоресцентна ангіографія очного дна	24
2.3 Офтальмоскопія	26
2.4 Фотореєстрація очного дна за допомогою фундус-камери.....	32
2.5 Висновки до розділу 2.....	36
3. ПОПЕРЕДНЯ ОБРОБКА ЗОБРАЖЕНЬ	37
3.1 Методи просторової фільтрації.....	37
3.2 Еквалізація гістограми яскравості	39
3.3 Гамма-корекція	41
3.4 Методи бінаризації	43
3.5 Методи виділення границь зображення	46
3.5.1 Оператор Кенні.....	48
3.5.2 Оператор Собеля	50
3.5.3 Оператор Лапласа.....	51
3.5.4 Оператор Робертса	52
3.5.5 Оператор Превітта.....	53
3.6 Висновки до розділу 3	54
4. МАШИННЕ НАВЧАННЯ.....	56
4.1 Машинне навчання з вчителем	56
4.2 Машинне навчання без вчителем.....	57
4.3 Навчання з підкріпленням	59

4.4	Нейронні мережі	60
4.4.1	Згорткові нейронні мережі	61
4.5	Висновки до розділу 4	65
5.	ЗАДАЧА РОЗПІЗНАВАННЯ ДІАБЕТИЧНОЇ РЕТИНОПАТІЇ ЗА АНАЛІЗОМ ЗОБРАЖЕНЬ СІТКІВКИ ОКА	67
5.1	Огляд бази даних.....	67
5.2	Розробка власної моделі нейронної мережі.....	71
5.3	Розробка моделі з використанням попередньо натренованої моделі	74
5.4	Висновки до розділу	87
	ВИСНОВКИ.....	89
	ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ.....	91
	Додаток А Скрипт до задачі розпізнавання діабетичної ретинопатії за аналізом зображень сітківки ока.....	99
	Додаток Б Підсумкова таблиця точності отриманої моделі з використанням методів обробки зображень.....	104

ВСТУП

Магістерська дисертація присвячена розробці моделі згорткової нейронної мережі для вирішення задачі класифікації зображень сітківки ока з метою діагностики діабетичної ретинопатії та визначення стадій захворювання. В роботі застосовано такі методи обробки біомедичних зображень, як еквалізація гістограми яскравості, методі виявлення країв, а також побудовано згорткові нейронні мережі на основі попередньо натренованих моделей.

Око є одним із основних органів відчуття, проте на сьогоднішній день все більше людей страждають на вади зору. Причиною часто є невчасне виявлення хвороби і відповідно пізні лікування, чого можна було б запобігти за умови виявлення хвороби на початкових стадіях. Проте на сьогоднішній день не велика кількість українських міст володіють належною апаратурою та кваліфікованими кадрами для якісної діагностики. Розвиток штучного інтелекту та електроніки дозволить покращити умови та якість діагностики захворювань очей, зокрема й діабетичної ретинопатії.

Практична значущість роботи полягає в розробці алгоритму з високим рівнем точності з можливістю його подальшого використання для виявлення діабетичної ретинопатії для зменшення навантаження на лікарів-офтальмологів та покращення якості діагностики хвороби.

Тема роботи актуальна у зв'язку зі збільшенням темпів розвитку діабетичної ретинопатії та цукрового діабету. В першому розділі роботи наведено статистичні дані, що підтверджують факт, що все більше людей у світі хворіють на цукровий діабет, а отже потрапляють до групи ризику. У розділі розглянуто стадії діабетичної ретинопатії, а також наведено основні методи діагностики діабетичної ретинопатії та фактори, що призводять до прогресу хвороби.

Другий розділ описує методи дослідження сітківки ока та прилади для отримання зображень. У цьому розділі основною метою є показати науковий прогрес у галузі оптики та електроніки, що призвело до переходу приладів реєстрації від об'ємних та важких, до максимально простих. Також розділ демонструє, що якісна діагностика сітківки ока вимагає кваліфікований персонал для роботи із сучасною апаратурою.

В третьому розділі наведено основні методи обробки зображень, що застосовуються в тому числі й до біомедичних зображень. Розглянуто алгоритми та наведено математичні формули. Варто звернути увагу, що у деяких статтях [1] описані методи використовують для сегментації судин, що має практичне застосування. Ці методи будуть також використовуватись у практичному розділі для попередньої обробки зображень перед навчанням моделі нейронної мережі. А у висновку буде порівняно ефективність кожного із запропонованих методів обробки зображень.

У четвертому розділі наведено основну інформацію про методи машинного навчання, його види, сфери застосування, основні задачі, які вирішує окремий вид машинного навчання. А також описано структуру нейронних мереж, зокрема згорткових. Пояснено призначення окремих шарів нейронних мереж та функцій активації. Окреслено методологію оцінки якості нейронної мережі, основні метрики. А також представлена архітектура найпопулярніших попередньо навчених згорткових нейронних мереж.

Останній розділ присвячено практичній реалізації розглянутих методів обробки зображень очного дна, а також застосування згорткових нейронних мереж для виявлення ознак діабетичної ретинопатії. Описано базу даних зображень сітківки та вказано на її позитивні та негативні аспекти, проведено попередній аналіз бази даних.. Показаний процес створення моделі на основі попередньо натренованої моделі, описано всі необхідні для цього функції та параметри. І найголовніше, проведено порівняльну роботу серед методів обробки зображень для виявлення методу, з

використанням якого модель матиме найвищу точність класифікації зображень очного дна для виявлення діабетичної ретинопатії та визначення її стадії. Як результат, вдалось отримати значення точності на рівні близько 83%, порівняно з точністю в 75% без попередньої обробки зображень.

У висновку узагальнено результати роботи, підведено підсумки та надано рекомендації, щодо подальшого впровадження алгоритмів машинного навчання у діагностику діабетичної ретинопатії та подальшого розповсюдження можливостей такої діагностики.

1. ДІАБЕТИЧНА РЕТИНОПАТІЯ. КЛАСИФІКАЦІЯ ТА ДІАГНОСТИКА

За даними Всесвітньої організації охорони здоров'я (ВООЗ) більше 2 мільярдів людей мають вади зору. З них майже для половини людей можна було б попередити недугу або навіть запобігти їй. Все ж основним захворюванням, що спричиняє вади зору серед цього мільярду людей є пресбіопія (826 млн), крім неї - катаракта (94 млн), аномалія рефракції (88 млн), вікова молекулярна дегенерація (8 млн), глаукома (7,7 млн) та діабетична ретинопатія (3,9 млн) [2].

Діабетична ретинопатія - це повне чи часткове ураження сітківки, що спричинене цукровим діабетом. Високий рівень глюкози у крові призводить до того, що судини стають більш крихкими, через що крововилив у сітківку ока відбувається значно частіше. З прогресуванням ретинопатії стан зору людини стає все гіршим. Діабетична ретинопатія може бути причиною зниження гостроти зору, молекулярного набряку, крововиливу, відшарування сітківки та навіть сліпоти.

Станом на 2014 рік ВООЗ повідомляє про 422 мільйони людей, що хворіють на цукровий діабет, враховуючи темпи поширення хвороби можемо сказати, що на сьогоднішній день близько півмільярда людей хворі на діабет [3].

Якщо ж говорити про Україну, то згідно з Міжнародною діабетичною федерацією в країні на 2021 рік проживало близько 2,325 млн людей з діабетом[4]. За даними Мінфіну в Україні на кінець 2021 року проживало 41,208 млн людей[5]. Тобто 5,64% населення хворі на діабет. Тоді можемо припустити, що на діабетичну ретинопатію в Україні хворіє близько 25 тисяч людей.



Рисунок 1.1 – Людина з діабетичною ретинопатією буде бачити розмите та спотворене зображення [6]

В більшій частині випадків до діабетичної ретинопатії схильні люди більш дорослі за віком, особливо хворі на діабет більше 20 років літні люди. До основних симптомів зазвичай відносять почуття дискомфорту в очах, загальне погіршення гостроти зору, поява чорних крапок в полі зору, почуття пелени перед очима. Проте всі ці симптоми свідчать про більш запущені стадії, при визначенні яких буде набагато складніше зберегти зір. У разі не лікування симптомів і відсутності своєчасного втручання можуть відбутись наступні ускладнення: підвищений внутрішньоочний тиск, крововилив у кришталик і далі відшарування сітківки і відповідно - сліпота.

Розглянемо, як виглядає здорова сітківка і на що звертати увагу на зображеннях при діагностуванні хвороби.

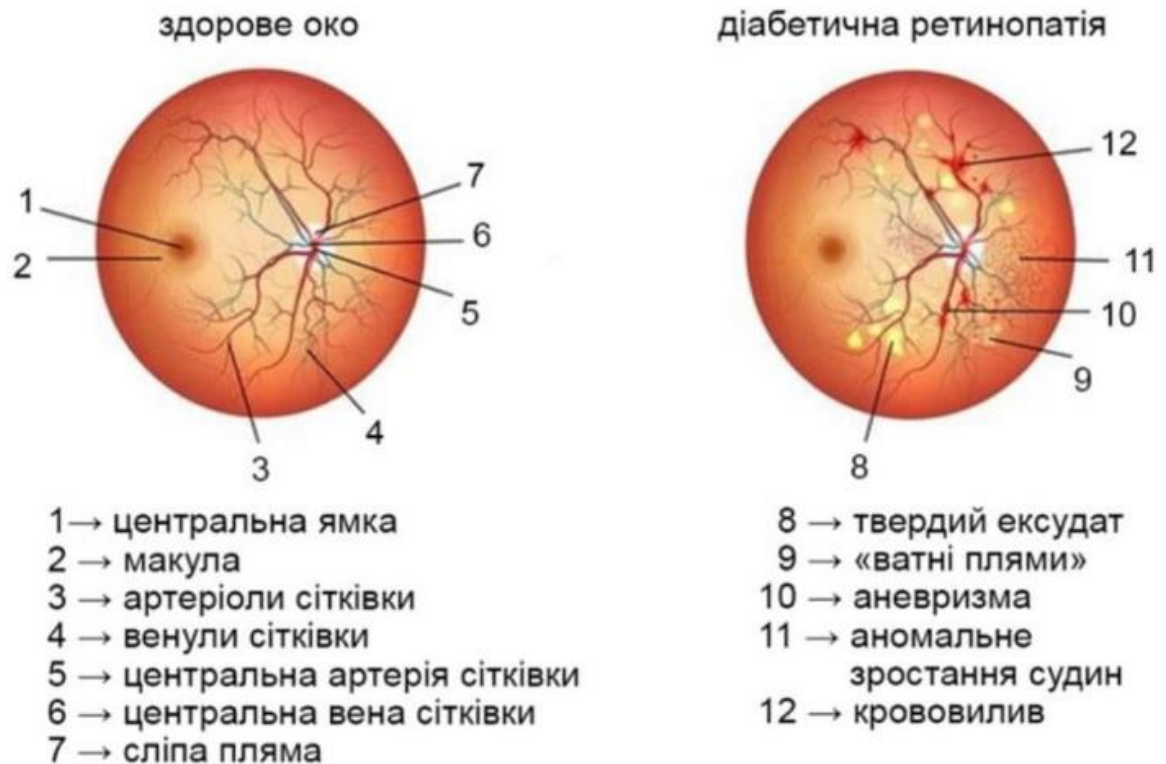


Рисунок 1.2 – Схематичне порівняння здорової та хворої на ретинопатію сітківки ока [7]

1.1 Стадії діабетичної ретинопатії

Дослідники виділяють основні 4 стадії діабетичної ретинопатії [8]:

1. Непроліферативна ретинопатія: ранній етап розвитку патології. Початок процесу ураження капілярів, що живлять сітківку ока. Дегенеративні процеси уражають сітківку і капіляри стають більш проникними, через що утворюється набряк сітківки. Зорові функції при цьому етапі не порушуються, проте людина може скаржитись на погіршення зору. Лікування на цій стадії не потрібне, буде достатньо контролю рівня цукру та спостереження за станом очного дна.

2. Препроліферативна ретинопатія: з'являються перші крововиливи та зони очної ішемії, в оці накопичується рідина. На цій стадії відмічають значну нестачу кисню для очей і цей етап прийнято вважати каталізатором необоротних трансформацій в оці у разі відсутності належного лікування. Через крововиливи починається загибель нервових клітин, що призводить до погіршення зору.
3. Проліферативна ретинопатія: відбувається масове розростання мережі нових кровоносних судин. Так організм намагається замінити пошкоджені шляхи доставки кисню. Нові судини формуються одразу слабкими і також не справляються зі своїм завданням, а тільки породжують нові вогнища крововиливів. Через крововиливи центральна зона внутрішньої оболонки ока набрякає. Відбувається закупорка тромбами дрібних судин і якість зору сильно падає.
4. Діабетична макулопатія: на цьому етапі відбуваються незворотні некротичні процеси. Можливі крововиливи у кришталик. Крововиливи формують багато кров'яних згустків, що навантажують сітківку, деформують її та ініціюють процес її відторгнення. Кришталик втрачає здатність фокусувати світлові промені на макулі та хворий поступово втрачає гостроту зору аж до сліпоти.

Вище, на рисунку 1.3 наведена лише одна з ймовірних категорій класифікації стадій діабетичної ретинопатії, проте часто виділяють лише 3 перші стадії як основні (класифікацією Eva Kohner і Massimo Porta, що прийнято ВООЗ), або ж непроліферативну ретинопатію розписують ще на декілька етапів, як наприклад у [10] та на рисунку 1.4. Зважаючи на зображення, виникає розуміння, що між непроліферативною та проліферативною фазами доволі видима різниця, проте між різновидами ретинопатії визначити цю різницю доволі складно. Саме у цьому випадку доцільно застосовувати штучний інтелект (ШІ) та машинне навчання (МН). Важливо отримати результати як результат бінарної класифікації (є чи немає діабетичної

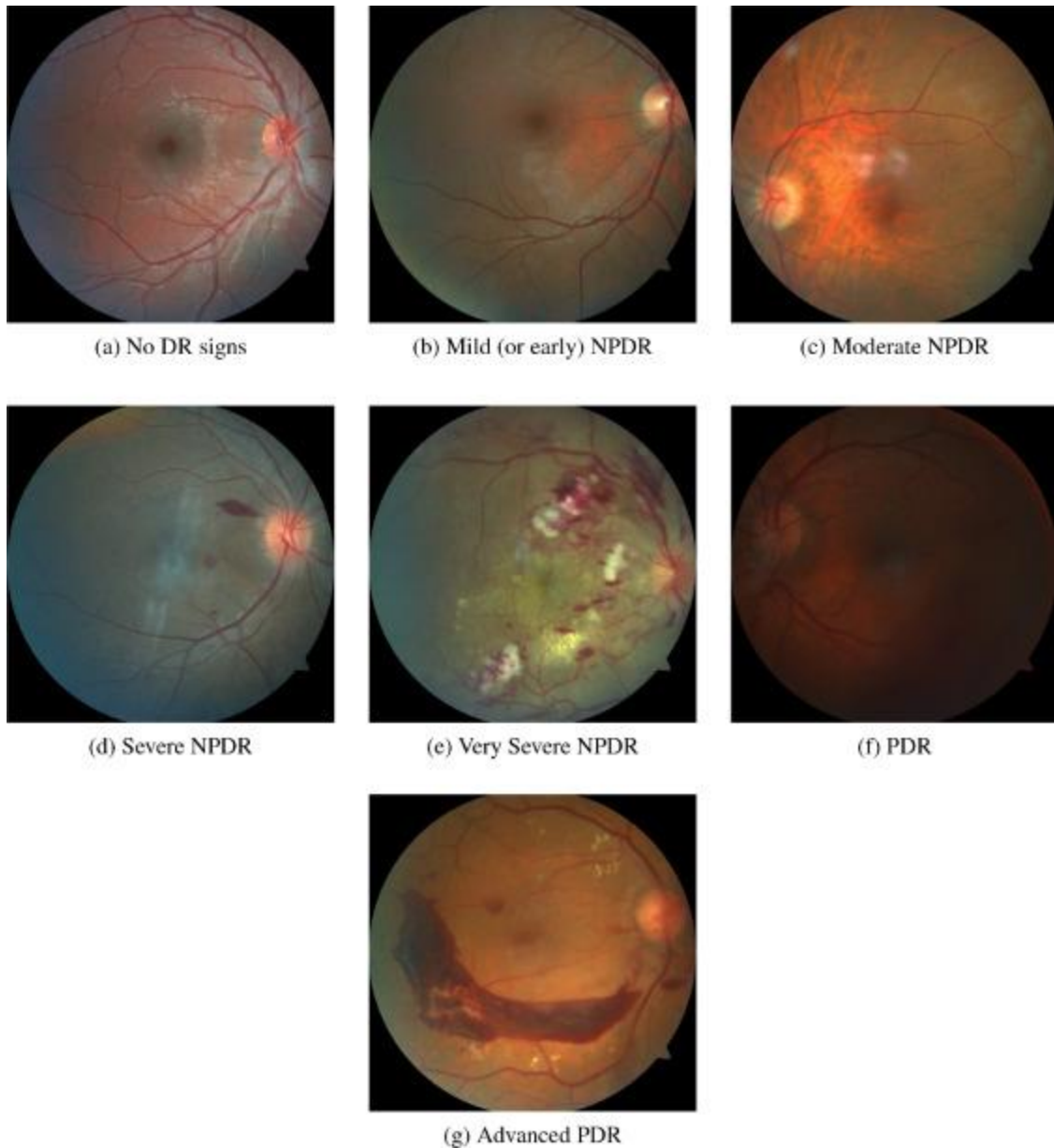


Рисунок 1.4 – Зображення сітківки ока для різних стадій діабетичної ретинопатії [10]

За ступенем ураження сітківки виділяють 3 стадії:

Легка - характерні мікропошкодження судин. При офтальмоскопії визначаються як малі червоні точки. Їх наявність неможливо визначити без спеціального обладнання.

Помірна - збільшується кількість мікропошкоджень, збільшується обсяг вен, проявляються ознаки ретинальної гіпоксії.

Важка - мікроскопічні крововиливи утворюються по всій площині сітківки. Значна частина очних капілярів перестає функціонувати.

Обов'язкові офтальмологічні методи діагностики ДР [11]:

- Визначення гостроти зору (візометрія) і полів зору (периметрія);
- Вимірювання внутрішньоочного тиску (тонометрія);
- Біомікроскопія кришталика і скловидного тіла з допомогою щілинної лампи;
- Офтальмоскопія з розширенням зіниці.

Додаткові офтальмологічні методи діагностики ДР:

- Фотографування судин очного дна за допомогою фундус-камери;
- Флюоресцентна ангіографія судин сітківки;
- Електрофізіологічні методи дослідження для визначення функціонального стану зорового нерва та сітківки;
- Ультразвукова діагностика (УЗД) за наявності значних помутнінь у скловидному тілі та кришталику;
- Гоніоскопія (огляд кута передньої камери ока).

Ключовим механізмом розвитку даного патологічного стану є токсична дія на сітчасту оболонку ока хронічно підвищеного рівня глюкози. На додаток прогрес захворювання може спричинити такі фактори:

1. Куріння. Призводить до посилення гіпоксії (зниження вмісту кисню в організмі), що сприяє спазму церебральних судин.
2. Вагітність. Період вагітності є сприятливим фактором для розвитку хвороби через загальну перебудову обмінних процесів організму. У зв'язку з цим вагітним особливо важливо спостерігатись у офтальмолога.
3. Високий рівень глюкози в крові. Негативний вплив на внутрішньоочні структури також завдає підвищений рівень глюкози та холестерину у крові.
4. Високий артеріальний тиск. Ризик ураження сітківки зростає вдвічі при підйомі діастолічного тиску на кожні 10 мм рт. ст.

5. Вік пацієнтів. Особливо актуальне це питання для людей з першим типом діабету під час статевого дозрівання та літніх людей з тривалим захворюванням на діабет.

1.2 Висновки до розділу 1

Узагальнюючи наведену інформацію, можна зробити висновок, що діабетична ретинопатія є серйозним ускладненням діабету, що впливає на судини сітківки ока та може призвести до порушення зору. Розглядаючи різні стадії захворювання, видно, як прогресує ураження та які конкретні зміни відбуваються в сітківці. Зображення, що відображають кожен етап, слугують важливим інструментом для визначення ступеня тяжкості та вчасної діагностики.

Враховуючи фактори ризику, стає очевидним, що раннє виявлення та ефективний контроль рівня цукру в крові можуть значно знизити ризик розвитку діабетичної ретинопатії. Наукові та медичні дослідження в цьому напрямку є невід'ємною частиною подальших зусиль у покращенні профілактики та лікування цієї серйозної проблеми. Крім того, своєчасна діагностика захворювання сприяє можливості лікування, а для діагностики захворювання потрібні інструменти, що можуть зробити якісне зображення сітківки для подальшого аналізу на наявність хвороби.

2. ОСНОВНІ МЕТОДИ ДОСЛІДЖЕННЯ СІТКІВКИ ОКА. ОТРИМАННЯ ЗОБРАЖЕНЬ

Зазвичай виділяють наступні методи обстеження очного дна:

- оптична когерентна томографія;
- флуоресцентна ангіографія;
- офтальмоскопія;
- фото-реєстрація очного дна.

2.1 Оптична когерентна томографія

Оптична когерентна томографія (ОКТ) і оптична когерентна томографічна ангіографія (ОКТА) є неінвазивними методами візуалізації, що використовують світлові хвилі для формування зображень перерізу сітківки ока. ОКТ аналогічна ультразвуковому зображенню, але замість звуку використовується світло. За допомогою ОКТ можна отримати зображення структури тканини в поперечному перерізі в мікронному масштабі, або ж спостерігати за ним у режимі реального часу.

ОКТ, яка відображає мікроструктуру внутрішнього поперечного перерізу тканин за допомогою вимірювань оптичного зворотного відбиття, була вперше продемонстрована ще в 1991 році [12]. ОКТ-візуалізацію проводили *in vitro* (в контрольованому середовищі) на сітківці ока людини та в атеросклеротичній бляшці (закупорення судин холестерином), як приклади візуалізації в прозорих середовищах зі слабким розсіюванням і непрозорих середовищах із високим розсіюванням. Не так давно прогрес у технології ОКТ зробив можливим зображення непрозорих тканин, що

дозволило застосовувати ОКТ в широкому діапазоні медичних спеціальностей. На сьогоднішній день ОКТ має найбільший клінічний вплив у офтальмології. На рисунку 2.1 можна побачити ОКТ зображення нормальної сітківки ока людини. ОКТ-зображення дозволяє отримати поперечний переріз сітківки з надзвичайно високою роздільною здатністю і дозволяє відрізнити детальні структури [13].

Були проведені клінічні дослідження, щоб дослідити доцільність використання ОКТ для діагностики та моніторингу захворювань і дефектів сітківки, таких як глаукома, макулярний набряк, макулярний отвір, центральна серозна хоріоретинопатія, вікова макулярна дегенерація, епіретинальні мембрани, ямки диска зорового нерва та хоріоїдні пухлини [14].

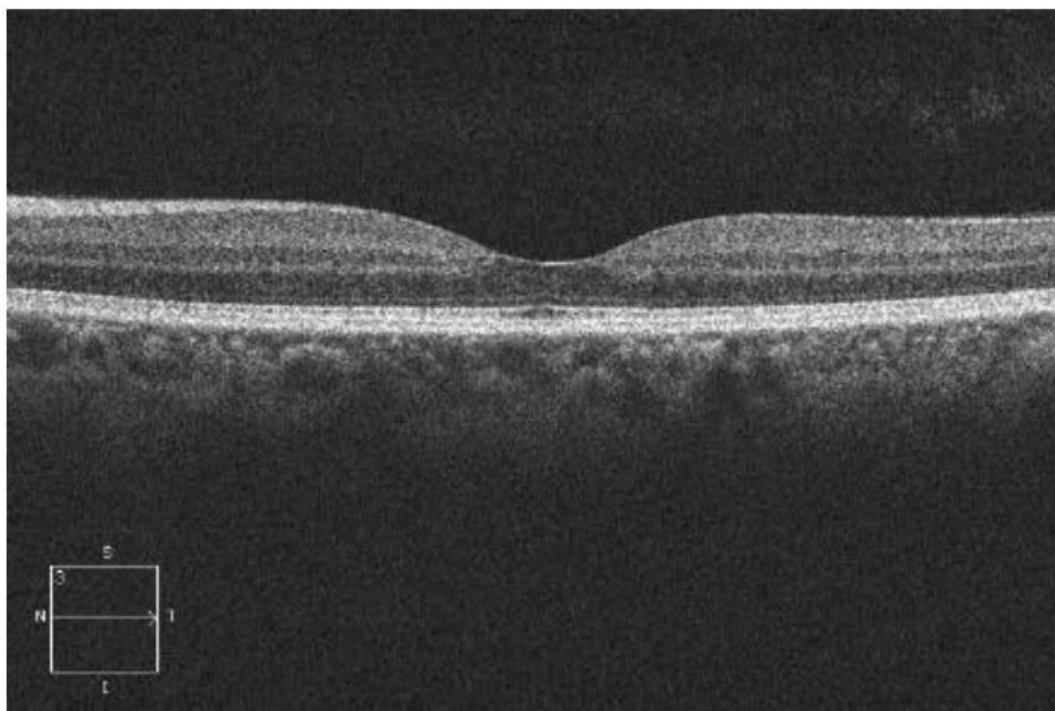


Рисунок 2.1 – ОКТ зображення здорової сітківки ока [15].

ОКТ часто порівнюють із медичним УЗД через подібні принципи роботи. Обидва методи отримання медичного зображення направляють хвилі на досліджувану тканину, де хвилі відбиваються від структури тканини. Хвилі, відбиті назад,

аналізуються та вимірюється їх затримка, щоб виявити глибину, на якій відбулося відбиття. ОКТ використовує світло в ближньому інфрачервоному діапазоні, яке поширюється набагато швидше, ніж ультразвук. Затримки зворотно відбитих хвиль не можна виміряти безпосередньо, тому використовується еталонне вимірювання. Завдяки використанню інтерферометра частина світла спрямовується на зразок, а інша частина надсилається на опорне плече з добре відомою довжиною.

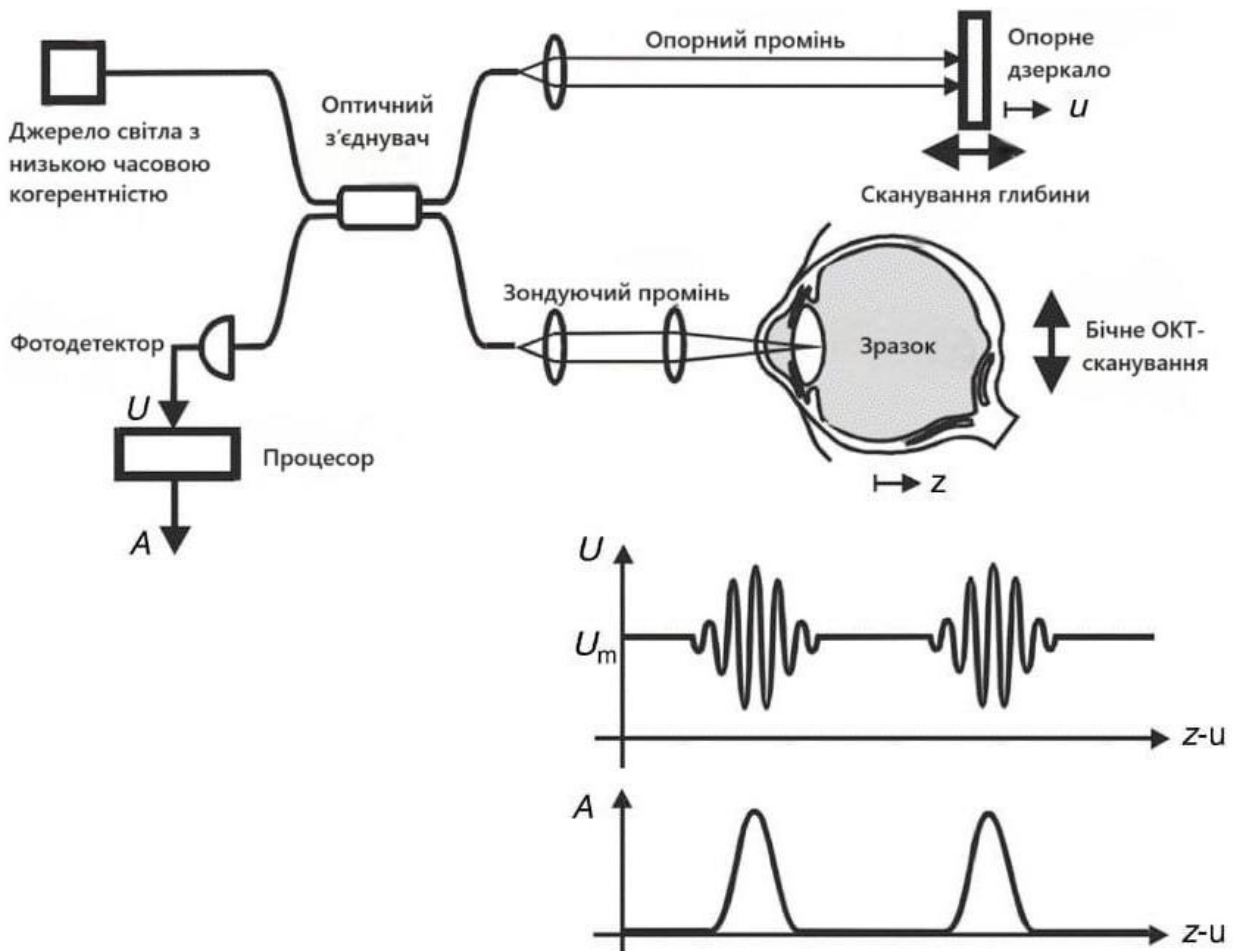


Рисунок 2.2 – Принцип роботи ОКТ [16]

Оптична когерентна томографія працює за допомогою інтерферометрії, яка дає змогу отримувати зображення тканини за допомогою ближнього інфрачервоного світла, а не за допомогою гамма-променів чи ультразвуку. Інтерферометрія працює,

спрямовуючи в око промінь світла, який відбивається тканинами на різній глибині. На основі цих відображень будуються образи. Зображення отримують чіткі через прозоре вікно, наприклад, рогівку. Світло, яке потрапляє в око, є безпечним, тому не завдає шкоди. [17]

2.2 Флюоресцентна ангіографія очного дна

Флюоресцентна ангіографія очного дна (ФАГ) - це інвазивна діагностична процедура при якій виконується серія фотографій обох очей після внутрішньовенної ін'єкції флуоресцеїну натрію. Інтерпретація та розуміння ФАГ допомагають у діагностиці та оцінці різного роду патологій. Тому такий вид діагностики часто рекомендують для виявлення і діагностики захворювань очей, зокрема:

- Макулярний набряк;
- Діабетична ретинопатія;
- Закупорка вен всередині ока;
- Макулярна складка;
- Очна меланома;
- Відслідковування змін захворювання очей з часом;
- Цільові зони лікування.

Сам процес ФАГ триває менше 30 хвилин. Зазвичай все розпочинається з того, що в очі пацієнту капають очні краплі для розширення зіниці. Далі вводять у вену флуоресцеїн. За 10-15 секунд він поширюється по всьому оку і досягає кровоносних судин ока, що змусить їх яскраво сяяти. Коли барвник проходить через сітківку, спеціальна камера робить знімки.

Для фотографування необхідна фундус-камера, оснащена фільтром збудження та бар'єрним фільтром. Кобальтовий блакитний фільтр збудження пропускає синє світло в діапазоні від 465 до 490 нм, і це в діапазоні піку поглинання флуоресцеїну. Бар'єрний фільтр пропускає світло в діапазоні від 520 до 530 нм, що є діапазоном піку випромінювання флуоресцеїну. З появою цифрової ангіографії для обробки отриманих зображень потрібен комп'ютер із програмним забезпеченням.

Доросла доза флуоресцеїну натрію становить 500 мг внутрішньовенно. Потрібен флакон барвника Sodium Fluorescein (2 мл 25%, 2,5 мл 20%, 5 мл 10% або 10 мл 5%). Також потрібен набір голок для вен голови 23 калібру та шприц на 5 мл з голкою для втягування барвника, підлокітник, джгут і спиртовий тампон. Стандартне обладнання екстреної допомоги для лікування анафілаксії завжди має бути готове до початку процедури [18].

Фундус-камера оснащена двома фільтрами. Фільтр збудження (від 465 до 490 нм) пропускає в око лише синє світло для збудження молекул флуоресцеїну барвника. Бар'єрний фільтр (від 520 до 530 нм) пропускає лише зелене світло, випромінюване барвником [19]. Фундус-камера більш детально розглянута в частині 2.4. Фільтр блокує світло будь-якого іншого діапазону, що відбивається від ока. Введення барвника своєчасно узгоджується з процесом фотозйомки.

Нормальні фази ФАГ [20]:

1. Хоріоїдальна фаза: барвнику потрібно від 10 до 15 секунд, щоб досягти сітківки після введення в руку. Спочатку він потрапляє в задні циліарні артерії, і, таким чином, в першу чергу спостерігається заповнення судинної оболонки.
2. Артеріальна фаза: барвник потрапляє в артерії зазвичай через 1-2 секунди після хоріоїдальної фази.
3. Артеріовенозна фаза: спостерігається через 1-2 секунди після артеріальної фази. Барвник надходить у прекапілярні артеріоли, капіляри та посткапілярні венули. У цій фазі можна побачити ламінарний потік, коли барвник потрапляє у великі

- вени. Ламінарний потік видно, оскільки пов'язаний з білком флуоресцеїн відсувається в сторони вільною плазмою та еритроцитами, які течуть у центрі.
4. Венозний період має три фази: ранню, середню та пізню залежно від наповнення барвника. Рання фаза демонструє ламінарний потік з повним венозним наповненням у середній фазі, а потім знижену концентрацію барвника в артеріях у пізній фазі. Пікова фаза ангиографії спостерігається, коли перифокальна капілярна мережа добре визначена.
 5. Рециркуляція: після венозної фази концентрація барвника починає знижуватися в судинах із повним виведенням барвника приблизно через 10 хвилин. У цій фазі видно забарвлення диска зорового нерва, оболонки Бруха, судинної оболонки та склери.

2.3 Офтальмоскопія

Офтальмоскопія – процедура, що передбачає огляд очного дна, включаючи сітківку, зоровий нерв, судинну оболонку, що проводиться за допомогою спеціального інструменту – офтальмоскопу.

Дослідження цієї частини очного яблука, яка розташована позаду циліндричного тіла і кришталика є важливим для оцінки стану ока, діагностики захворювань і пошкоджень зорового нерву, сітківки, а також неврологічних і системних захворювань.

Для проведення офтальмоскопії необхідно розширити зіницю ока. Це досягається за допомогою спеціальних очних крапель, спроектованих для розширення зіниці та тимчасового блокування її закриття під впливом яскравого світла. Велике розширене зіницею дає лікарю можливість спостерігати за "периферійними"

областями сітківки та склоподібного тіла. Це має важливе значення, оскільки багато захворювань очей розпочинаються саме в цих зовнішніх областях сітківки.

Існують три основних варіанти офтальмоскопії:

- **Пряма офтальмоскопія:** під час цього виду огляду пацієнт сидить у темній кімнаті, а лікар використовує ручний офтальмоскоп для освітлення очей яскравим світлом. Цей пристрій, розміром приблизно з тонкий ліхтар, використовує ряд дрібних лінз, які фокусуються на внутрішніх структурах ока та відображають зображення для огляду.

- **Непряма офтальмоскопія:** цей метод часто використовується, коли пацієнт лежить на спинці крісла для огляду. Лікар утримує око пацієнта відкритим та використовує офтальмоскоп, який закріплений на голові, схожий на шахтарську фару. Лікар направляє світло через лінзу, яка збільшує зображення, а потім використовує спеціальні дзеркала та лінзи для огляду очей. Пацієнта просять рухати очима для огляду різних ділянок.

- **Офтальмоскопія зі щілинною лампою:** це, можливо, найпоширеніший тип офтальмоскопії, де пацієнт зазвичай сидить перед офтальмологічним мікроскопом із щілинною лампою. Цей пристрій дозволяє лікареві оглядати різні частини очного яблука з різним збільшенням. Пацієнт сидить прямо в оглядовому кріслі перед щілинною лампою, спираючись підборіддям на опору, щоб голова залишалася нерухомою. Лікар тримає збільшувальну лінзу перед оком і просвічує через неї джерело світла щілинної лампи. Збільшене зображення внутрішньої сторони ока відбивається крізь лінзу і лікар бачить її крізь бінокль щілинної лампи. За допомогою цього методу лікар має можливість збільшити або зменшити збільшене зображення [21].

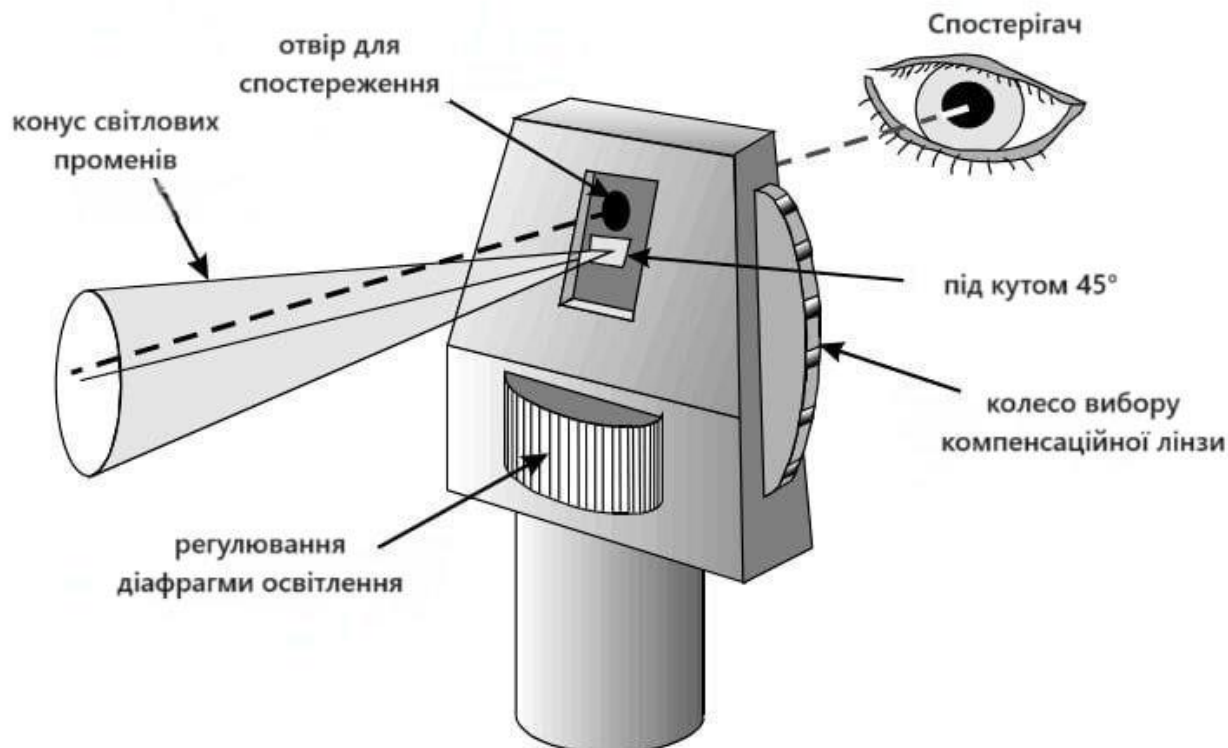


Рисунок 2.3 - Креслення сучасного прямого офтальмоскопа.

На рисунку 2.3 показано сучасний прями́й офтальмоскоп, який має схожу конструкцію з офтальмоскопом Гельмгольца-Бebbіджа. Однак у цій моделі замість дзеркала з отвором, є маленьке дзеркало, яке розташоване під оглядовим отвором. Поряд з оглядовим отвором є ще один отвір діаметром приблизно 3 мм. Це можливо завдяки винаходу електричної лампи розжарювання, яка дозволяє об'єднати джерело світла з рештою офтальмоскопа. Офтальмоскоп також має два коліщатка. Одне з них дозволяє регулювати діаметр, форму та колір променя освітлення, а інше дозволяє вставляти маленькі «компенсуючі» лінзи безпосередньо за оглядовим отвором.

Тепер детальніше розглянемо будову офтальмоскопа. На рисунку 2.4 офтальмоскоп показаний у поперечному розрізі разом із створеною комп'ютером моделлю очей спостерігача та об'єкта. Око обстежуваного знаходиться на відстані 2 см від офтальмоскопа, що є типовою відстанню для спостереження за сітківкою [22].

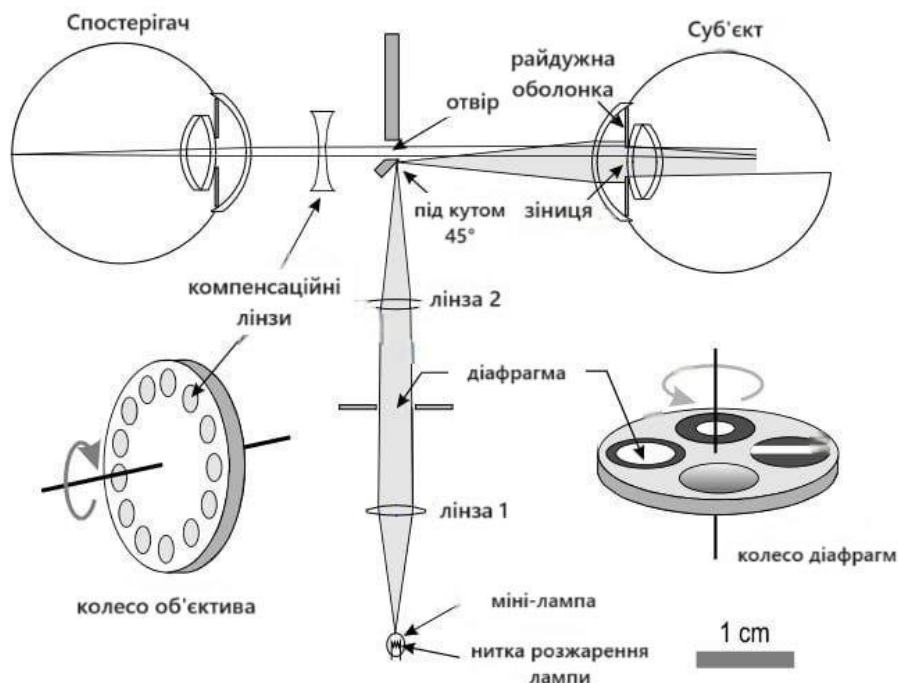


Рисунок 2.4 – Детальна структура прямого офтальмоскопа із зображенням освітлювальної та оглядової оптичних систем

Офтальмоскоп - це прилад для дослідження ока, який використовує систему освітлення з електричною лампою, діафрагмою, лінзами та дзеркалом. Промені світла від лампи збираються першою лінзою та фокусуються другою лінзою до того, як на дзеркалі утворюється зображення нитки розжарення лампи. Діафрагма між лінзами дозволяє використовувати різні кольори та форми світла. На діафрагмі є кілька отворів, які можна вибирати за допомогою горизонтально орієнтованого коліщатка з отворами. Більшість офтальмоскопів мають кілька різних отворів, включаючи два круглих, синю та зелену круглі апертури та щілину. Діафрагма розташована на такій відстані від лінзи, щоб бути у фокусі сітківки об'єкта. Спостерігач може побачити добре сфокусований світловий диск на сітківці об'єкта при використанні круглих апертур. Промені світла від дзеркала утворюють конусоподібний пучок, який

потрапляє в око досліджуваного. Пучок променів проходить через рогівку та передню камеру, а потім досягає сітківки ока.

Розмір видимої частини сітківки залежить від того, наскільки вона освітлюється. Якщо зробити конусну зону світлового випромінювання великого розміру, то буде можливість бачити більшу площу сітківки. Однак на практиці це не завжди вірно. На рисунку 2.5 зображено око людини з зіницею діаметром 2 мм, це типовий розмір, коли око освітлюється яскравим світлом від офтальмоскопа. На лівій стороні рисунку показано велику зону світлового випромінювання. Застосування такої апертури призводить до того, що значна частина світла просто відбивається від райдужної оболонки очей і не попадає на сітківку. У правій частині рисунку показано використання меншої зони світлового випромінювання, де всі промені пройшли через зіницю і освітлюють сітківку. Природна розширена зіниця використовує всю доступну світлову зону, тому використання великої зони освітлення не має переваги. На розмір освітленої області сітківки в основному впливають відстань від офтальмолога та діаметр зіниці. На рисунку 2.6 докладно показано, як кожен з цих параметрів впливає на зображення сітківки. Промені освітлення показані від моменту, коли вони залишають дзеркало в офтальмоскопі (решта системи освітлення опущена для ясності). Велика апертура освітлення (тобто великий конус освітлення) використовується в А, В і С. Праворуч показана видима ділянка сітківки в області диска зорового нерву.

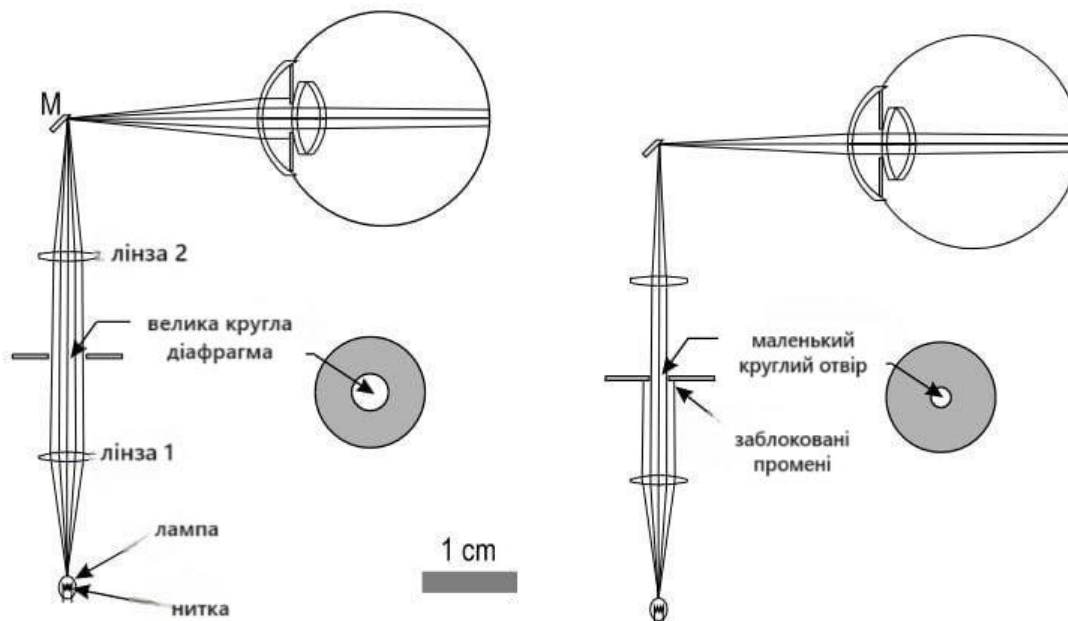


Рисунок 2.5 – Вплив розміру апертури підсвічування на освітлену ділянку сітківки.
Діаметр зіниці 2 мм, око знаходиться на відстані 2 см від офтальмоскопу в обох випадках

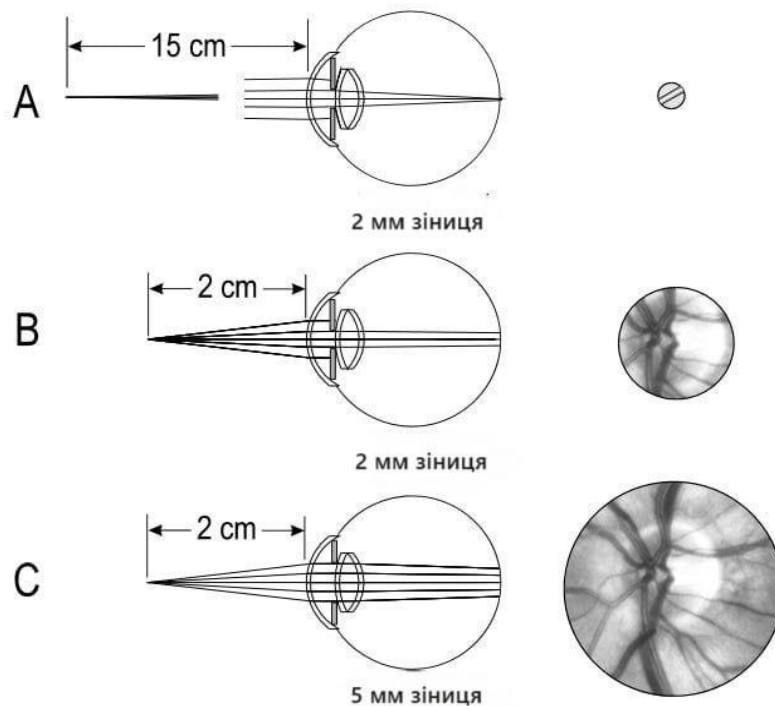


Рисунок 2.6 – Ілюстрація впливу відстані джерела освітлення до ока та діаметра зіниці на площу видимої сітківки

2.4 Фотореєстрація очного дна за допомогою фундус-камери

В попередніх розділах було оглянуто основні методи отримання зображень очного дна, окрім докладного розбору установки фундус-камери та методології отримання зображень з її допомогою. В цьому розділі детально розглянуто цей вид реєстрації сітківки ока, оскільки у практичній частині роботи використовувались саме зображення з фундус-камери.

Фундус-камера - це інструмент, що використовується для фотографування очного дна. Зображення очного дна дозволяє виявляти та відслідковувати різні причини сліпоты, зокрема діабетичної ретинопатії, вікової макулярної дегенерації, глаукоми. Протягом десятиліть якість зображень очного дна та легкість фотореєстрації значно покращились. Зображення очного дна важлива складова для діагностики та лікування різних очних захворювань. Джекман і Вебстер були першими, хто опублікував зображення сітківки ока в 1886 році [23]. Карл Цайс був першою комерційно доступною камерою очного дна в 1926 році. Хенселл і Бісон приєднали електронну спалахову трубку, яка дозволяла направляти світло через зіницю [24].

Камера очного дна є складною системою з кількома лінзами та камерою, яка використовує непрямий офтальмоскоп. Оптика фундус-камери включає кругле дзеркало, набір фільтрів і серію лінз. Світло від оглядової лампи проходить через цю оптику і фокусується у вигляді пучка у формі бублика. Далі світловий промінь падає на кругле дзеркало з центральним отвором, відбивається і проходить через об'єктив у розширену зіницю ока, щоб зробити фотографію очного дна.

Коли промені зображення і освітлювальні промені правильно вирівняні, вони проходять через центр зіниці і потім повертаються до системи камери з одним об'єктивом. Цей процес дозволяє отримати зображення очного дна.

Існують різні типи камер очного дна, такі як iCam, 3nethra, цифрова ретинографія, EasyScan, TRCNW8Fplus, Visucam, Nonmyd7 і Canon CR-2. Вони забезпечують зручність та точність при дослідженні очного дна.

Останні оновлення в цій галузі призвели до розробки фундус-камер широкого та ультраширокого поля. Ширококутна візуалізація дозволяє отримати зображення сітківки під кутом понад 50 градусів, тоді як надширококутне зображення має кут площі понад 100 градусів. Ці технології забезпечують збільшену область візуалізації очного дна і графічні зображення сітківки.

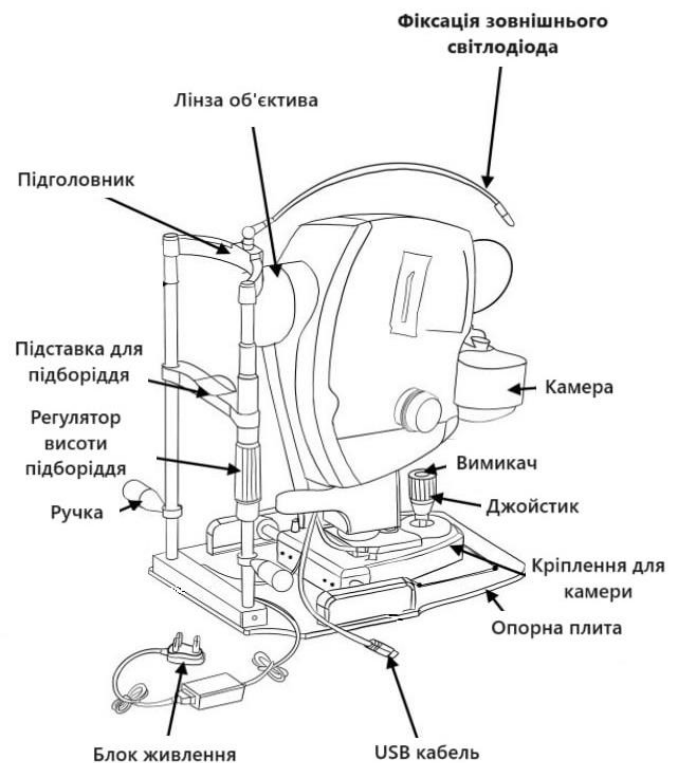


Рисунок 2.7 – Зовнішній вигляд фундус-камери та схема типової установки [24]

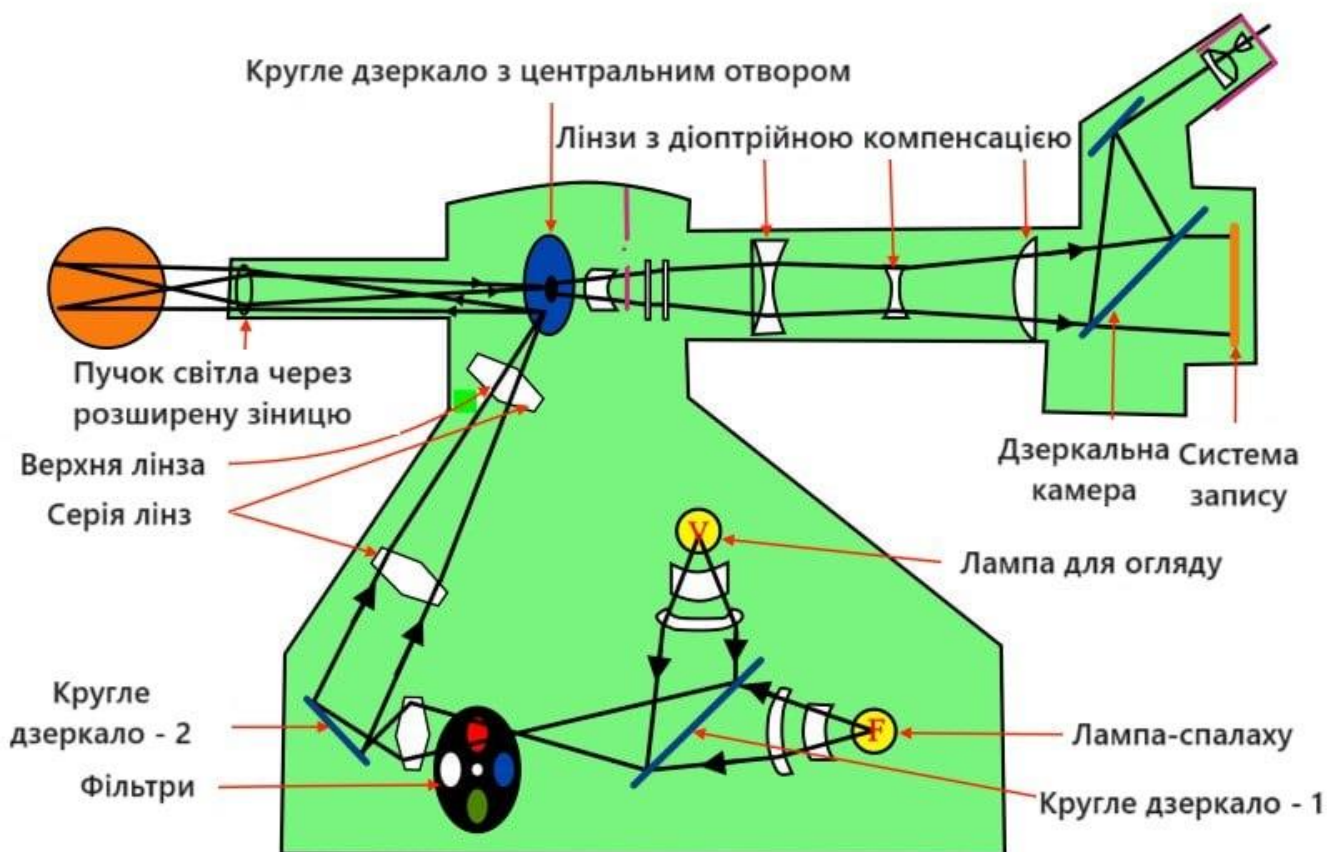


Рисунок 2.8 – Внутрішня будова фундус-камери [24]

Показано, що камери очного дна, які можна використовувати зі смартфонами, є економічно ефективною альтернативою традиційній фотографії очного дна. Ці камери здатні знімати зображення задньої частини ока, навіть із розширеними та нерозширеними зіницями. Ця технологія виявилася особливо корисною для виявлення таких захворювань, як діабетична ретинопатія, вікова дегенерація жовтої плями, ретинопатія недоношених і глаукома, що робить її корисною для телеофтальмології. Ці камери очного дна на основі смартфона були розроблені для виявлення та діагностики захворювань сітківки у віддалених районах, де доступ до спеціалізованого обладнання може бути обмеженим, зокрема й в Україні [25].

Дослідження показали, що використання смартфона для фотографування очного дна є ефективним у виявленні діабетичної ретинопатії. Діагностична точність виявлення проліферативної діабетичної ретинопатії становить 92% [26].

Управління з продовольства та медикаментів США (FDA) схвалили декілька систем візуалізації очного дна на основі смартфонів, зокрема Welch Allyn iExaminer, Remidio Fundus on Phone, D-Eye та iNview. Використання ШІ також розширило можливості візуалізації очного дна, особливо при скринінгу діабетичної ретинопатії.

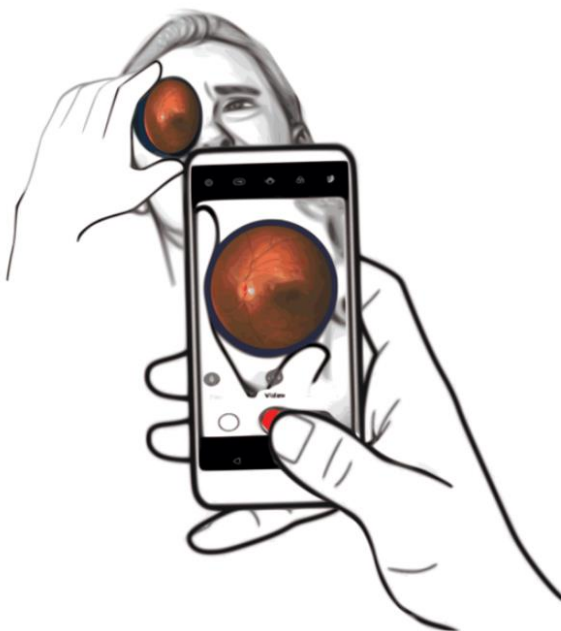


Рисунок 2.9 – Принцип використання смартфонів для отримання зображення сітківки [27]

Одним із прикладів автономної системи штучного інтелекту є IDx-DR, яка була схвалена FDA для виявлення стадій діабетичної ретинопатії, крім легкої або відсутності захворювання. Він має точність 96,1% [28]. Інша схвалена FDA система, EyeArt від Eyeuk, може автономно виявляти не тільки легку діабетичну ретинопатію а й випадки, що загрожують зору.

2.5 Висновки до розділу 2

Розділ присвячений видам апаратури для діагностики діабетичної ретинопатії. Розглянуто основні методи формування зображень, за допомогою яких можна визначити наявність діабетичної ретинопатії у пацієнта. Основну увагу сконцентровано на оптичній складовій апаратури для кращого розуміння логіки роботи приладів. Оскільки у подальшій роботі будуть використовуватись зображення сітківки, що зроблені за допомогою фундус-камери, то для цього інструменту виділено значну частину розділу. Зокрема, варто звернути увагу, що об'ємні установки (наприклад рисунок 2.7) все частіше замінюють на більш прості, щоб все більше поширити технологію отримання цифрових зображень очного дна. Як приклад, на рисунку 2.9 один з видів сучасного отримання зображень сітківки за допомогою спеціальних лінз та звичайної камери телефону. Технологія стрімко розвивається і ймовірно в найближчому майбутньому можливо буде побачити спеціалістів у наших лікарнях, що вміють використовувати такі прості та ефективні технології. Звичайно якість та роздільна здатність зображень зроблених на повноцінній фундус-камери буде кращою, а такі технології матимуть за перевагу швидкість отримання зображення та простоту використання, що також дозволить зменшити навантаження на кваліфіковані кадри.

3. ПОПЕРЕДНЯ ОБРОБКА ЗОБРАЖЕНЬ

3.1 Методи просторової фільтрації

Зазвичай, коли зображення реєструються різними інформаційними системами, вони спотворюються шумами. Це ускладнює їх візуальний аналіз і автоматичну обробку. У деяких задачах обробки зображень саме компоненти зображення можуть виступати в ролі шуму. Наприклад, при аналізі космічного знімка земної поверхні може бути потрібно визначити границі між різними ділянками - лісом і полем, водою і сушею. Деталі зображення всередині цих границь є шумом у рамках цієї задачі.

Шум можна послабити за допомогою фільтрації. Під час фільтрації яскравість кожної точки спотвореного шумом зображення замінюється новим значенням яскравості, яке менш спотворене. Фільтрація зображень може проводитися в просторовій або частотній областях. Просторова область зображення містить пікселі зображення, і відстані між ними відповідають реальним відстаням на поверхні. Частотна область зображення отримується за допомогою перетворення Фур'є. При просторовій фільтрації зображень перетворення виконується над значеннями пікселів. Результатом фільтрації є оцінка корисного сигналу зображення.

Щоб відфільтрувати зображення, використовується маска або фільтр, що представляє собою двовимірне квадратне вікно. Ця маска рухається по зображенню, змінюючи значення кожного пікселя відповідно до коефіцієнтів фільтру. Цей процес називається згорткою або згортанням маски зображення. Якщо маску застосовують до пікселів на межі зображення, потрібно додати "рамку" навколо зображення, що заповнюється за певними правилами.

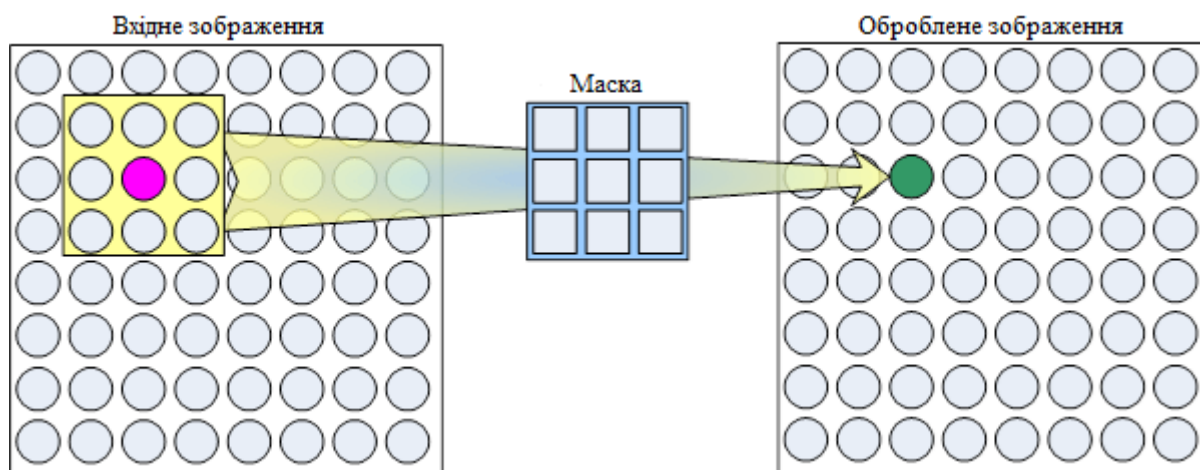


Рисунок 3.1 – Приклад обрахунку значення яскравості пікселя за допомогою маски [29]

Існують різні типи заповнення, такі як заповнення нулями, сталим значенням, значенням найближчого сусіднього пікселя або повторенням останнього рядка (стовпця). Лінійні фільтри, такі як усереднюючий фільтр, є одним із найпоширеніших типів фільтрів. Цей фільтр прибирає шум і покращує якість зображення, але розмиває границі і зменшує розкид значень пікселів. На відміну від усереднюючого, медіанний фільтр призначає кожному пікселю медіанне значення з його околиці [30].



Рисунок 3.2 – Приклад застосування медіанного фільтру.

Загалом, формула медіанного фільтру визначається наступним чином:

$$\text{Im}_{i,j} = \text{med} \left[\widehat{\text{Im}}_{i+s,j+t}; (s,t) \in W \right], \quad (1)$$

де $\text{Im}_{i,j}$ - відновлене зображення в будь-якій точці, $\widehat{\text{Im}}_{i,j}$ - початкове зображення, s,t – індекси певного околу W .

Формулу усереднюючого фільтру зазвичай визначають як:

$$\text{Im}_{i,j} = \frac{\sum_{s,t \in W} \widehat{\text{Im}}_{s,t}}{m \cdot n},$$

де m, n – розміри початкового зображення.

З формули (1) випливають і інші фільтри, що засновані на виборі мінімального чи максимального значень. Фільтр максимуму задається виразом:

$$\text{Im}_{i,j} = \max \left[\widehat{\text{Im}}_{i+s,j+t}; (s,t) \in W \right].$$

Він буде корисним при виявленні найбільш яскравих точок на зображенні. А щоб знайти найменш яскраві точки варто скористатись фільтром мінімуму:

$$\text{Im}_{i,j} = \min \left[\widehat{\text{Im}}_{i+s,j+t}; (s,t) \in W \right].$$

Такі фільтри, крім виявлення найбільш світлих та темних точок на зображенні, також призводять до зменшення уніполярного шуму.

3.2 Еквалізація гистограми яскравості

Гістограма яскравості – це графічне зображення розподілу інтенсивності зображення, або, простіше кажучи, це кількість пікселів для кожного рівня яскравості. На рисунку 3.3 вісь X представляє тонову шкалу, де ліворуч – чорний, праворуч – білий. А вісь Y представляє кількість пікселів для кожного рівня яскравості. Відповідно пік на гістограмі відповідає найбільш поширеному рівню яскравості на зображенні.

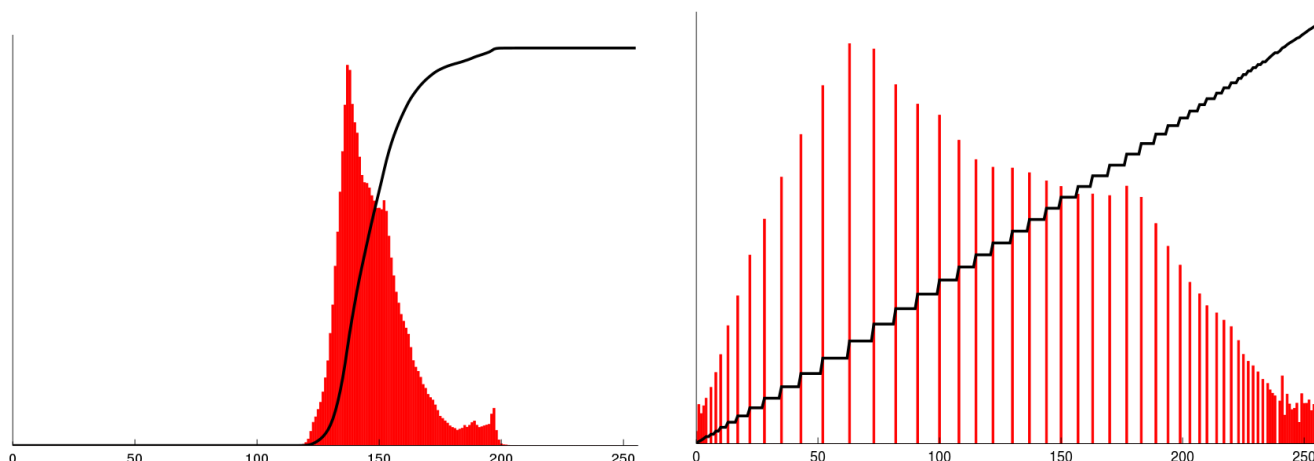


Рисунок 3.3 – Представлення гістограми яскравості зображення до та після еквалізації

Вирівнювання гістограми — це метод обробки зображень, який служить для покращення контрастності. Це досягається шляхом розподілу значень інтенсивності таким чином, щоб найбільш часто зустрічаються значення мали ширший діапазон. При вирівнюванні гістограми загальна контрастність зображення збільшується, особливо у тих областях, де контрастність була низька. Колірна гістограма зображення відображає кількість пікселів для кожного типу кольору. Важливо зауважити, що вирівнювання гістограми не можна застосовувати окремо до кожного колірної компонента, так як це призведе до зміни кольорового балансу. Проте, якщо зображення перед вирівнюванням гістограми перетворити в інший колірний простір (наприклад, HSL/HSV), то можна застосувати алгоритм до каналу яскравості або значення, зберігаючи відтінок і насиченість. Зазвичай вирівнювання гістограми відбувається за формулою:

$$f(x) = (x - x_{\min}) \cdot \frac{255}{x_{\max} - x_{\min}},$$

де $f(x)$ - зображення після еквалізації, x_{\min} , x_{\max} - мінімальне і максимальне значення яскравості пікселя в поточному зображенні, x - значення яскравості поточного пікселя.

3.3 Гамма-корекція

У найпростішому випадку гамма-корекція визначається наступним чином:

$$V_{out} = A \cdot V_{in}^{\gamma},$$

де A – константа, вхідні та вихідні значення є невід’ємними дійсними значеннями.

У звичайному випадку $A=1$. Значення гамми < 1 іноді називають гамою кодування, а процес кодування з цією гамою звать гамма-стисненням, і навпаки, гамма-значення > 1 називається гамма-декодуванням, а процес – гамма-розширенням [31].

Гамма-регулювання є одним з найпростіших глобальних операторів відтворення тону. Якщо зображення надто яскраве або надто темне, його можна зробити приємним, застосувавши гамму більше одиниці (що призведе до темнішого зображення) або менше ніж одиниця (що призведе до більш яскравого зображення) відповідно [32].

Гамма-корекція спочатку була розроблена для компенсації нелінійної реакції моніторів з електронно-променевою трубкою (ЕПТ) на вхідний сигнал. ЕПТ не могли самі посилити вхідний сигнал, тому вихідний сигнал з ПК потрібно було налаштувати, що призвело до стандартної корекції гама і колірному простору sRGB (standard Red Green Blue). Компанії HP та Microsoft розробили такий стандарт для уніфікації колірного спектру для RGB моделі на принтерах та моніторах.

Гамма-кодування дозволяє ефективніше зберігати тони на зображеннях. Це досягається перерозподілом тональних рівнів залежно від сприйняття нашим оком, що вимагає меншої кількості бітів для опису певного тонального діапазону. У разі невикористання гамма-кодування надлишок бітів буде використовуватись для опису яскравіших тонів (коли камера більш чутлива), тоді як недостача бітів залишиться для опису темніших тонів (коли камера менш чутлива). Низька або неправильна гамма-корекція також впливає на баланс кольорів. Занадто велика корекція зсуває кольори в бік нейтрально-сірої палітри, тоді як недостатня корекція зсуває кольори в бік основних кольорів дисплею.

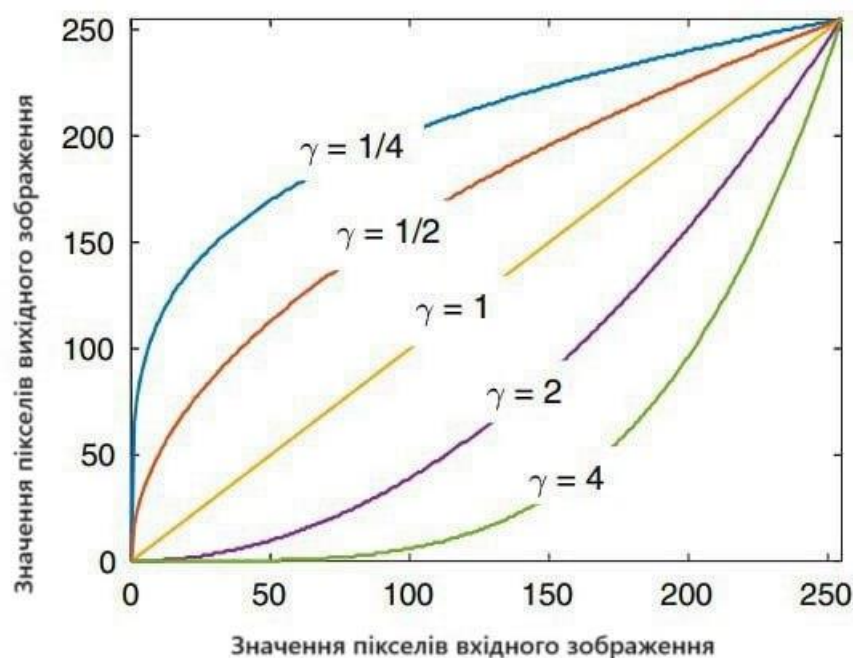


Рисунок 3.4— Відображення функції гамма корекції за різних значень гамми [33]

Як видно з рисунку 3.4, коли зображення має низьку яскравість і контраст, його можна обробити за допомогою гамма-корекції, встановивши значення гами менше 1. Це дозволяє розширити низькі рівні сірого і стиснути високі рівні сірого, коли $\gamma < 1$.

3.4 Методи бінарizaції

Бінарizaція зображення - це процес перетворення зображення з градаціями сірого на чорно-біле, що значно зменшує обсяг інформації, яка міститься в зображенні. Цей процес часто використовується для вилучення об'єктів з зображення, але його успішність залежить від вмісту конкретного зображення. Завдання бінарizaції може бути досить складним, адже не завжди очевидно, які частини зображення можна легко перетворити на чорно-білі.

Процес бінарizaції включає пошук порогового значення на гистограмі, що розділяє зображення на дві частини, кожна з яких представляє один із двох об'єктів або об'єкт і фон. В цьому контексті це називається глобальним пороговим значенням.

Більшість алгоритмів порогового значення базуються на використанні конкретного типу інформації для визначення порогового значення. Ця інформація може бути статистичною, включаючи середнє значення, медіану, ентропію, або вона може подаватися у вигляді характеристик форми гистограми. Один із класичних алгоритмів порогового значення — це алгоритм Отцу, який був представлений Нобуюкі Оцу в 1979 році [34]. Цей алгоритм використовує метод вичерпного пошуку порогу бінарizaції для визначення порогового значення.

Подібно багатьом аспектам обробки зображень, якість результату бінарizaції зображення безпосередньо залежить від якості вихідних даних на зображенні. Інколи здається, що зображення легко піддаватиметься бінарizaції, але це не зовсім так, оскільки алгоритми бачать речі не так як людина. Людська система зору отримує повну картину того, на що вона дивиться, і може інтерпретувати це в контексті, у той час як комп'ютер отримує просто набір пікселів.

Метод Отцу є популярним способом глобальної бінарizaції зображень. Цей метод визначає поріг t , який мінімізує середню помилку сегментації, тобто середню

помилку в прийнятті рішення про те, чи належать пікселі зображення об'єкту чи фону. Яскравості пікселів зображення розглядаються як випадкові величини, їх гістограму можна розглядати як оцінку щільності розподілу ймовірностей. Якщо густини розподілу ймовірностей відомі, то можна визначити оптимальний поріг (у сенсі мінімізації помилки) для сегментації зображення на два класи: об'єкти (c_0) та фон (c_1) [34].

Гістограма формується на основі значень $p_i = \frac{n_i}{N}$, де N - загальна кількість пікселів на зображенні, n_i - кількість пікселів з рівнем яскравості i . Поріг t може приймати цілі значення від 0 до $L = \max$. З використанням гістограми всі пікселі розділяються на "корисні" (об'єкт) та фонові. Кожному типу відповідають відносні частоти W_0 та W_1 :

$$W_0(t) = \sum_{i=1}^t p_i;$$

$$W_1(t) = \sum_{i=t+1}^L 1 - W_0(t).$$

Далі рахуємо середні рівні кожного виду зображення, з урахуванням:

$$\mu_0(t) = \sum_{i=1}^t 1 - \frac{ip_i}{W_0};$$

$$\mu_1(t) = \sum_{i=t+1}^L t + \frac{ip_i}{W_1}.$$

Дисперсія в межах класу визначається за формулами:

$$\delta_0^2(t) = \sum_{i=1}^t (i - \mu_0(t))^2 \frac{p_i}{W_0};$$

$$\delta_1^2(t) = \sum_{i=t+1}^L (i - \mu_1(t))^2 \frac{p_i}{W_1}.$$

Загальна дисперсія буде сумою дисперсії в межах класу та між класами:

$$\delta^2(t) = \delta_0^2(t) + \delta_1^2(t).$$

Потім розраховується максимальне значення для оцінки якості поділу зображення, що й відповідає пороговому значенню:

$$\eta(t) = \max \left(\frac{\delta_{кл}^2(t)}{\delta^2(t)} \right).$$

Даний метод є оптимальним рішенням для визначення глобального порогу на зображенні. Його простота в реалізації та здатність адаптуватися до різноманітних зображень, з вибором оптимального порогу, роблять його ефективним у більшості сценаріїв. Крім того, однією з переваг методу Отцу є час виконання. З іншого боку, він має свої недоліки, такі як чутливість до шуму на зображенні та обмеження сегментації зображення лише для одного класу.

Адаптивні методи бінаризації або методи, що працюють локально, розділяють зображення на різні області, для кожної з яких необхідно визначити поріг, базуючись на інтенсивності пікселів. Ці алгоритми припускають поділ зображення на блоки з певним розміром, де розмір блоку повинен бути мінімальним, але достатнім для збереження особливостей та деталей зображення. У той же час блоки повинні бути достатньо великими, щоб мінімізувати вплив шумів на результат. Використання функції згладжування у адаптивній бінаризації дозволяє досягти задовільного результату без потреби в додаткових фільтрах. До найбільш популярних адаптивних методів бінаризації відносять метод Ніблека, Саувола та метод Ніка [35].

Методі Ніблека був запропонований ще в 1985-му році [36]. Порогове значення кожного вікна визначається на основі середнього значення m , стандартного відхилення σ , значень в пікселях цього вікна:

$$T = m + k \cdot \sigma,$$

де $k = -0.2$ за пропозицією Ніблека, а розмір вікна заздалегідь визначається користувачем.

З досвіду експериментів, цей метод доволі добре ідентифікує об'єкти, але генерує чорний шум в порожніх вікнах.

Метод Сауволи був запропонований у 1997-му році [37]. Цей підхід фактично успадкований від Ніблека, він може подолати проблему чорного шуму, а його порогова формула має вигляд:

$$T = m \cdot \left(1 - k \cdot \left(1 - \frac{\sigma}{R} \right) \right),$$

де k - контрольований коефіцієнт з діапазону $[0,2; 0,5]$, R - попередньо визначене значення рівня сірого зображення.

Автор пропонує встановити коефіцієнти на рівні $k = 0.2$; $R = 125$, проте цей метод менш ефективний, коли контраст зображення між об'єктом та фоном відносно малий.

Метод Ніка запропонований у 2010-му році [38]. Він намагався вирішити проблему низького контрасту, знизивши порогове значення, а кінцева формула має вигляд:

$$T = m + k \cdot \sqrt{\frac{(\sum p_i^2 - m^2)}{NP}},$$

де k - контрольований коефіцієнт з діапазону $[-0.2; -0.1]$, p_i - сірошкальне значення пікселя, NP - загальна кількість пікселів у зображення.

Автор пропонує значення $k = -0.1$. Деякі джерела підтверджують, що метод дає найкращі результати порівняно з попередніми методами [39]. Проте неможливо сказати, що даний метод вирішив проблему низької контрастності.

3.5 Методи виділення границь зображення

Краї на цифровому зображенні представляють собою важливі зміни інтенсивності, утворюючи межу між різними областями. Їх можна класифікувати як горизонтальні, вертикальні та діагональні. Виявлення країв – це процес, який включає математичні методи пошуку точок на зображенні, де чітко змінюється яскравість пікселів, або ми можемо сказати, що це техніка обробки зображень для пошуку меж об'єктів на зображеннях. Він працює, виявляючи перепади яскравості. Така сегментація полегшує аналіз та збереження структурних властивостей. Оператори виявлення меж можуть бути градієнтними, використовуючи перші похідні, такі як оператори Собеля, Превітта та Роберта, або такі, що базуються на операторі Гауса, обчислюючи другі похідні, наприклад, детектор країв Canny та лапласіан Гаусса. Перші похідні визначаємо за формулами:

$$\frac{\partial f}{\partial x} = f(x+1) - f(x);$$

$$\frac{\partial f}{\partial y} = f(y+1) - f(y).$$

Тобто обчислюємо зміну інтенсивності пікселів по відповідній змінній. Другі похідні матимуть вигляд:

$$\frac{\partial^2 f}{\partial x^2} = f(x+1) + f(x-1) - 2f(x);$$

$$\frac{\partial^2 f}{\partial y^2} = f(y+1) + f(y-1) - 2f(y).$$

У такому випадку шукають перетини нуля в похідних другого порядку зображень, щоб знайти ребра.

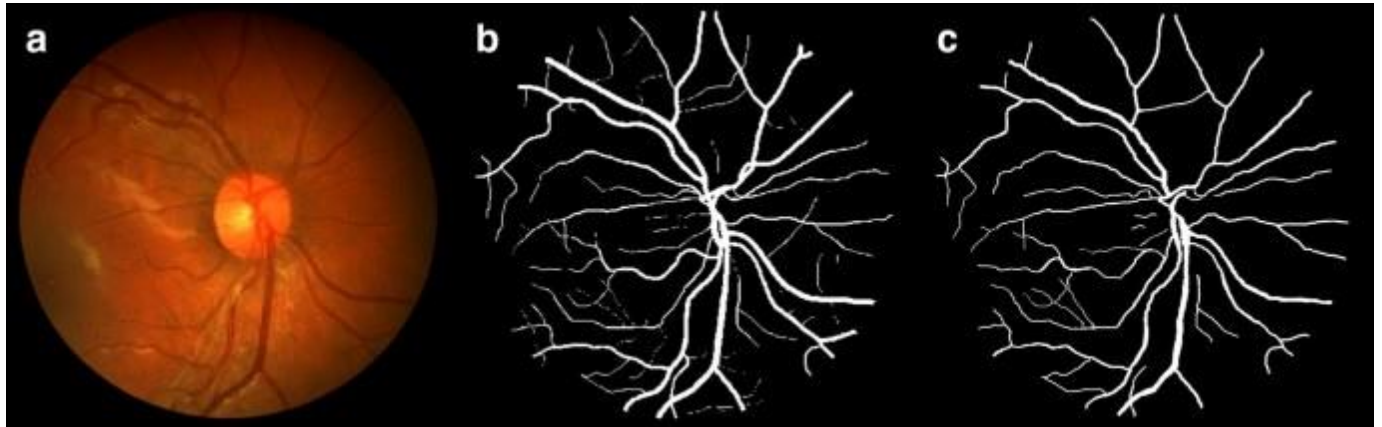


Рисунок 3.5 – Приклад використання методів виділення границь на зображенні сітківки ока [1]

3.5.1 Оператор Кенні

Джон Ф. Кенні займався математичним вивченням проблеми розробки оптимального фільтра за критеріями виділення, локалізації та мінімізації численних відгуків відносно одного краю. Основною метою було створення детектора, який реагує на контури, при цьому ігноруючи неправильні межі. Кенні ввів поняття Non-Maximum Suppression, що означає, що точки границь визначаються як пікселі, де градієнт досягає локального максимуму в напрямку вектора градієнта. Попри те, що з публікації Кенні у 1986 році [40] пройшло 37 років, його запропонований фільтр залишається актуальним і використовується досі.

Алгоритм складається із чотирьох кроків [41]:

- Згладжування. Розмиття зображення для видалення шуму.

Видаляємо небажаний шум, для цього зазвичай використовують фільтр Гауса [42]. Для прикладу можемо взяти ядро розміром 5×5 :

$$K = \frac{1}{159} \begin{bmatrix} 2 & 4 & 5 & 4 & 2 \\ 4 & 9 & 12 & 9 & 4 \\ 5 & 12 & 15 & 12 & 5 \\ 4 & 9 & 12 & 9 & 4 \\ 2 & 4 & 5 & 4 & 2 \end{bmatrix}.$$

- Пошук градієнтів. Границі відмічаються там, де градієнт зображення набуває максимального значення.

Градієнт уздовж напрямків x та y обчислюється за допомогою масок згортки 3×3 :

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, \quad G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}.$$

Далі ще розраховується сила градієнта та напрямок країв:

$$G = \sqrt{G_x^2 + G_y^2};$$

$$\theta = \arctan\left(\frac{G_y}{G_x}\right).$$

- Приглушення не-максимумів. Тільки локальні максимум відмічаються як межі.
На цьому кроці пікселі, що не вважаються частиною краю (поодинокі) видаляють, щоб залишились лише тоні лінії, що містять пікселі, які вважаються частиною краю.
- Гістерезис.
На цьому етапі використовують два пороги, які називають верхнім та нижнім.
 - Якщо значення градієнта пікселя менше за нижнє порогове, то піксель відхиляється;
 - Якщо значення пікселя перевищує верхнє порогове значення, то піксель вважається граничним.

- Якщо значення градієнта пікселя знаходиться між нижнім і верхнім пороговими значеннями, то піксель буде прийнято, за умови, що він підключений до пікселя, що перевищує верхній поріг.

До переваг цього методу відносять хорошу локалізацію, виділяє функції зображення, не змінюючи їх та він менш чутливий до шуму. Проте серед недоліків звертають увагу на складні обчислення та високі затрати часу, а також помилкові переходи через нуль [43].

3.5.2 Оператор Собеля

Оператор Собеля - це дискретний диференціальний оператор для виявлення границь, що поєднує в собі гаусівське згладжування та диференціювання [44]. Використовується для обчислення приблизного значення яскравості зображення. Оператор Собеля вводить концепцію ваги на основі оператора Превітта, вважаючи, що відстань між сусідніми точками впливає на поточний піксель по-різному. Чим ближче піксель, тим сильніший вплив поточного пікселя, щоб підвищити чіткість зображення та виділити контури країв.

Оператор Собеля визначає край згідно з різницею у відтінках сірого верхнього та нижнього пікселів, а також лівої та правої сусідніх точок, досягаючи максимального значення на краю. Він має згладжуючий ефект на шум та забезпечує більш точну інформацію щодо напрямку краю. Оскільки оператор Собеля поєднує в собі гаусівське згладжування та диференціювання, результат виявляє значну стійкість до шуму. Коли немає вимоги щодо високої точності, оператор Собеля є найбільш часто використовується для виявлення границь.

Оператор виявлення краю Sobel реалізований як сума двох спрямованих фронтів. І отримане зображення є односпрямованим контуром вихідного зображення [45].

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, \quad G_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

Для градієнту використовують стандартну формулу:

$$G = \sqrt{G_x^2 + G_y^2}.$$

Проте, щоб збільшити швидкість обчислень, часто обчислюється лише приблизна величина:

$$|G| = |G_x| + |G_y|.$$

Як переваги даного методу виділяють простоту та економічність обчислень, а також легкість в пошуку рівних країв. А до недоліків відносять чутливість до шуму, низьку точність, помилки при діагональному напрямку країв та невідповідність результатів при виявленні товстих та нерівних країв.

3.5.3 Оператор Лапласа

Одним з найпопулярніших операторів другого порядку є оператор Лапласа. Лапласіан функції $f(x, y)$ позначається $\nabla^2 f(x, y)$ і він визначається:

$$\nabla^2 f(x, y) = \frac{\partial^2 f(x, y)}{\partial x^2} + \frac{\partial^2 f(x, y)}{\partial y^2}.$$

Оператор Лапласа також можна представити у вигляді маски згортки розміром 3×3 (позитивний та негативний оператор Лапласа):

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \text{ або } \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

Оператор Лапласа також є похідним оператором, який використовується для пошуку країв у зображенні. Основна відмінність між операторами Лапласа та іншими операторами полягає в тому, що всі вони є похідними масками першого порядку, але Лапласіан є маскою похідних другого порядку. У цій масці розрізняють ще дві класифікації: одна — позитивний оператор Лапласа, а інша — негативний оператор Лапласа.

Оператор Лапласа не виділяє ребра в якомусь конкретному напрямку, але він виділяє внутрішні та зовнішні краї. Де позитивний оператор використовується для виділення зовнішніх країв, а негативний оператор – для внутрішніх країв. Якщо використовувати позитивний оператор, то від оригінального зображення потрібно буде відняти результуюче, а якщо негативний оператор, то відповідно до вхідного зображення потрібно додати результуюче [46].

Перевагами оператора Лапласа найчастіше називають простоту його реалізації, виявлення раптових змін у інтенсивності та використання ядер різних форм, що забезпечує гнучкість у виявленні різних типів країв.

Серед основних недоліків зазвичай говорять про чутливість до шуму, низьку роздільну здатність та залежність від ядра, що призводить до відсутності універсальності [43].

3.5.4 Оператор Робертса

Оператор Робертса, також відомий як алгоритм перехресного диференціалу, це градієнтний алгоритм, заснований на перехресному диференціалі, який виявляє лінії країв, обчислюючи локальний диференціал [47]. Його часто використовують для обробки крутих зображень з низьким рівнем шуму. Коли край зображення близький до плюс-мінус 45 градусів, ефект обробки цим алгоритмом є більш точним.

У перехресному операторі Роберта значення пікселів у кожній точці представляють абсолютну величину вхідного зображення в цій точці. Він використовує дві перехресні згортки, що слугують для апроксимації градієнта по x та по y . Оператори виглядають так:

$$G_x = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}; \quad G_y = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}.$$

Градієнт та приблизне його значення визначають за тими ж формулами, що й в операторі Собеля.

До переваг відносять простоту та збережені точки діагонального напрямку, а до недоліків – чутливість до шуму та низька точність.

3.5.5 Оператор Превітта

Оператор Превітта був створений Джудіт М. С. Превітт [48] для виявлення країв на зображеннях. Цей оператор виявляє обидва типи країв: горизонтальні (вздовж осі X) та вертикальні (вздовж осі Y). Маска Превітта, яка використовується для цього, визначає край там, де відбувається різка зміна інтенсивності пікселів. Краї визначаються як локальні максимуми або мінімуми у графічному представленні результату маски Превітта. Цей оператор більше акцентує увагу на пікселях, що розташовані ближче до центру маски.

Похідні маски мають три властивості: вага - визначає точність виявлення країв, протилежний знак у масці (+ і -), і сума значень маски, яка дорівнює нулю. Оператор Превітта надає дві маски: одну для виявлення горизонтальних країв та іншу для вертикальних.

$$G_x = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}, \quad G_y = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}.$$

Для вертикального напрямку центральний стовпець нульовий і не включений. Обчислюється лише різниця між значеннями правого та лівого пікселів навколо цього краю. Це збільшує інтенсивність країв, і вони покращуються порівняно з вихідним зображенням. В той же час маска горизонтального напрямку обчислює різницю між інтенсивністю пікселів певного краю. Оскільки центральний рядок маски складається з нулів, він не включає початкові значення краю на зображенні, а обчислює різницю інтенсивності пікселів вище та нижче певного краю. Таким чином збільшується зміна інтенсивності, що робить край більш помітним. В цілому сам оператор дуже схожий до оператора Собеля.

До переваг оператора Превітта відносять хорошу ефективність у визначенні вертикальних та горизонтальних країв і визначення орієнтації зображення. В той час як основними недоліками називають фіксовану величину коефіцієнту, посередню точність та високу чутливість до шуму.

3.6 Висновки до розділу 3

Цей розділ було присвячено основним методам обробки зображень, що використовуються саме для медичних зображень, зокрема й зображень сітківки ока.

Такі методи зазвичай використовують для виділення кровоносних судин ока на зображеннях сітківки з фундус-камери [49] або для пошуку аномалій на ОКТ зображеннях [50]. Попри те, що більшість методів були винайдені ще в минулому столітті вони все ще актуальні та часто використовуються в різного роду наукових роботах.

В практичному розділі наведені методи будуть використані для попередньої обробки зображень перед навчанням моделі нейронної мережі.

4. МАШИННЕ НАВЧАННЯ

Машинне навчання (МН) представляє собою галузь наукових досліджень, яка займається розробкою алгоритмів і статистичних моделей, що дозволяють комп'ютерним системам виконувати конкретні завдання без прямого програмування. Алгоритми навчання застосовуються у різних областях та щоденних додатках. Наприклад, при використанні пошукової системи, такої як Google, для пошуку в інтернеті, успіх цього процесу частково пояснюється використанням алгоритмів машинного навчання, які навчилися ефективно сортувати веб-сторінки.

Ці алгоритми застосовуються в різноманітних областях, таких як аналіз даних, обробка зображень та інші. Однією з основних переваг використання машинного навчання є можливість автоматичного виконання завдань після того, як алгоритм вивчить, як їх вирішувати.

Впродовж свого еволюційного розвитку людство виготовляло різноманітні інструменти з метою полегшення виконання різних завдань. Творчий потенціал людського мозку призвів до створення різноманітних машин, що значно спростили повсякденне життя. Ці машини дозволили людям задовольняти різноманітні потреби, включаючи подорожі, промисловість та обчислювальну техніку [51].

4.1 Машинне навчання з вчителем

Навчання з учителем (*англ.* Supervised Learning) є видом машинного навчання, що полягає у створенні функції, яка визначає відповідність між вхідними та вихідними даними на основі пар прикладів вхід-вихід. Цей тип навчання визначає

функцію з маркованих навчальних даних, які представляють собою набір прикладів для навчання. Алгоритми машинного навчання з учителем потребують зовнішньої допомоги для виставлення міток.

У процесі навчання алгоритми використовують навчальний та тестовий набори даних. Навчальний набір включає в себе вихідні змінні, які потрібно передбачити або класифікувати. Алгоритми вивчають закономірності з навчального набору та застосовують їх до тестового набору для прогнозування або класифікації [51].

Є два основні завдання машинного навчання з учителем: класифікація і регресія. Мета класифікації полягає в тому, щоб спрогнозувати мітку класу, яка являє собою вибір із заздалегідь визначеного списку можливих варіантів. Мета регресії полягає в тому, щоб спрогнозувати безперервне число або число з плаваючою точкою, якщо висловлюватись у термінах програмування, або дійсне число, якщо говорити мовою математичних термінів. Загалом великим мінусом такого виду навчання є формування вхідного масиву з об'єктів та відповідей, за допомогою яких алгоритм зміг би навчатись.

Найчастіше такий вид навчання використовується [52]:

- Виявлення шахрайства - використовується для ідентифікації аномалій та шахрайської діяльності у фінансових транзакціях.
- Сегментація зображень - допомагає розпізнавати та класифікувати об'єкти на зображеннях для подальшого аналізу.
- Медична діагностика - використовується для класифікації зображень та діагностики захворювань на основі медичних даних.
- Виявлення спаму - застосовується для фільтрації та виявлення спам-повідомлень у електронній пошті.

4.2 Машинне навчання без вчителем

Навчання без учителя (*англ.* Unsupervised Learning) – це метод машинного навчання, у якому для корекції параметрів алгоритму, який навчають, не використовують цільову функцію, тобто у об'єктів не буде відповіді і тоді алгоритм повинен сам, на основі схожості параметрів, вчитись і шукати відповідь. Алгоритми машинного навчання без вчителя зазвичай використовуються для обробки даних, які не містять міток, тобто ситуацій, коли правильна відповідь невідома. Це ускладнює оцінку якості роботи моделі, оскільки немає чітко визначених відповідей для порівняння. Внаслідок цього такі алгоритми часто використовуються для дослідницьких завдань, коли важливо краще вивчити вміст даних. Основним завданням для такого методу навчання є задача кластеризації – розбиття вхідних даних на кластери (групи) за певними параметрами та задача зменшення розмірності.

Алгоритми кластеризації призначені для розбиття даних на групи, або кластери, які об'єднують схожі між собою елементи. Головна мета - розділити дані так, щоб елементи в одному кластері були дуже схожі, тоді як елементи в різних кластерах відрізнялися. Подібно до алгоритмів класифікації, алгоритми кластеризації намагаються призначити кожній точці даних номер кластера, до якого вона належить [53].

Основної метою алгоритмів зменшення розмірності є представлення даних в просторі меншої розмірності, ніж вхідні, при цьому зберігаючи важливі характеристики повного набору даних [54].

Такий вид навчання зазвичай використовується в [52]:

- Виявлення аномалій – використовується для виявлення виняткових або непередбачуваних подій в даних.
- Мережевий аналіз – вивчає структуру та взаємодії в графічних або соціальних мережах.
- Рекомендаційні системи – надає рекомендації користувачам на основі їх попередніх взаємодій та інтересів.

- Розкладання сингулярних значень – використовується для розкладання матриць та виявлення важливих ознак в даних.

4.3 Навчання з підкріпленням

Навчання з підкріпленням (*англ.* Reinforcement Learning) — це галузь машинного навчання, в якій модель приймає рішення та вчиться взаємодіяти в середовищі, максимізуючи винагороду чи досягаючи певної мети. Відмінною рисою є те, що, на відміну від навчання з вчителем, де дані надають ключ до правильних відповідей, навчання з підкріпленням вимагає, щоб агент самостійно визначав, які дії призводять до оптимальних результатів. Без наявності конкретного набору навчальних даних, агент вчиться на власних взаємодіях і експериментах, долаючи завдання через проби та помилки.

Навчання з підкріпленням є галуззю науки прийняття рішень, спрямованих на вивчення оптимальної поведінки в середовищі з метою максимізації винагороди. У RL дані акумулюються завдяки системам машинного навчання, які використовують метод проб і помилок. Дані тут не є статичним набором вхідних даних, а замість цього формуються під час взаємодії з середовищем.

Використовуючи алгоритми, які навчаються на основі отриманих результатів, RL розв'язує завдання, визначаючи, яку дію варто виконати далі. Після кожної дії алгоритм отримує зворотний зв'язок, що допомагає йому оцінити, чи був вибір правильним, нейтральним або неправильним. Цей підхід ефективний для використання в автоматизованих системах, які повинні приймати безліч невеличких рішень без прямого втручання людини [55].

Такий вид навчання зазвичай використовується в [52]:

- Робототехніка – впровадження позитивного або негативного підкріплення може вдосконалити робототехнічні системи, забезпечуючи їхню більшу ефективність та адаптацію.
- Безпілотні автомобілі – використання навчання з підкріпленням у програмах для безпілотних автомобілів дозволяє вдосконалювати навички автономного водіння через взаємодію з навколишнім середовищем.
- Відеоігри – в навчанні агентів у відеоіграх з підкріпленням застосовуються для покращення та адаптації до різних сценаріїв гри.

4.4 Нейронні мережі

Розглядаючи сферу машинного навчання, неможливо обійти увагою нейронні мережі. Цей вид машинного навчання, також відомий як штучні нейронні мережі, фактично є комп'ютерною імплементацією мозку. Це система взаємодіючих та з'єднаних штучних нейронів, що використовуються на базі порівняно простих процесорів. Кожен процесор регулярно обмінюється сигналами з іншими та спрямовує сигнали до інших процесорів. Всі ці процесори утворюють одну мережу і спроможні виконувати складні завдання. Зростання Інтернету та процеси глобалізації призвели до експоненціального збільшення обсягу інформації, обробка якої фізично важлива для людини. Нейронні мережі застосовуються в аналізі та класифікації даних за заданими параметрами, у формуванні аналітичних прогнозів, порівнянні та розпізнаванні ідентичних даних. Вони використовуються як альтернатива існуючим алгоритмам у сферах машинного перекладу, розпізнавання мови та музики, обробки зображень та визначення об'єктів на фото та відео [56].

4.4.1 Згорткові нейронні мережі

Згорткові нейронні мережі (ЗНМ) (*англ.* Convolutional Neural Network (CNN)) - це спеціальний вид штучних нейронних мереж, які використовуються для розпізнавання зображень безпосередньо з піксельних даних. CNN засновані на подібностях зорової кори людського мозку і можуть досягати дуже хороших результатів у завданнях комп'ютерного зору. Перевага CNN полягає в тому, що їм не потрібно попередньо витягати ознаки зображень. Система сама навчається виділяти ознаки, використовуючи згортку зображень та фільтри для створення інваріантних ознак, які передаються далі для подальшого аналізу [57].

Згорткові нейронні мережі мають декілька основних компонентів [58]:

1. Згортковий шар (convolutional layer): В цьому шарі нейрони впорядковані у вигляді карт ознак (features). Кожен нейрон в карті фільтрує (зортає) вхідні дані для виявлення особливих ознак. Різні карти відповідають різним типам ознак.

2. Перехідний шар (pooling layer): В цьому шарі розмір зображення зменшується шляхом заміщення активацій поруч розташованих нейронів їх максимумом або середнім значенням. Це допомагає зменшити кількість параметрів і значення, а також забезпечує інваріантність до масштабу та пересування.

3. Повнозв'язаний шар (fully connected layer): В останньому шарі карти ознак згорткового шару згладжуються і з'єднуються з класифікатором. Це робить нейронну мережу здатною до здійснення класифікації чи розпізнавання.

Архітектура CNN базується на ідеї багатошарових перцептронів. Вона включає шари входу, приховані шари та шар виходу.

Згорткові шари виконують операцію згортки, що дозволяє моделі виявляти локальні особливості в зображеннях. Ці особливості можуть бути краї, форми або

текстури. Перехідні шари здійснюють підсилення ознак та зменшення розмірності зображення. Повнозв'язані шари виконують класифікацію або регресію.

Головною перевагою архітектури CNN є її здатність автоматично вивчати репрезентативні характеристики зображень без потреби в ручній розробці фільтрів. Це зроблено шляхом тренування моделі на великій кількості прикладів. В результаті, згорткові нейронні мережі можуть досягати вражаючих результатів у завданнях класифікації та розпізнавання зображень[59].

Точність і втрати є двома основними метриками машинного навчання. Точність вимірює ефективність класифікаційної моделі і виражається у відсотках. Вона показує, яка частка правильно передбачених значень моделі відносно загальної кількості передбачених значень. Точність є важливою метрикою, оскільки дозволяє оцінити, наскільки добре модель класифікує дані.

Функції втрат вимірюють втрату або помилку моделі. У задачах класифікації, втрата і перехресна ентропія є поширеними функціями втрат. Вони використовуються для оцінки відповідності між прогнозованою імовірністю класу та фактичним класом. Мета полягає в мінімізації цієї втрати під час навчання моделі.

У задачах регресії використовуються інші функції втрат, такі як середньоквадратична помилка і ймовірнісна втрата. Середньоквадратична помилка вимірює квадратичне відхилення прогнозу моделі від фактичного значення, а ймовірнісна втрата використовується, коли прогноз залежить від ймовірності.

Використання правильної функції втрат є важливим кроком при навчанні моделі. Вона допомагає оцінити ефективність моделі і пошукати оптимальні параметри, щоб мінімізувати втрату [60].

CNN часто використовують для задачі класифікації медичних зображень, що включає в себе розпізнавання та розподіл зображень. Метою є використання комп'ютерних алгоритмів для автоматизації ідентифікації та класифікації медичних

зображень на основі їх змісту, що сприяє діагностиці [61]. У зв'язку з цим було розроблено та навчено окремі архітектури, такі як AlexNet, VGG, ResNet, EfficientNet та інші, що орієнтовані на класифікацію через об'єднання повнозв'язних шарів з різними класифікаторами.

Наприклад, у [62] використання CNN допомогло отримати точність в 88-89% для класифікації стадій діабетичної ретинопатії. Або [63] за допомогою архітектури VGG отримали точність 99,17% для класифікації захворювань ока з використанням ОКТ зображень. Схожа до задачі диплому задача вирішувалась за допомогою CNN [64] і як результат, отримано точність в 93%.

Можна розглянути архітектуру найпопулярніших моделей для класифікації, наприклад для ResNet вона зображена на рисунку 4.1.

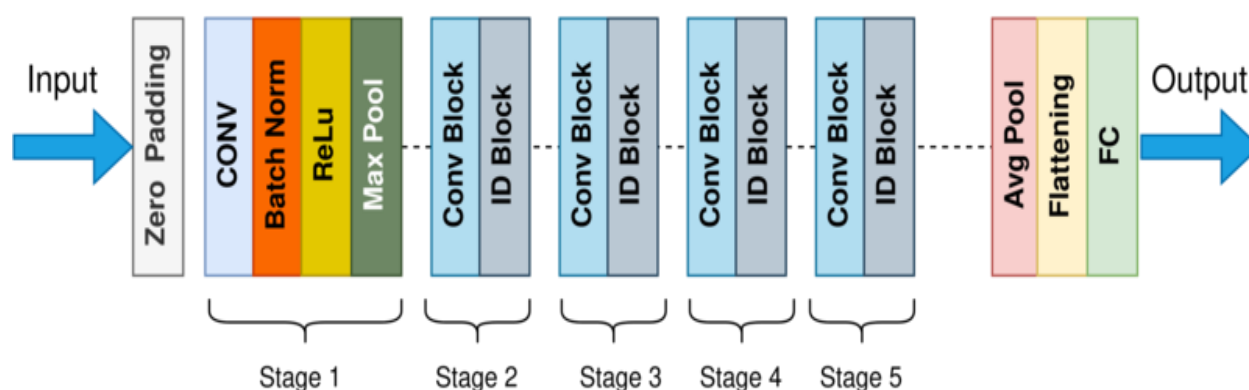


Рисунок 4.1 – Архітектура мережі ResNet50.

Згорткові шари в ResNet50 складаються з кількох згорткових шарів з подальшою пакетною нормалізацією та активацією ReLU. Ці шари відповідають за вилучення функцій із вхідного зображення, таких як краї, текстури та форми. За згортковими шарами слідує шар максимального об'єднання, які зменшують просторові розміри карт об'єктів, зберігаючи найважливіші об'єкти.

Ідентифікаційний блок і згортковий блок є ключовими будівельними блоками ResNet50. Ідентифікаційний блок — це простий блок, який пропускає вхідні дані через серію згорткових шарів і додає вхідні дані назад до вихідних даних. Це дозволяє мережі вивчати залишкові функції, які відображають вхідні дані та бажані вихідні дані. Згортковий блок подібний до ідентифікаційного блоку, але з додаванням згорткового шару 1×1 , який використовується для зменшення кількості фільтрів перед згортковим шаром 3×3 .

Останньою частиною ResNet50 є повністю з'єднані шари. Ці шари відповідають за остаточну класифікацію. Вихідні дані останнього повністю зв'язаного рівня подаються у функцію активації softmax для отримання кінцевих ймовірностей класу [65].

Архітектура VGG16 зображена на рисунку 4.2.

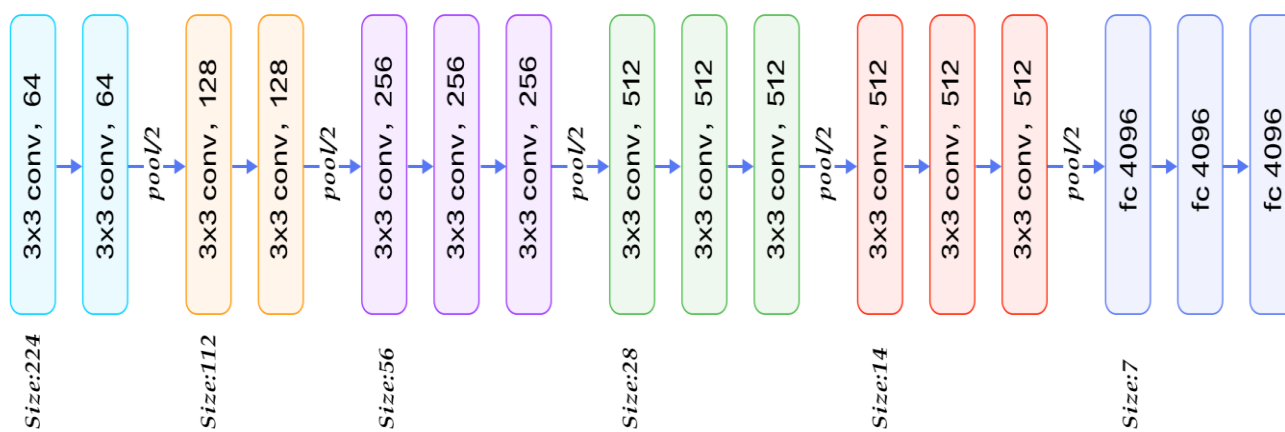


Рисунок 4.2 – Архітектура ЗНМ VGG16

Число 16 у назві VGG вказує на те, що це 16-рівнева глибока нейронна мережа (VGGnet). Це означає, що VGG16 є досить розгалуженою мережею та має загалом близько 138 мільйонів параметрів. Навіть за сучасними мірками це величезна мережа.

Однак простота архітектури VGGNet16 робить мережу більш привабливою. Лише дивлячись на його архітектуру, можна сказати, що він досить однорідний. Є кілька шарів згортки, за якими слідує шар об'єднання, який зменшує висоту та ширину. Якщо ми подивимося на кількість фільтрів, які ми можемо використовувати, доступні близько 64 фільтрів, які можна подвоїти приблизно до 128, а потім до 256 фільтрів. В останніх шарах можна використовувати 512 фільтрів [66].

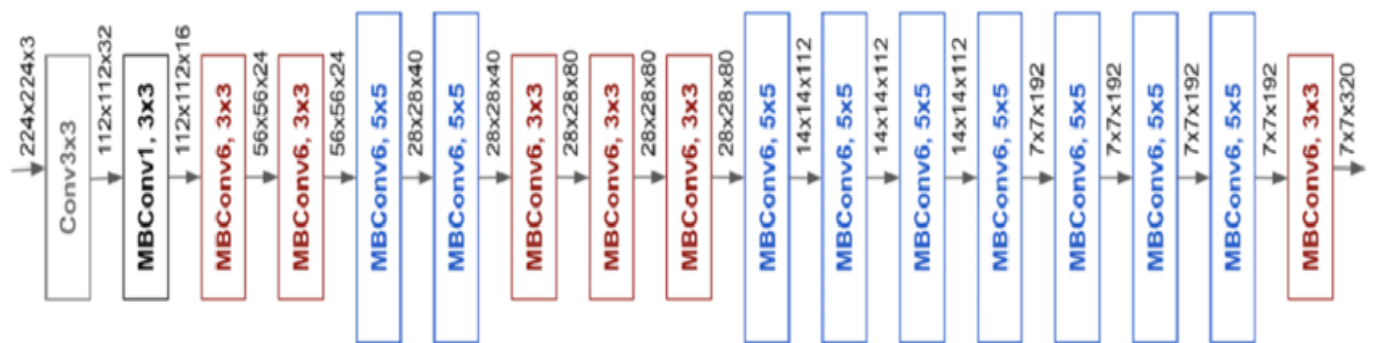


Рисунок 4.3 – Архітектура ЗНМ EfficientNetB0

ЗНМ EfficientNetB0 приймає на вхід зображення розміром 224x224x3, після чого використовує декілька згорткових шарів з різними розмірами ядра. Модель має всього 5.3 мільйони параметрів (згідно рисунку 5.6), що порівняно з іншими моделями не так багато. Через невелику кількість параметрів модель повинна найшвидше серед представлених обробляти дані.

4.5 Висновки до розділу 4

Нейронні мережі активно поширюють свій вплив на наше життя. Ми зустрічаємось з ними в повсякденному житті, вони спрощують рутинну роботу та підвищують нашу продуктивність. Це потужний інструмент, який має неймовірні перспективи для розвитку та надзвичайно широкий спектр застосування. Зокрема й в медицині для задач класифікації медичних зображень. Уже сьогодні впроваджують системи, що здатні класифікувати зображення не гірше за спеціалістів з точністю близькою до 100%. Саме такий підхід дозволить спростити діагностування ранніх стадій діабетичної ретинопатії у хворих, де немає достатньо кваліфікованих кадрів.

5. ЗАДАЧА РОЗПІЗНАВАННЯ ДІАБЕТИЧНОЇ РЕТИНОПАТІЇ ЗА АНАЛІЗОМ ЗОБРАЖЕНЬ СІТКІВКИ ОКА

Як відомо, діабетична ретинопатія доволі небезпечна хвороба, а для її виявлення потрібні не абиякі ресурси. На сьогоднішній день виявлення ДР є ручним процесом, що вимагає наявності у лікарні кваліфікованого спеціаліста. Проте часто складається так, що результати обстежень затримуються через людський фактор або класифікуються неправильно. Прогресування до погіршення зору можна запобігти, якщо вчасно діагностувати хворобу і розпочати лікування чи профілактику, однак хвороба на початкових стадіях проявляється незначними симптомами, поки не стає занадто пізно для ефективного лікування. Необхідного досвіду чи обладнання зазвичай не вистачає в регіонах. Кількість людей з діабетом продовжує зростати, а інфраструктури для коректної діагностики більше не стає, це призводить до збільшення часу діагностики хвороби та зменшення точності. Це сформувало потребу в автоматизації процесу діагностики ДР. З появою машинного навчання у сфері автоматизації було досягнуто значного прогресу.

5.1 Огляд бази даних

Для виконання практичної частини дипломної роботи було підібрано базу даних зображень сітківки ока. З ресурсу Kaggle було використано публічно доступну базу даних зображень очного дна в нормі та за наявності діабетичної ретинопатії різних стадій Diabetic Retinopathy 2015 Data Colored Resized [9], яка побудована на базі Diabetic Retinopathy Detection [67]. У цій базі даних міститься більше 35-ти тисяч зображень сітківки ока, які можна використовувати для визначення наявності

діабетичної ретинопатії. В оригінальній базі даних [67] зображення великого розміру, що призводить до загального розміру бази даних у майже 90 Гб. В той час як обрана оброблена база даних [9] має розмір трохи більше 2 Гб.

Розподіл даних у цій базі даних наступний:

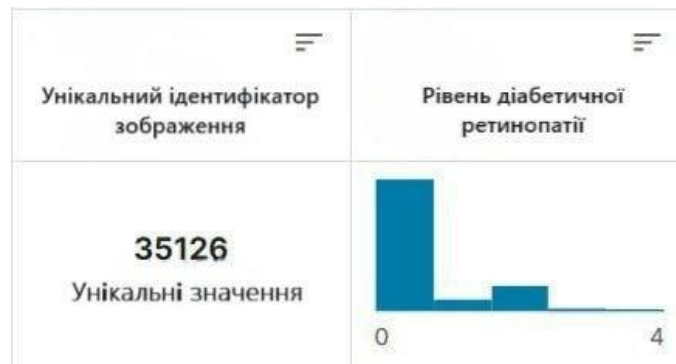


Рисунок 5.1 – Загальний розподіл вхідних даних у базі даних.

Згідно інформації на сайті з базою даних усі дані поділені на 5 категорій, згідно опису в розділі 1. За кількістю даних розподіл на рисунку 5.1 відповідає наступним кількостям:

- Відсутність діабетичної ретинопатії (здорова сітківка) – 25810 зображень;
- Непроліферативна ретинопатія – 2443 зображень;
- Препроліферативна ретинопатія – 5292 зображень;
- Проліферативна ретинопатія – 873 зображення;
- Діабетична макулопатія – 708 зображень.

Загалом 35126 зображень. Проте така велика кількість вхідних даних вимагає значних ресурсів для навчання моделей машинного навчання, тому було прийнято рішення вибрати тільки частину з цих даних, результати зображені на рисунках 5.2 та 5.3. А також зменшити кількість класів з 5 до 2 і вирішувати задачу бінарної класифікації в межах класів «Здоровий» та «Хворий».

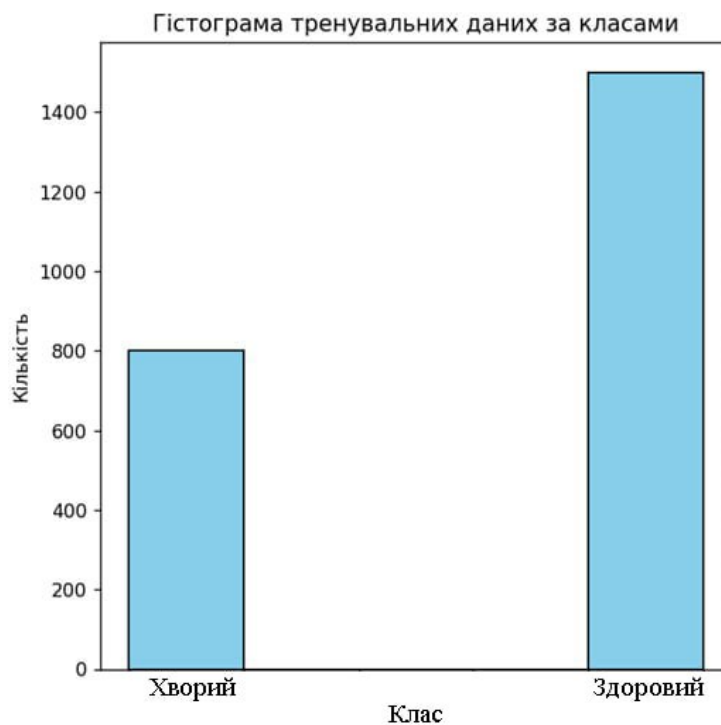


Рисунок 5.2 – Розподіл тренувальних даних на 2 класи

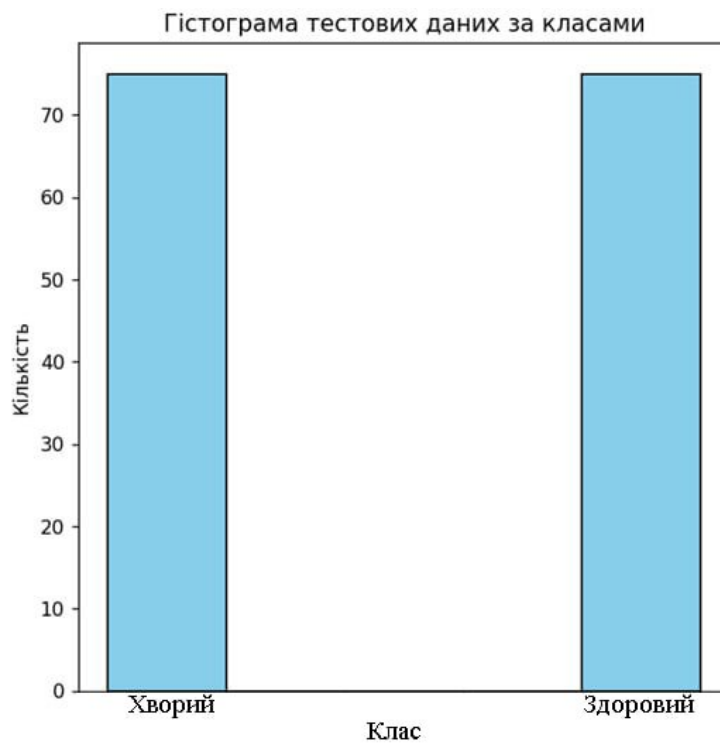


Рисунок 5.3 – Розподіл тестових даних на 2 класи

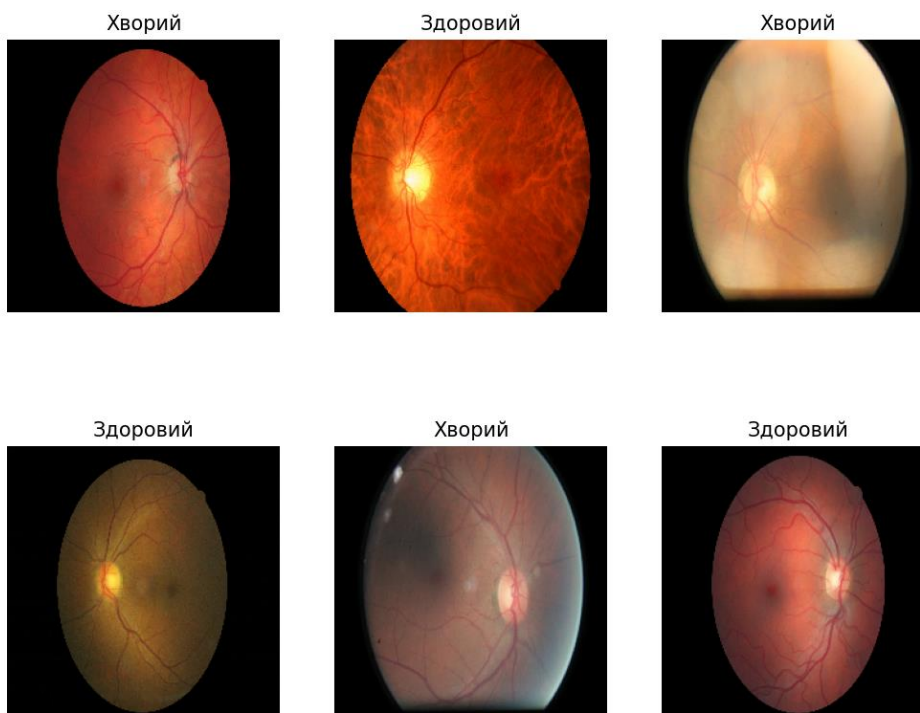


Рисунок 5.4 – Приклад випадкових зображень з тренувальних даних.

На рисунку 5.4 можна виділити деякі особливості обраної бази даних. Тут зібрані дані, що зроблені на різних фундус-камерах, тому видно різні види яскравості зображень, різну форму та відтінки кольору. А також, наприклад, зображення в правому верхньому куті з міткою класу «Хворий» поганої якості, на ньому майже не видно кровоносних судин і попри наявну мітку класу важко сказати до якого з класів належить це зображення. В цілому обрана база даних доволі посередньої якості. Багато зображень не сфокусовані, деякі просто повністю чорні, іноді трапляються зображення, на яких зовсім не видно судин ока. Якщо розглядати цю базу даних з боку точності передбачування діагнозу, то не варто очікувати надвисокої точності. В реальному житті рідко коли виходить отримати достатньо чітке зображення, а тому можна вважати, що обираючи цю базу даних ми обираємо дані, які найближчі до реальності. При обробці літератури було визначено декілька джерел, що демонстрували велику точність своїх алгоритмів, проте у тих роботах, крім іншого,

зазначалось, що кількість вхідних даних невелика і збільшена за допомогою аугментації, що цілком ймовірно і призвело до високої точності.

5.2 Розробка власної моделі нейронної мережі

Для реалізації нейронної мережі було обрано бібліотеку Keras з відкритим вихідним кодом на мові програмування Python [68]. Ця бібліотека спроектована для швидких експериментів з глибокими нейронними мережами. Вона зручна у використанні та модульна.

У 4-му розділі дипломної роботи описано загальні шари з яких зазвичай складаються моделі. На основі цієї інформації було створено першу версію власної нейронної мережі, що складається:

- 1 Input шару, що використовується для входу моделі і є точкою входу.
- 6 шарів Conv2D з функцією активації ReLu, що використовуються для операції згортки для виявлення різних характеристик, а функція активації додає нелінійність в модель і покращує продуктивність;
- 3 шарів MaxPool2D, що зменшують розмірність карт ознак, обираючи найбільш важливі для кожної області;
- 1 шару Flatten для конвертації вхідних даних у меншу розмірність;
- 1 шару Dense із 2 нейронами на виході та функцією активації softmax. У даному випадку 2 нейрони означають 2 вихідних класи здоровий/хворий, а функція активації softmax відповідає за розподіл ймовірностей набору чисел, що належать до певного класу.

Окрім параметрів шару ще існують параметри моделі, серед яких:

- Epoch – вся база даних пройшла через нейронну мережу. У випадку для власної моделі використовуємо 15 epoch.

- Batch_size – число об'єктів у одному тренувальному батчі. Батч – частина з датасету певного розміру, у дослідженні використовуються батчі з 32 зображень.

Щоб з'єднати все це в єдину модель потрібно використати Sequential. Для навчання моделі використовують метод fit. Щоб навчити модель потрібно задати функцію втрат, оптимізатор та вибірково метрики для моніторингу. Зазвичай для цього використовують метод compile. У випадку створеної моделі було передано параметри:

- optimizer=Adam(learning_rate=0.0001) – оптимізатор із значенням зсуву 0,0001;
- loss='binary_crossentropy' – вид втрат, бінарний, оскільки класифікуємо на 2 класи;
- metrics=['accuracy'] – для обрахунку точності використовуємо метрику точності.

Після всього описаного навчаємо модель та будуємо графіки точності та втрат на кожній епосі для тренувальних та тестових даних. Результати наведено на рисунку 5.5.

Для тренувальних даних точність росте до 70%, а втрати зменшуються. Проте з тестовими даними ситуація не найкраща. На даному етапі дослідження модель для тестових даних зовсім не стабільна. Точність не зростає, втрати не падають. Єдиний пік точності в 60% на 11 епосі, але модель все ще залишається не придатною для використання. Все ж, доволі складним є завдання правильного підбору шарів та їх параметрів для власної моделі, тому й результати сумнівні.

Існують попередньо моделі з виваженою кількістю шарів та їх параметрами. Деякі з них перераховані на рисунку 5.6.

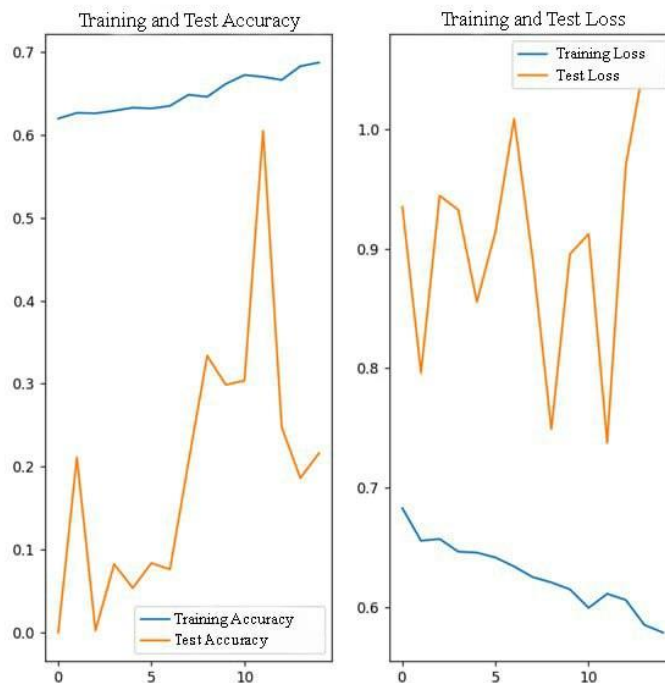


Рисунок 5.5 – Показники точності та втрат для навчальної та тестової вибірок даних для власної моделі з даними без попередньої обробки.

Модель	Розмір (Мб)	Параметри	Час (мс) на крок виведення (CPU)	Час (мс) на крок виведення (GPU)
Xception	88	22.9M	109.4	8.1
VGG16	528	138.4M	69.5	4.2
VGG19	549	143.7M	84.8	4.4
ResNet50	98	25.6M	58.2	4.6
ResNet50V2	98	25.6M	45.6	4.4
ResNet101	171	44.7M	89.6	5.2
ResNet101V2	171	44.7M	72.7	5.4
ResNet152	232	60.4M	127.4	6.5
ResNet152V2	232	60.4M	107.5	6.6
InceptionV3	92	23.9M	42.2	6.9
InceptionResNetV2	215	55.9M	130.2	10.0
MobileNet	16	4.3M	22.6	3.4
MobileNetV2	14	3.5M	25.9	3.8
DenseNet121	33	8.1M	77.1	5.4
DenseNet169	57	14.3M	96.4	6.3
DenseNet201	80	20.2M	127.2	6.7
NASNetMobile	23	5.3M	27.0	6.7
NASNetLarge	343	88.9M	344.5	20.0
EfficientNetB0	29	5.3M	46.0	4.9

Рисунок 5.6 – Список популярних попередньо тренованих моделей та деякі їх параметри.

5.3 Розробка моделі з використанням попередньо натренованої моделі

Зважаючи на низький результат роботи моделі без попереднього тренування, на наступному кроці дослідження буде застосовано ряд попередньо натренованих моделей, наприклад ResNet, Inception, EfficientNet. Оскільки ресурси для тренування обмежені, то обирати модель доцільно з високою швидкістю виконання на процесорі у поєднанні з простотою у використанні для розробника. Для початку використаємо декілька моделей для розуміння, яка з них показує кращу точність за обраних параметрів. Для використання попередньо натренованої моделі потрібно у Sequential вставити імпортовану модель із вхідними параметрами. Для усіх моделей використовуються однакові параметри:

- `input_shape=(224, 224, 3)` – розмір вхідного зображення, передаємо 224 і 224 як ширину та висоту зображення та 3 як кількість каналів для RGB зображення;
- `weights='imagenet'` – включення попередньо підготовлених ваг, що відповідають набору даних ImageNet та задачі класифікації;
- `include_top=False` – відноситься до повнозв'язних шарів після згорткової основи, оскільки оригінальне завдання відрізняється від нашого і нам не потрібні ці шари.

Зазвичай для попередньо навчених моделей рекомендують дописати ще 2 шари [69] для ЗНМ. Після цього випробуємо отриману модель на вхідних даних і отримані результати на рисунках 5.7–5.8.

З рисунків 5.7 і 5.8 росте точність тренувальних та тестових даних тільки для попередньо тренованої моделі EfficientNetB0. Цю модель візьмемо за основу для наступних варіацій. Для початку, до основної моделі було додано шар Flatten та вихідний шар Dense з функцією активації sigmoid та з двома нейронами на виході, що забезпечують бінарну класифікацію. Вхідні дані без обробки завантажуюмо функцією ImageDataGenerator, що дозволить одразу розбити вхідні дані на групи (батчі).

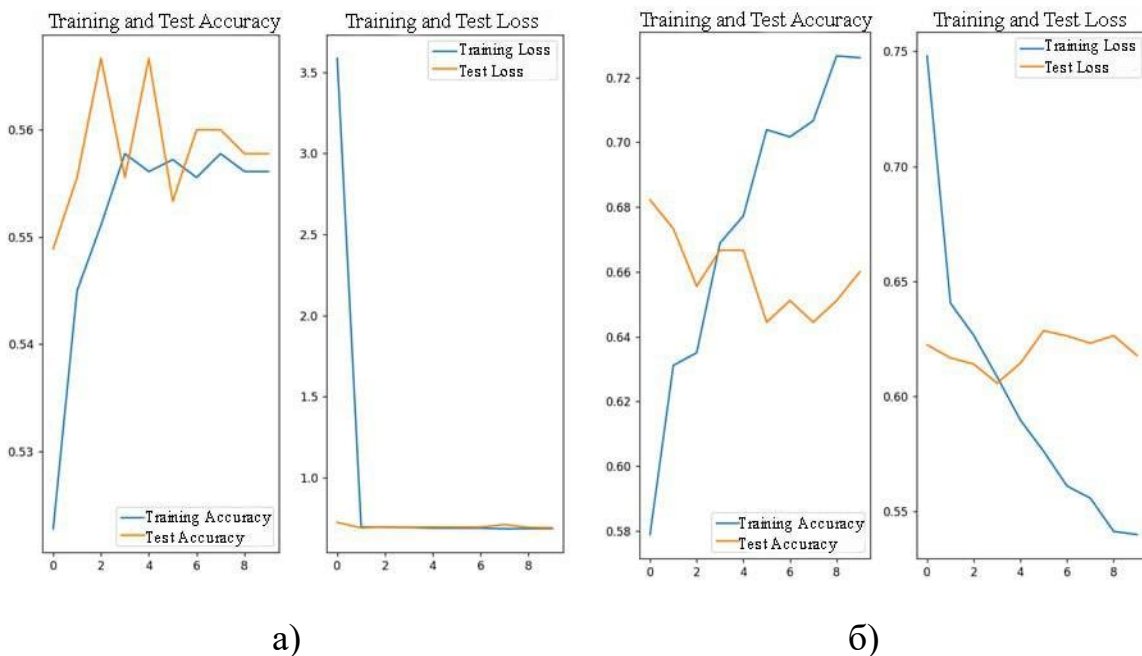


Рисунок 5.7 – Показники точності та втрат для навчальної та тестової вибірок даних для моделей а) InceptionV3 та б) ResNet50

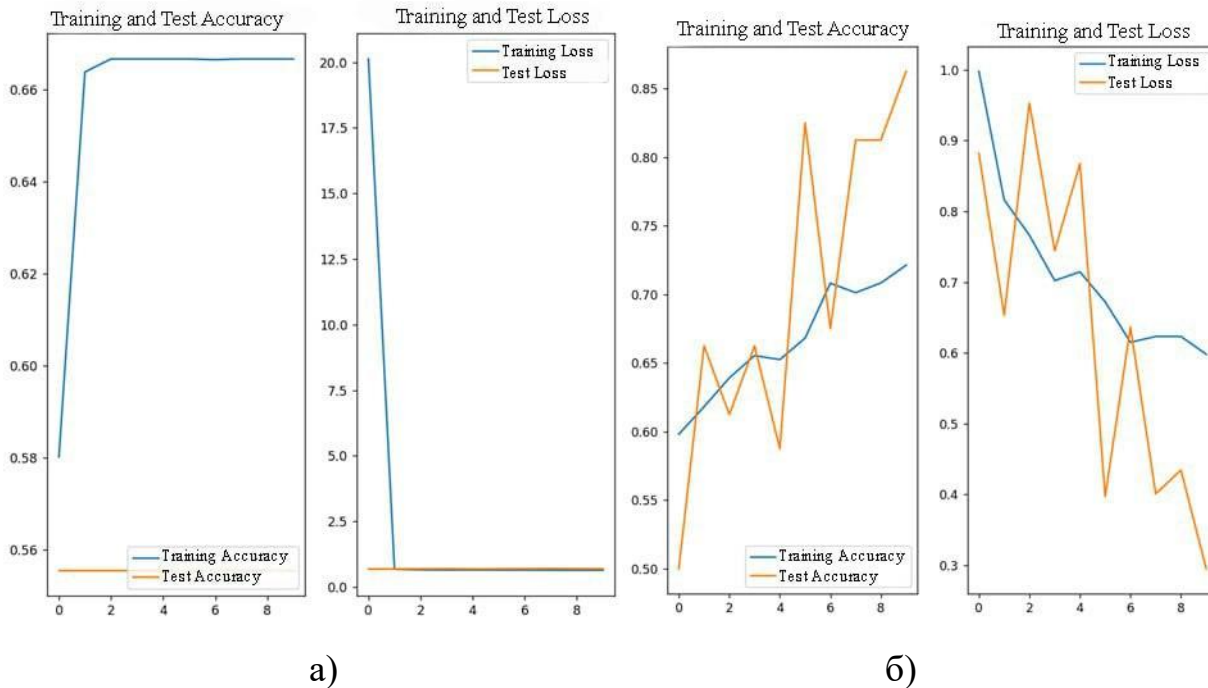


Рисунок 5.8 - Показники точності та втрат для навчальної та тестової вибірок даних для моделей а) InceptionResNetV2 та б) EfficientNetB0.

Сформувавши кінцевий варіант моделі, навчаємо її з параметрами, що наведені вище і результати навчання представлені на рисунку 5.9.

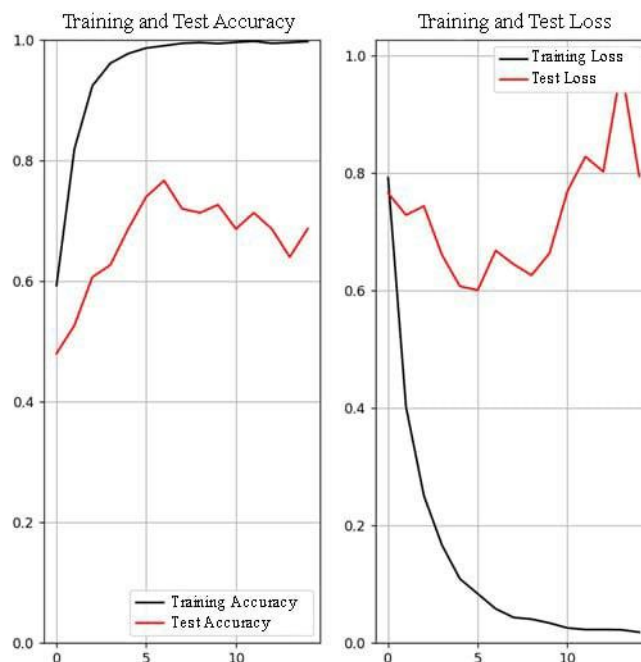


Рисунок 5.9 – Показники точності та втрат для навчальної та тестової вибірок для моделі EfficientNetB0.

З рисунку 5.9 помітно, що точність для тренувальних даних швидко росте і досягає майже 100% в той час як для тестових даних точність росла тільки 6-ї епохи, а далі почала спадати. Така поведінка властива для перенавчених моделей. В цілому, бачимо результативність моделі на рівні 70-75%. Проте перенавчання не є хорошим результатом, а щоб його уникнути використаємо вбудовані методи аугментації зображень з функції ImageDataGenerator. Аугментація або розширення - це набір методів, що покращують розмір та якість навчальних наборів даних для кращого навчання моделей глибокого навчання. До цих алгоритмів відносять геометричні перетворення, змішування зображень, випадкове стирання, розширення простору ознак та інші [70]. Для власної моделі використаємо геометричні перетворення, серед яких `width_shift_range` та `height_shift_range`, які відповідають за зсув зображення по

горизонталі та вертикалі на випадкове значення із заданого діапазону, `zoom_range` для випадкового наближення та `rotation_range` для випадкового повороту в межах заданого кута. Ці кроки дозволять зробити зображення більш випадковим та знизити швидкість перенавчання моделі. За таких змін зображення сітківки матимуть вигляд як на рисунку 5.10.

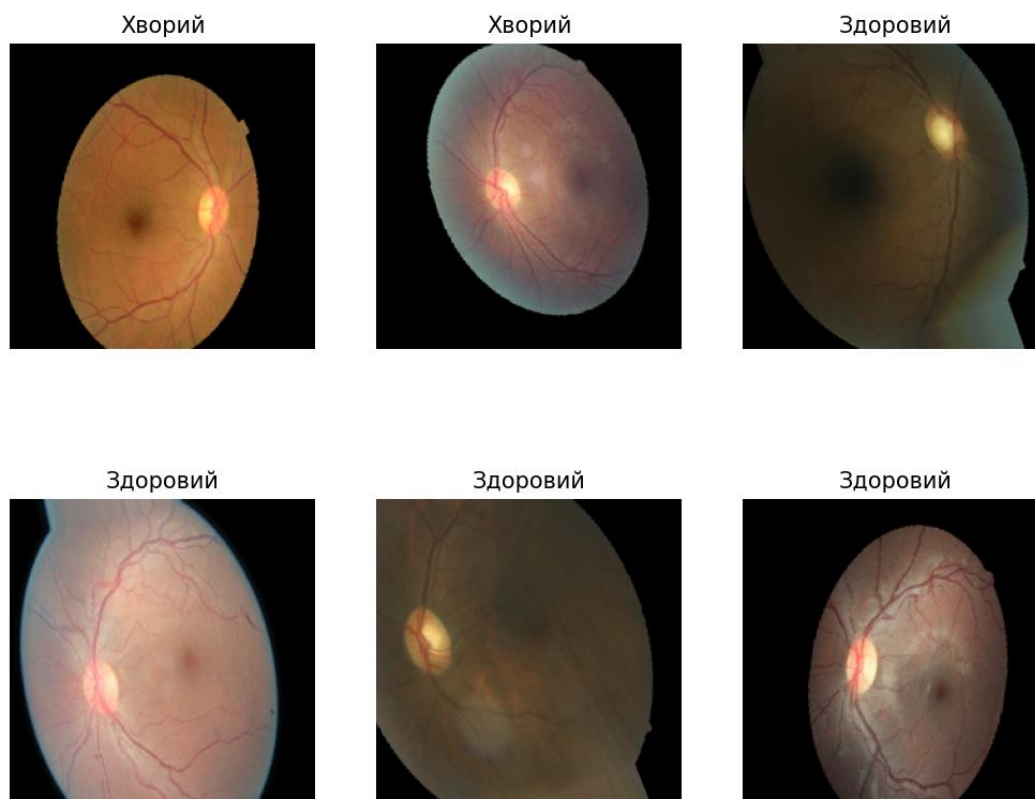


Рисунок 5.10 - Приклад випадкових зображень з тренувальних даних з використанням методів аугментації.

Відносно центрованих зображень з рисунку 5.4 помітно суттєву різницю. Така обробка зображень повинна вирішити проблему перенавчання моделі. Навчаємо модель на оновлених даних. Результат навчання відображено на рисунку 5.11.

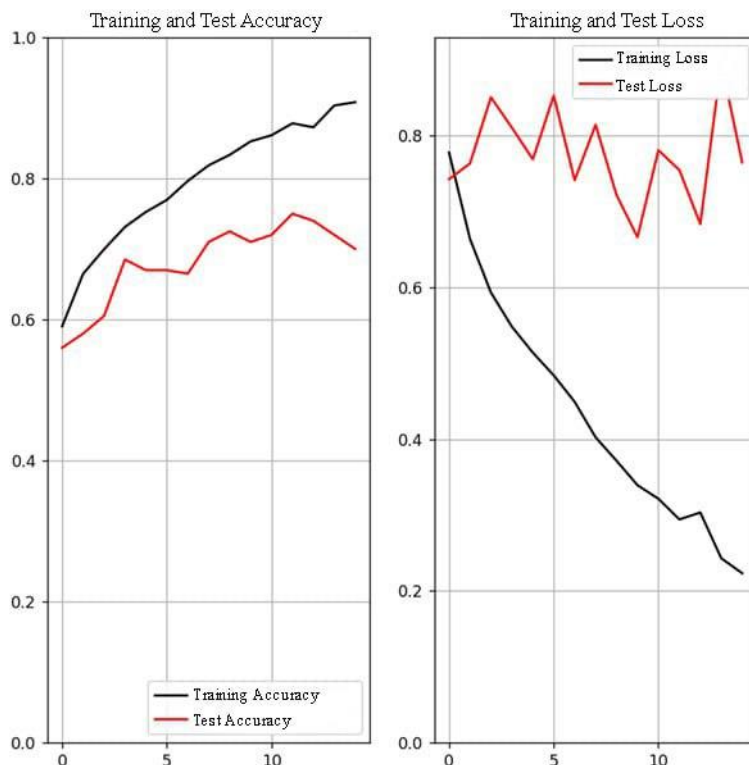


Рисунок 5.11 - Показники точності та втрат для навчальної та тестової вибірок з використанням методів аугментації.

Точність моделі залишилась на рівні 70-75%, а проблему перенавчання вдалось вирішити. Проте залишається проблема посередньої якості зображень. На рисунку 5.10, як і на рисунку 5.4 присутні зображення із загальної бази даних, що мають низьку контрастність, що ускладнює виявлення кровоносних судин на зображенні. Одним серед описаних методів покращення контрасту в 3 розділі був метод вирівнювання гістограми яскравості. Для обробки зображень використовуємо модуль cv2. За допомогою функції equalizeHist цього модулю реалізовано покращення гістограми яскравості. Проте функція працює лише для одноканальних (сірошкальних) зображень [71], тому вирівнювати гістограму необхідно по кожному каналу окремо. На рисунку 5.12 показано результат обробки зображення до та після еквалізації, а також гістограми яскравості.

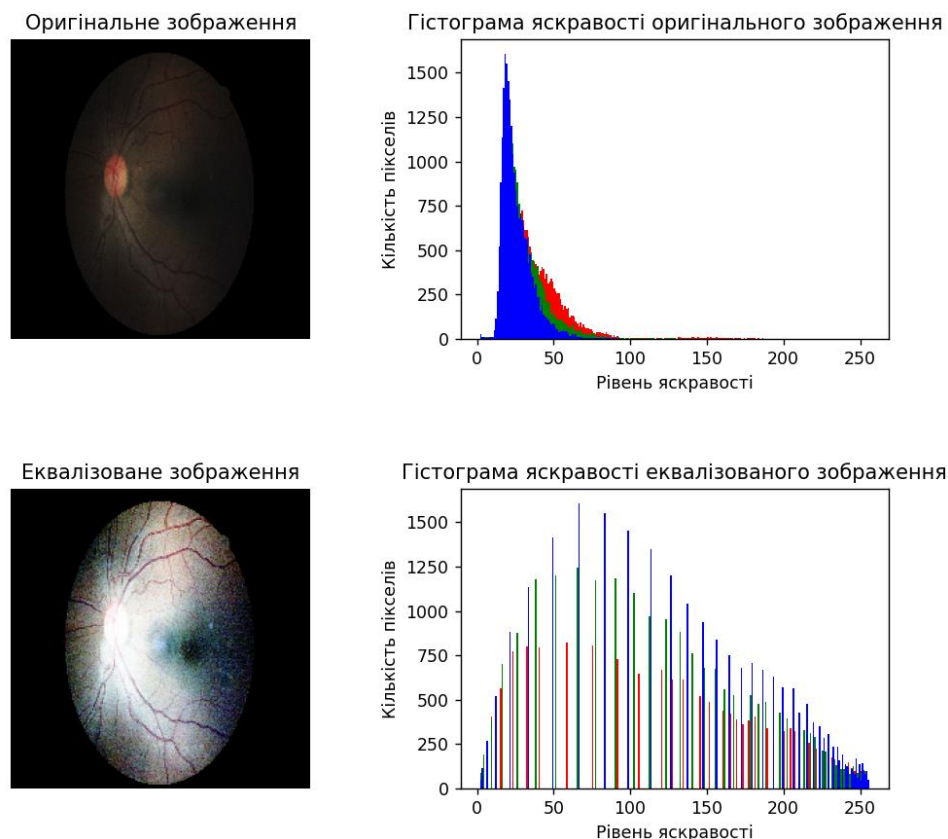


Рисунок 5.12 – Застосування методу еквалізації зображень до зображення з низькою контрастністю.

Після застосування даного методу до тренувальних та тестових зображень результат виведено на рисунку 5.13. Загалом, помітно, що зображення стали більш контрастними і на них значно краще видно судини очного дна, особливо в темних ділянках зображення. Тепер навчаємо модель на нових даних. Таке ж вирівнювання гістограми застосовується і для тестової вибірки. Результати навчання зображені на рисунку 5.14.

Після такої попередньої обробки зображень точність моделі наблизилась до рівня 80-83%, що є доволі хорошим результатом, зважаючи на якість вхідних даних.

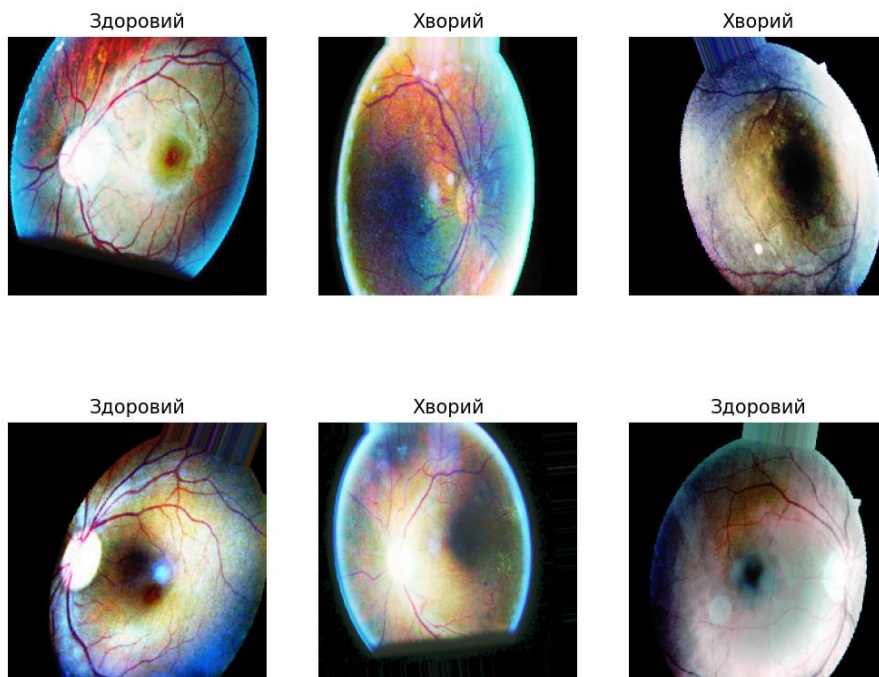


Рисунок 5.13 Приклад випадкових зображень з тренувальних даних з додаванням методів аугментації та після вирівнювання гистограми.

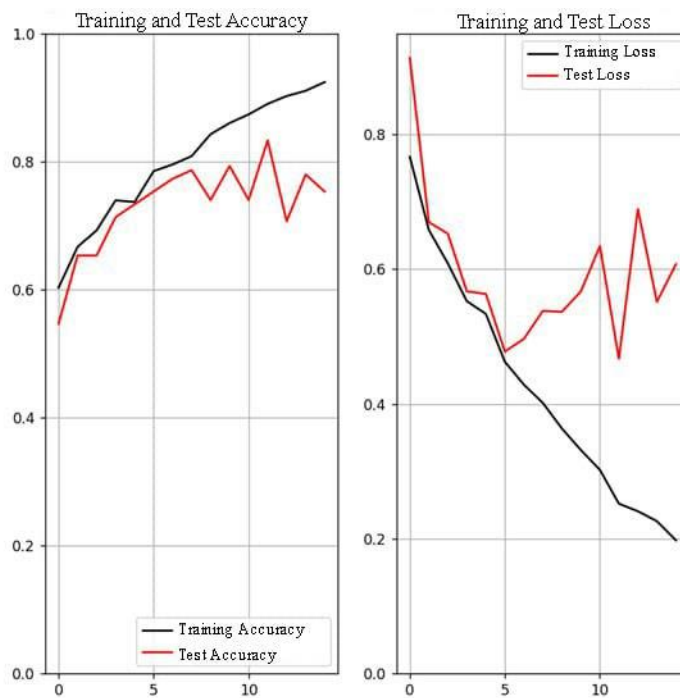


Рисунок 5.14 - Показники точності та втрат для навчальної та тестової вибірок з додаванням методів аугментації та вирівнювання гистограми.

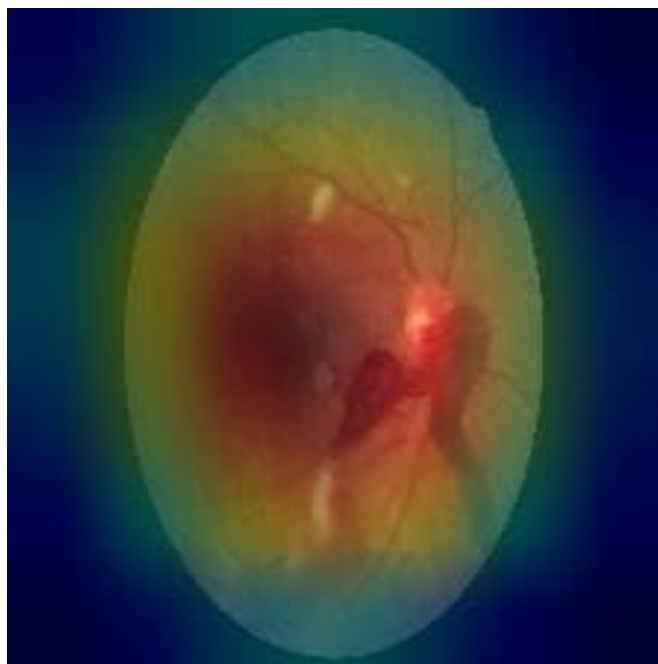


Рисунок 2.15 – Результат накладання теплової карти останнього згорткового рівня на початкове зображення.

Оскільки метод вирівнювання гистограми допоміг покращити точність, то при використанні інших методів обробки зображень будуть використовуватись зображення власне після еквалізації. Зокрема, у третьому розділі було розглянуто сферу детектування країв і описано методи Кенні, Собеля, Лапласа та Превітта. Далі у роботі буде застосовано кожен з методів до тренувальних та тестових зображень та перевірено точність моделі для виявлення найбільш точного методу для розв'язку поставленої задачі.

Почнемо з оператора Кенні. Застосувавши його до тренувальних і тестових вибірок були отримані наступні результати на рисунку 5.16.

Через різноманітність зображень складно підібрати пороги для гістерезису при пошуку країв, оскільки для деяких зображень пороги можуть бути низькими, а для інших вже зависокими. Власне через це результати для моделі з такими даними можуть бути невтішні. Результати зображені на рисунку 5.17.

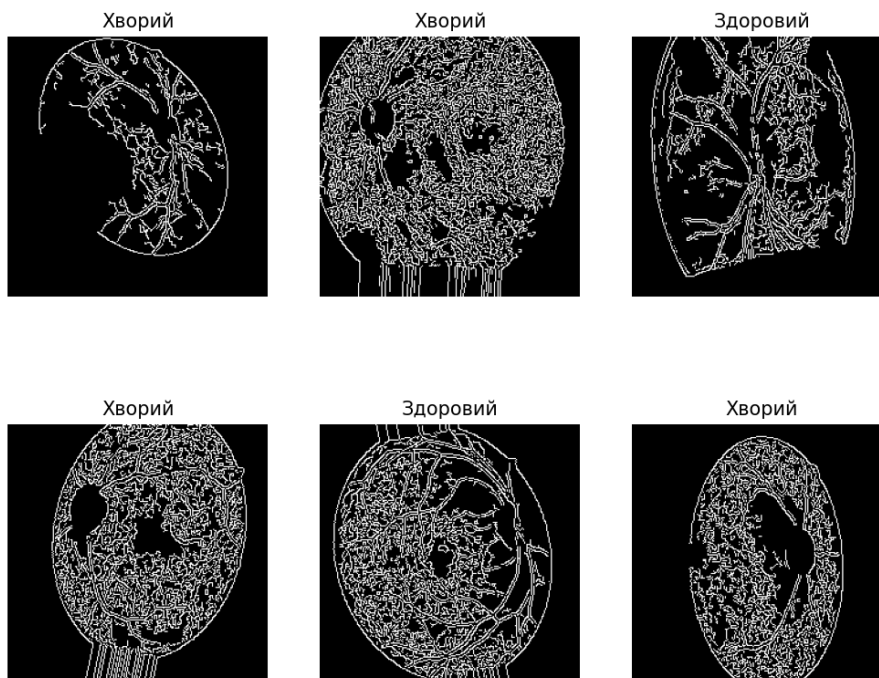


Рисунок 5.16 - Приклад випадкових зображень з тренувальних даних з додаванням методів аугментації, після вирівнювання гістограми та з додаванням оператора Кенні.

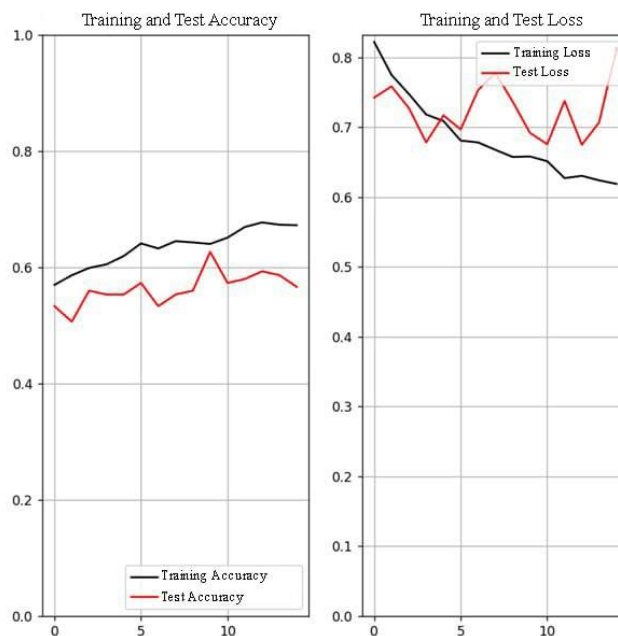


Рисунок 5.17 - Показники точності та втрат для навчальної та тестової вибірок з додаванням методів аугментації, вирівнювання гістограми та додаванню оператора Кенні.

Рівень точності такої моделі близький до 60%, що порівняно з даними без обробки значно гірше. Проте це не означає, що в цілому такий підхід не варто використовувати, це лише означає, що для заданої бази даних не вдалось підібрати параметри для методу Кенні так, щоб точність моделі була вищою.

Наступним кроком буде використано оператор Собеля з такою ж логікою як і оператор Кенні.

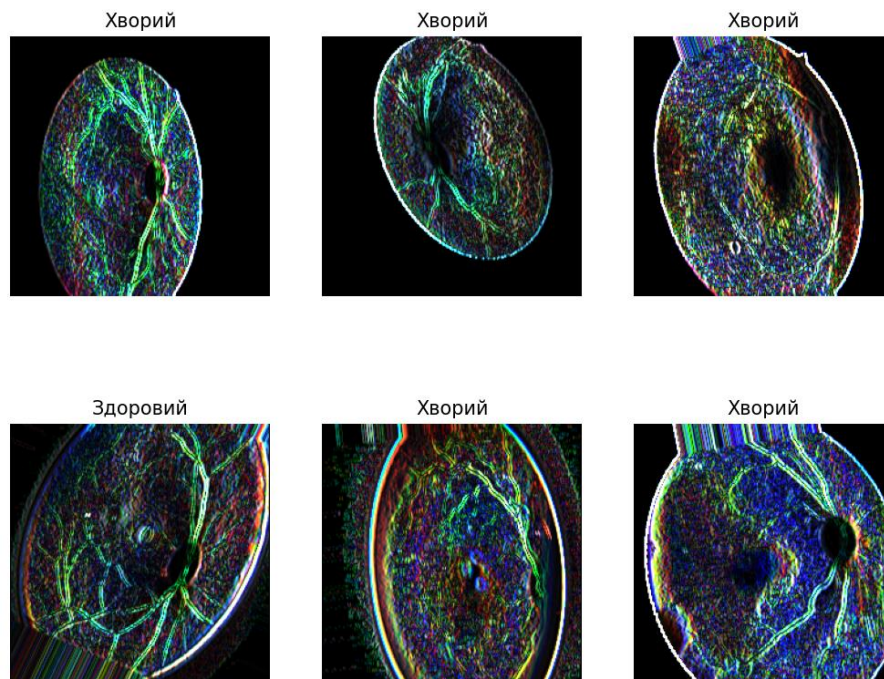


Рисунок 5.18 – Приклад випадкових зображень з тренувальних даних з додаванням методів аугментації, після вирівнювання гістограми та оператора Собеля.

В цілому, зображення виглядають краще, ніж після оператора Кенні, хоч на них багато дефектів. Результат навчання моделі на рисунку 5.19. Порівнюючи з початковим результатом маємо приріст на 1-2%, тобто загальну точність на рівні 75-76%. В цілому робота такої моделі задовільна, проте не найкраща серед представлених. Крім того, швидкість навчання на тренувальних даних дещо нижча за швидкість навчання в моделі з даними після еквалізації гістограми чи навіть моделі з початковими даними.

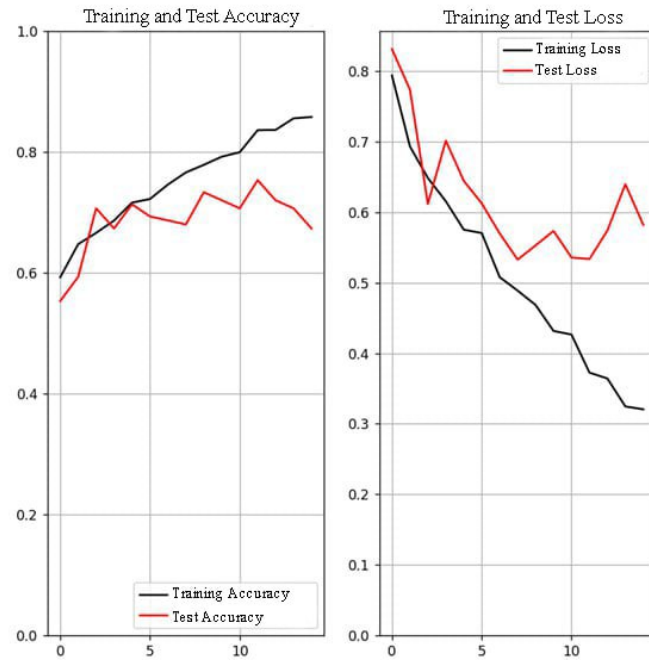


Рисунок 5.19 – Показники точності та втрат для навчальної та тестової вибірок з додаванням методів аугментації, вирівнювання гістограми та оператора Собеля.

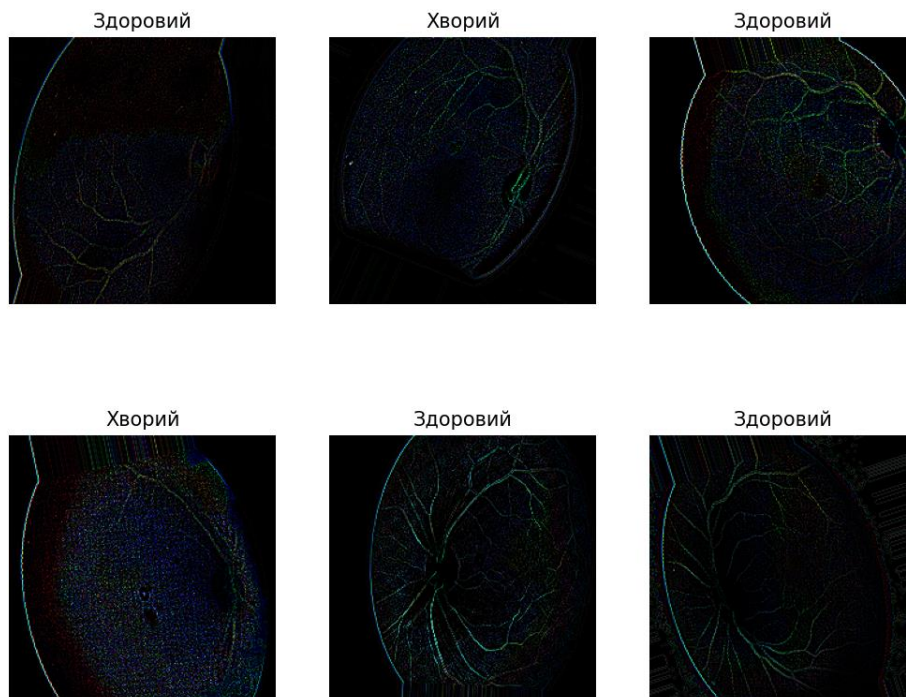


Рисунок 5.20 - Приклад випадкових зображень з тренувальних даних з додаванням методів аугментації, після вирівнювання гістограми та використання оператора Лапласа.

З оператором Лапласа робимо все те ж саме. Обробляємо зображення та навчаємо на них модель, аналізуємо результати. Оброблені зображення тренувальних даних наведені на рисунку 5.20. А результати навчання представлені на рисунку 5.21.

Оператор Лапласа доволі добре виділив судини на зображеннях, проте на них з'явилося дуже багато шумів і в зображення доволі темні.

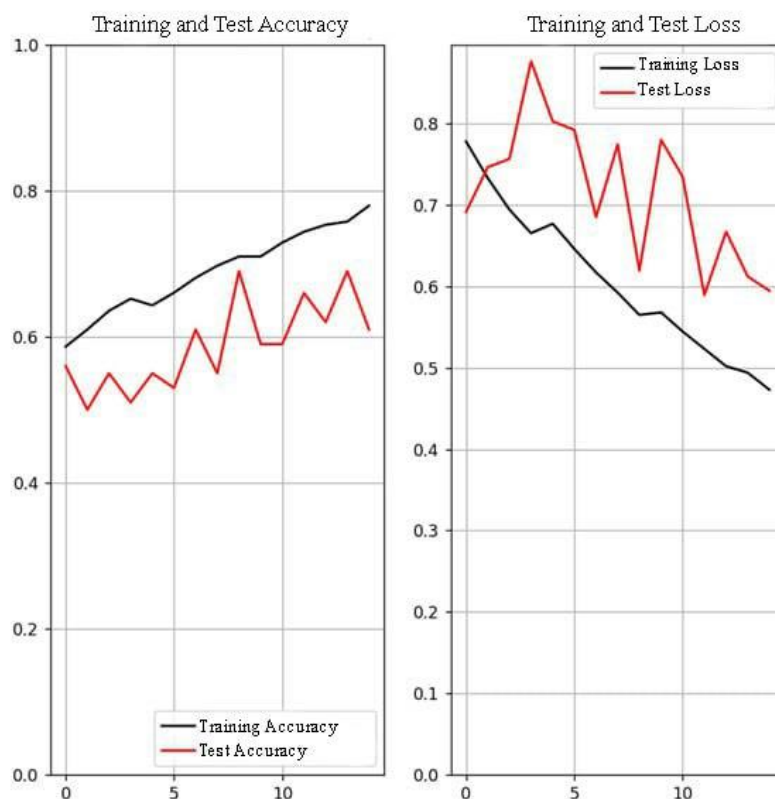


Рисунок 5.21 – Показники точності та втрат для навчальної та тестової вибірок з додаванням методів аугментації, вирівнювання гистограми та оператора Лапласа.

Аналізуючи результати, помітно, що точність значно нижче, ніж за попередніх варіантів обробки зображень. В цілому зберігається точність на рівні 65-68%, що є найгіршим варіантом серед оглянутих раніше, навіть без обробки зображень точність була вище.

Оператор Превітта буде останнім серед операторів, що буде застосовано до зображень. Загальна процедура незмінна – застосовуємо оператор до тренувальних та тестових зображень, не змінюємо конфігурацію моделі та аналізуємо результати. Приклад оброблених зображень на рисунку 5.22, а результат навчання моделі на рисунку 5.23.

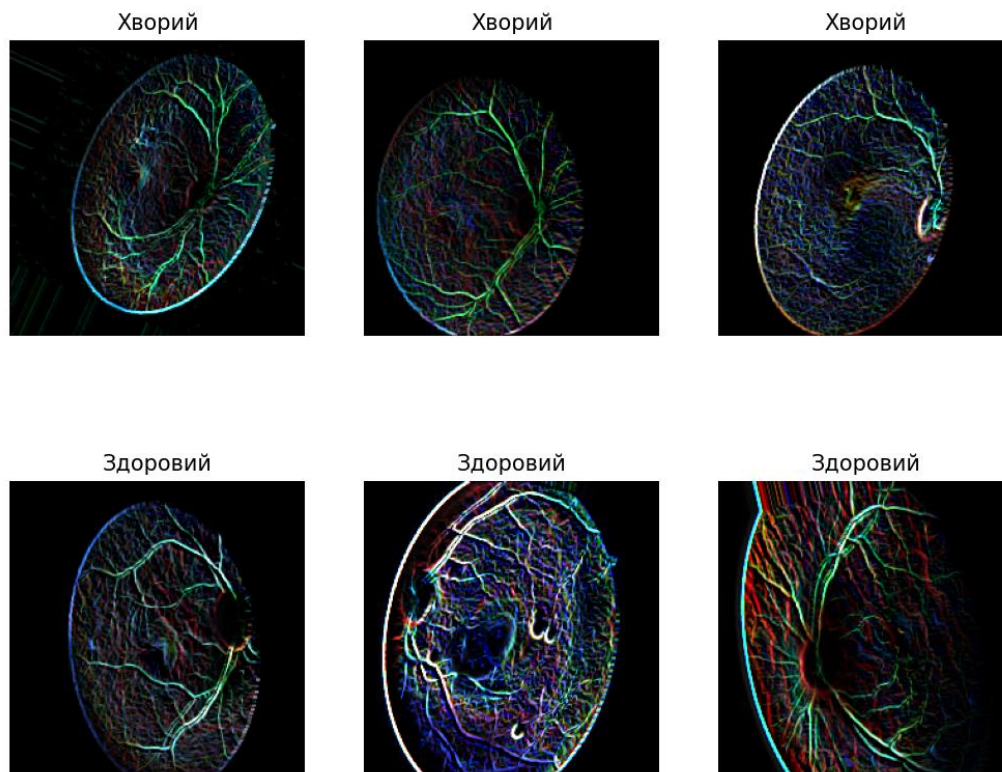


Рисунок 5.22 - Приклад випадкових зображень з тренувальних даних з додаванням методів аугментації, після вирівнювання гістограми та використання оператора Превітта.

В результаті на рисунку 5.22 помітно майже всі судини очного дна, значно менше темних плям, порівняно з оператором Лапласа і більше корисної інформації в порівнянні з оператором Собеля та Кенні. А рисунок 5.23 свідчить про точність на рівні близько 80%, що найбільше серед усіх моделей з використанням операторів для знаходження країв. Кінцеві результати внесені у таблицю в Додатку Б.

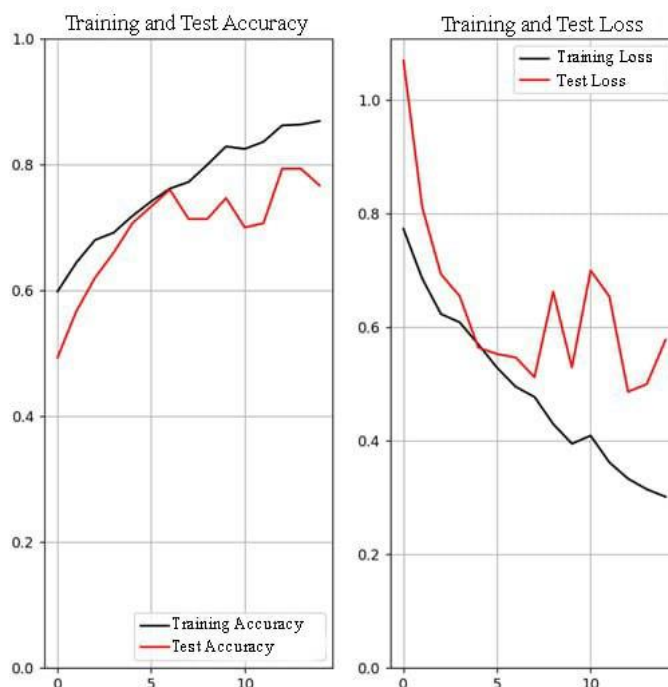


Рисунок 5.23 - Показники точності та втрат для навчальної та тестової вибірок з додаванням методів аугментації, вирівнювання гістограми та оператора Превітта.

5.4 Висновки до розділу

У розділі було проведено роботу по створенню нейронної мережі, що дозволить класифікувати зображення сітківки ока за наявністю чи відсутністю діабетичної ретинопатії. Модель навчалась на зображення з бази даних [9]. При створенні власної моделі не вдалось досягнути хороших результатів, тому було прийнято рішення використати за основу попередньо навчені моделі, з яких модель EfficientNetB0 показала найкращі результати, згідно рисунків 5.7 та 5.8.

При використанні даної моделі виникли небажані результати у вигляді перенавчання моделі, які було вирішено шляхом додавання методів аугментації зображень. За таких умов модель нейронної мережі демонструвала точність

класифікації на рівні 70-75%. Щоб покращити точність моделі було вирішено покращувати якість зображень методами, що описані в третьому розділі роботи.

Використовуючи попередню обробку зображень вдалось отримати точність моделі на рівні до 83%, що суттєво краще за точність без обробки. Така точність була отримана методом вирівнювання гістограми яскравості для зображень навчальної та тестової вибірок.

З інших методів варто виділити оператор Превітта, з використанням якого модель демонструє точність близько 80%.

ВИСНОВКИ

Робота присвячена розпізнаванню діабетичної ретинопатії за аналізом сітківки ока. Основними результатами проведеного дослідження є наступні:

1. Темпи поширення цукрового діабету швидші, ніж темпи розвитку систем діагностики супутніх захворювань. Це породжує проблему відсутності можливості безперешкодно діагностуватись для людей поза великими населеними пунктами. Сучасні методи діагностики дуже габаритні та дорогі, тому є недоступними для великої частини планети, а отже потрібно докладати зусиль, щоб доступ до сучасної медицини був у якомога більшої кількості людей. В свою чергу прогрес у сфері електроніки призвів до появи систем, що дозволяють отримати зображення сітківки ока за допомогою камери смартфона та невеликого приладу, що об'єднує в собі декілька лінз та джерело світла.

2. Методи обробки біомедичних зображень, що були винайдені у минулому столітті все ще підтверджують свою актуальність. Проте все більше дослідницьких робіт проводиться із новими методами, все більше нових методів з'являється. Такі методи часто вирішують недоліки попередників, проте створюють недоліки наступникам, адже прогрес не стоїть на місці і з приходом нової апаратури з'являються нові проблеми.

3. Машинне навчання це не тільки наше майбутнє, а й наше сьогодення. Штучний інтелект впроваджують у більшість сфер людського життя задля його полегшення. Цей факт не оминув і медицину, про що свідчать численні дослідження у сфері виявлення ранніх стадій хвороби за допомогою машинного навчання. Використання таких систем однозначно пішло на користь людям і допомогло покращити та спростити багато сфер нашого життя.

4. Задача розпізнавання діабетичної ретинопатії за аналізом зображень сітківки ока доволі актуальна на сьогоднішній день. У роботі було наведено декілька досліджень на цю ж тему і точність їх моделей складала 99% та 93%. Можемо говорити про те, що такі системи дійсно можуть краще за людей визначати такі складні захворювання як діабетична ретинопатія. Загалом, серед власних моделей найкращу точність на рівні 83% показала модель, для якої вхідні зображення, як і тестові, оброблялись за допомогою функції еквалізації гістограм яскравості. Близькою за точністю була модель, для якої вхідні зображення, окрім вирівнювання гістограм також оброблялись за допомогою оператора Превітта. Результат близький до 80% обумовлений низькими шумами на виході, а також хорошою ефективністю у визначенні вертикальних та горизонтальних країв.

5. Попри непоганий результат, варто зазначити, що нейронні мережі мають суттєвий недолік у вигляді потреби в ресурсах. Щоб збільшити точність необхідно більше даних, а отже і більші потужності для навчання моделі. Обрана модель EfficientNetB0 має суттєву перевагу у вигляді порівняно швидкої обробки. Фактично дана модель вже не нова і є можливість використовувати більш нові моделі, які фактично є доопрацюванням цієї, але питання в тому, чи є економічно обґрунтованим витрачання вдвічі більших потужностей чи часу. Наприклад, якщо EfficientNetB0 в експерименті проходила одну епоху приблизно за 160 секунд, то модель VGG16 витрачала на ту ж епоху більше 300 секунд.

Отже, для покращення точності для даної моделі можна використовувати більш нові попередньо навчені моделі, більш сучасні методи обробки зображень та в цілому більше вхідних даних, а тому залучити значно більше потужностей для навчання.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Retinal Blood Vessel Segmentation by Using Matched Filtering and Fuzzy C-means Clustering with Integrated Level Set Method for Diabetic Retinopathy Assessment / N. Memari et al. *Journal of Medical and Biological Engineering*. 2018. Vol. 39, no. 5. P. 713–731. DOI: <https://doi.org/10.1007/s40846-018-0454-2>
2. Vision impairment and blindness. *World Health Organization (WHO)*. URL: <https://www.who.int/news-room/fact-sheets/detail/blindness-and-visual-impairment>
3. Diabetes. *World Health Organization (WHO)*. URL: <https://www.who.int/news-room/fact-sheets/detail/diabetes>
4. Ukraine diabetes report 2000 – 2045. *IDF Diabetes Atlas*. URL: <https://diabetesatlas.org/data/en/country/207/ua.html>
5. Населення України (2021). URL: <https://index.minfin.com.ua/ua/reference/people/2021/> (дата звернення: 05.01.2024).
6. Klein, Ronald, and Barbara EK Klein. "Diabetic eye disease." *The Lancet* 350.9072 (1997)
7. Що таке діабетична ретинопатія? | Офтальмологічний центр "Діоптрія". *Офтальмологічний центр "Діоптрія"*. URL: <https://dioptriya.com.ua/uk/medychna-biblioteka/shho-take-diabetychna-retynopatiya/>
8. Діабетична ретинопатія сітківки очей при цукровому діабеті: симптоми, стадії і класифікація. URL: <https://perebus.com.ua/diabetichna-retinopatiya-sitkivki-ochey-pri-cukrovomu-diabeti-simptomi-stadii-i-klasifikaciya/>
9. Diabetic retinopathy 2015 data colored resized. *Kaggle: Your Machine Learning and Data Science Community*.

URL: <https://www.kaggle.com/datasets/sovitrath/diabetic-retinopathy-2015-data-colored-resized>

10. Hytowitz A. N. Review of using the Dyop optotype for acuity and refractions per the article: <https://www.sciencedirect.com/science/article/pii/S1888429622000656>. *Journal of Optometry*. 2023. URL: <https://doi.org/10.1016/j.optom.2022.12.002> (date of access: 05.01.2024).

11. Протоколи надання медичної допомоги хворим на цукровий діабет. *Здоров'я України*. URL: [https://angiology.com.ua/ua/archive/2009/7\(26\)/article-260/protokoli-nadannya-medichnoyi-dopomogi-hvorim-na-cukroviiy-diabet](https://angiology.com.ua/ua/archive/2009/7(26)/article-260/protokoli-nadannya-medichnoyi-dopomogi-hvorim-na-cukroviiy-diabet).

12. Huang D, Swanson EA, Lin CP, Schuman JS, Stinson WG, Chang W, Hee MR, Flotte T, Gregory K, Puliafito CA, Fujimoto JG. Optical coherence tomography. *Science*. 1991;**254**:1178–1181. DOI: <https://doi.org/10.1126/science.1957169>

13. Optical coherence tomography: an emerging technology for biomedical imaging and optical biopsy / J. G. Fujimoto et al. *Neoplasia*. 2000. Vol. 2, no. 1-2. P. 9–25. DOI: <https://doi.org/10.1038/sj.neo.7900071>

14. Lincoff H., Kreissig I. Optical coherence tomography of pneumatic displacement of optic disc pit maculopathy. *British journal of ophthalmology*. 1998. Vol. 82, no. 4. P. 367–372. DOI: <https://doi.org/10.1136/bjo.82.4.367> (date of access: 05.01.2024).

15. OCTID: optical coherence tomography image database / P. Gholami et al. *Computers & electrical engineering*. 2020. Vol. 81. P. 106532. DOI: <https://doi.org/10.1016/j.compeleceng.2019.106532>

16. Fercher AF, et al. Optical coherence tomography - principles and applications. *Rep Prog Phys*. 2003;66(2):239–303.

17. Chopra R, Wagner SK, Keane PA. Optical coherence tomography in the 2020s—outside the eye clinic. *Eye*. 2021;35(1):236-243. DOI:10.1038/s41433-020-01263-6

18. What is fluorescein angiography?. *American Academy of Ophthalmology*. URL: <https://www.aao.org/eye-health/treatments/what-is-fluorescein-angiography> (date of access: 05.01.2024).
19. Xu R, Teich W, Frenzel F, Hoffmann K, Radke J, Rösler J, Faust K, Blank A, Brandenburg S, Misch M, Vajkoczy P, Onken JS, Resch-Genger U. Optical Characterization of Sodium Fluorescein *In Vitro* and *Ex Vivo*. *Front Oncol* 2021.
20. Athertya J. S., Saravana Kumar G. Automatic segmentation of vertebral contours from CT images using fuzzy corners. *Computers in biology and medicine*. 2016. Vol. 72. P. 75–89. DOI: <https://doi.org/10.1016/j.compbimed.2016.03.009>
21. Ophthalmoscopy eye examination - maculacenter.com. *MaculaCenter.com*. URL: <https://maculacenter.com/eye-procedures/ophthalmoscopy/>
22. George T. Timberlake, Michael Kennedy M.D. The direct ophthalmoscope how it works and how to use it. University of Kansas Medical Center, 2005. 39 p.
23. Panwar N, Huang P, Lee J, Keane PA, Chuan TS, Richhariya A, Teoh S, Lim TH, Agrawal R. Fundus Photography in the 21st Century--A Review of Recent Technological Advances and Their Implications for Worldwide Healthcare
24. Bilson J. D., Lapworth S. J. J.S. Athertya, G. Saravana Kumar, “Automatic segmentation of vertebral contours from CT images using fuzzy corners” *Computers in biology and medicine*. 2017. Vol. 85. P. 24. DOI: <https://doi.org/10.1016/j.compbimed.2017.04.005>
25. Chalam K. V., Chamchikh J., Gasparian S. Optics and utility of low-cost smartphone-based portable digital fundus camera system for screening of retinal diseases. *Diagnostics*. 2022. Vol. 12, no. 6. P. 1499. DOI: <https://doi.org/10.3390/diagnostics12061499>
26. Use of smartphones to detect diabetic retinopathy: scoping review and meta-analysis of diagnostic test accuracy studies / C. H. Tan et al. *Journal of medical internet research*. 2020. Vol. 22, no. 5. P. e16658. DOI: <https://doi.org/10.2196/16658>

27. How to use a smartphone for fundus photography to capture quality images on the fly. *Eyes On Eyecare*. URL: <https://eyesoneyecare.com/resources/how-to-use-a-smartphone-for-fundus-photography-to-capture-quality-images-on-the-fly/>
28. Pivotal evaluation of an artificial intelligence system for autonomous detection of referable and vision-threatening diabetic retinopathy / E. Ipp et al. *JAMA network open*. 2021. Vol. 4, no. 11. P. e2134254. DOI: <https://doi.org/10.1001/jamanetworkopen.2021.34254>
29. Фильтрация изображений на C# - Программирование на C, C# и Java. *Программирование на C, C# и Java*. URL: <https://vscode.ru/prog-lessons/filtratsiya-izobrazheniy-na-c-sharp.html>
30. Лавер В.О., Левчук О.М. Обробка зображень: навч.-метод. посіб. / В.О. Лавер, О.М. Левчук. – Ужгород : вид-во ПП «АУТДОР - ШАРК», 2021. – 51 с.
31. How to use gamma correction for image processing with opencv. *Dynamsoft Developers Blog*. URL: <https://www.dynamsoft.com/codepool/image-processing-opencv-gamma-correction.html>
32. Singnoo J., Finlayson G. D. Understanding the gamma adjustment of images. *Color and imaging conference*. 2010. Vol. 18, no. 1. P. 134–139. DOI: <https://doi.org/10.2352/cic.2010.18.1.art00024>
33. Palanisamy G., Ponnusamy P., Gopi V. P. An improved luminosity and contrast enhancement framework for feature preservation in color fundus images. *Signal, image and video processing*. 2018. Vol. 13, no. 4. P. 719–726. DOI: <https://doi.org/10.1007/s11760-018-1401-y>
34. Otsu, N., “A threshold selection method from gray-level histograms”, *IEEE Trans. Systems, Man, and Cybernetics*, 9(1), pp.62–66 (1979)
35. Bataineh B., Abdullah S. N. H. S., Omar K. An adaptive local binarization method for document images based on a novel thresholding method and dynamic

windows. *Pattern recognition letters*. 2011. Vol. 32, no. 14. P. 1805–1813.
DOI: <https://doi.org/10.1016/j.patrec.2011.08.001>

36. Niblack, W.: An introduction to digital image processing (1985)

37. Sauvola, J., Seppanen, T., Haapakoski, S., Pietikainen, M.: Adaptive document binarization. In: Fourth International Conference Document Analysis and Recognition (ICDAR), Ulm, Germany (1997)

38. Khurshid, K., Siddiqi, I., Faure, C., Vincent, N.: Comparison of Niblack inspired Binarization methods for ancient documents. In: 16th International conference on Document Recognition and Retrieval. SPIE, USA (2010)

39. Kefali, A., Sari, T., Sellami, M.: Evaluation of several binarization techniques for old Arabic documents images. In: The First International Symposium on Modeling and Implementing Complex Systems MISC 2010, Constantine, Algeria, pp. 88–99 (2010)

40. Canny J. A computational approach to edge detection. *IEEE transactions on pattern analysis and machine intelligence*. 1986. PAMI-8, no. 6. P. 679–698.
DOI: <https://doi.org/10.1109/tpami.1986.4767851> (date of access: 05.01.2024).

41. Pradeep Kumar Reddy R, Nagaraju C, Rajasekhar Reddy I (2015) Canny scale edge detection. *IJETT*. DOI: <https://doi.org/10.14445/22315381/IJETT-ICGTETM-N3/ICGTETM-P121>

42. Ito, K. KaiqiXiong, "Gaussian filters for nonlinear filtering problems", *Automatic Control, IEEE Transactions on* Volume:45 , Issue: 5 ISSN : 0018-9286, Pages 910 – 927

43. Image edge detection operators in digital image processing - geeksforgeeks. *GeeksforGeeks*. URL: <https://www.geeksforgeeks.org/image-edge-detection-operators-in-digital-image-processing/>

44. Irwin Sobel. An Isotropic 3x3 Image Gradient Operator. 2014.

45. Concept of Edge Detection - Javatpoint. *www.javatpoint.com*.
URL: <https://www.javatpoint.com/dip-concept-of-edge-detection>

46. Anirban Kar Chaudhuri. Edge Detection Methods Comparison. medium.com. URL: <https://anirban-karchaudhuri.medium.com/edge-detection-methods-comparison-9e4b75a9bf87>.
47. Roberts Machine Perception of 3-D Solids, Optical and Electro-optical Information Processing, MIT Press 1965
48. Prewitt, J.M.S. (1970). "Object Enhancement and Extraction". *Picture processing and Psychopictorics*. Academic Press
49. Fuzzy based image edge detection algorithm for blood vessel detection in retinal images / F. Orujov et al. *Applied soft computing*. 2020. Vol. 94. P. 106452. DOI: <https://doi.org/10.1016/j.asoc.2020.106452> (date of access: 05.01.2024).
50. The edge detectors suitable for retinal OCT image segmentation. *PubMed Central (PMC)*. URL: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5585566/>
51. Batta M. Machine Learning Algorithms -A Review. *International Journal of Science and Research (IJSR)*. 2018. DOI: <https://doi.org/10.21275/ART20203995>.
52. A-Z guide to the types of machine learning problems. *ProjectPro*. URL: <https://www.projectpro.io/article/types-of-machine-learning/623>
53. Машинне навчання та обробка сигналів в біомедичних електронних системах. Конспект лекцій [Електронний ресурс]: навч. посіб. для студ. спеціальності 153 «Мікро- та наносистемна техніка», освітньої програми «Електронні мікро- і наносистеми та технології» / КПІ ім. Ігоря Сікорського;– Київ: КПІ ім. Ігоря Сікорського, 2020. – 97 с.
54. VanderPlas J. Python Data Science Handbook. — O'Reilly Media; 1st edition, 2016. — P. 388–390.
55. Reinforcement learning - GeeksforGeeks. *GeeksforGeeks*. URL: <https://www.geeksforgeeks.org/what-is-reinforcement-learning/>

56. Grossi E., Buscema M. Introduction to artificial neural networks. *European journal of gastroenterology & hepatology*. 2007. Vol. 19, no. 12. P. 1046–1054. DOI: <https://doi.org/10.1097/meg.0b013e3282f198a0>
57. Sanghvi K. Image classification techniques. *medium.com*. URL: <https://medium.com/analytics-vidhya/image-classification-techniques-83fd87011cac>.
58. Що таке нейронні мережі і як вони працюють - *Root-Nation.com*. *Root-Nation.com*. URL: <https://root-nation.com/ua/articles-ua/tech-ua/ua-what-is-neural-network/>
59. Редчиць Є. АЛЬТЕРНАТИВНИЙ МЕТОД ДЕКОДУВАННЯ МОВНОГО СИГНАЛУ ЗГОРТКОВОЮ НЕЙРОННОЮ МЕРЕЖЕЮ : магістерська робота. 2021. 67 с.
60. Accuracy and loss - AI wiki. *Artificial Intelligence Wiki - AI Wiki*. URL: <https://machine-learning.paperspace.com/wiki/accuracy-and-loss#relationship-between-accuracy-and-loss>
61. Papers with code - medical image classification. *The latest in Machine Learning | Papers With Code*. URL: <https://paperswithcode.com/task/medical-image-classification>
62. A convolutional neural network for the screening and staging of diabetic retinopathy. *Home* - *PLOS*. URL: <https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0233514#pone.0233514.ref014>
63. A deep learning-based framework for retinal disease classification / A. Choudhary et al. *Healthcare*. 2023. Vol. 11, no. 2. P. 212. DOI: <https://doi.org/10.3390/healthcare11020212>

64. Convolutional neural networks for diabetic retinopathy / H. Pratt et al. *Procedia computer science*. 2016. Vol. 90. P. 200–205. DOI: <https://doi.org/10.1016/j.procs.2016.07.014> (date of access: 05.01.2024).
65. Kundu N. Exploring resnet50: an in-depth look at the model architecture and code implementation. *medium.com*. URL: <https://medium.com/@nitishkundu1993/exploring-resnet50-an-in-depth-look-at-the-model-architecture-and-code-implementation-d8d8fa67e46f>.
66. VGG very deep convolutional networks (vggnet) - what you need to know - viso.ai. URL: <https://viso.ai/deep-learning/vgg-very-deep-convolutional-networks/>
67. Diabetic retinopathy detection | kaggle. *Kaggle: Your Machine Learning and Data Science Community*. URL: <https://www.kaggle.com/c/diabetic-retinopathy-detection/overview>
68. Keras: deep learning for humans. *Keras: Deep Learning for humans*. URL: <https://keras.io>
69. Lenko M. Instructions for transfer learning with pre-trained CNNs. *medium.com*. URL: <https://medium.com/@mikhailenko/instructions-for-transfer-learning-with-pre-trained-cnns-203ddaefc01>.
70. A survey on image data augmentation for deep learning - journal of big data. *SpringerOpen*. URL: <https://journalofbigdata.springeropen.com/articles/10.1186/s40537-019-0197-0>
71. OpenCV: histogram equalization. *OpenCV documentation index*. URL: https://docs.opencv.org/4.x/d4/d1b/tutorial_histogram_equalization.html

Додаток А

Скрипт до задачі розпізнавання діабетичної ретинопатії за аналізом зображень
сітківки ока

```
import matplotlib.pyplot as plt
import cv2 as cv2
import numpy as np
from keras.models import Sequential
from keras.optimizers import Adam
import efficientnet.tfkeras as efn
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.layers import Flatten, Dense

def image_processing(image):
    image_uint = image.astype('uint8')

    ### Equalization 1 channel
    # g_image = image_uint[:, :, 1]
    # image_equalized = cv2.equalizeHist(g_image)

    ### Equalization 3 channel
    image_equalized = cv2.merge([cv2.equalizeHist(channel) for channel in
cv2.split(image_uint)])

    ### LAPLACIAN
    # output = cv2.Laplacian(image_equalized, cv2.CV_8U)

    ### GAUSIAN
    # output = cv2.GaussianBlur(res, (3, 3), 0)

    ### SOBEL
    # sobelx = cv2.Sobel(image_equalized, cv2.CV_64F, 1, 0, ksize=1)
    # sobely = cv2.Sobel(image_equalized, cv2.CV_64F, 1, 0, ksize=1)
    # sobel = np.sqrt(sobelx ** 2 + sobely ** 2)
    # output = np.clip(sobel, 0, 255)
```

```

#### CANNY
# canny = cv2.Canny(image_equalized, 100, 200).astype('float64')
# output = cv2.merge((canny, canny, canny))

#### PREWITT
# kernel_x = np.array([[1, 1, 1], [0, 0, 0], [-1, -1, -1]])
# kernel_y = np.array([[ -1, 0, 1], [-1, 0, 1], [-1, 0, 1]])
#
# prewitt_x = cv2.filter2D(image_equalized, -1, kernel_x).astype('float64')
# prewitt_y = cv2.filter2D(image_equalized, -1, kernel_y).astype('float64')
# prewitt = np.sqrt(prewitt_x ** 2 + prewitt_y ** 2)
# output = np.clip(prewitt, 0, 255)

#### GAMMA
# output = np.clip(1 * image_uint ** 1.25, 0, 255)
return output.astype('float64')

```

```

# Розмір вхідного зображення
IMAGE_SIZE = (224, 224)

```

```

# Шлях до папок з тренувальними та тестовими даними
TRAIN_DIR = 'C:\\Study\\Diploma_data\\Train'
VALID_DIR = 'C:\\Study\\Diploma_data\\Valid'

```

```

# Формування зображень із застосуванням аугментації та методів обробки
train_datagen = ImageDataGenerator(rescale=1. / 255,
                                   zoom_range=0.1,
                                   width_shift_range=0.1,
                                   height_shift_range=0.1,
                                   rotation_range=25
                                   # , preprocessing_function=image_processing
                                   )
valid_datagen = ImageDataGenerator(rescale=1. / 255,
                                   zoom_range=0.1,
                                   width_shift_range=0.1,
                                   height_shift_range=0.1,
                                   rotation_range=25,
                                   validation_split=0.75
                                   # , preprocessing_function=image_processing
                                   )

```

```
# Формування пар значень зображення-мітка
train_data = train_datagen.flow_from_directory(TRAIN_DIR,
                                              target_size=IMAGE_SIZE,
                                              batch_size=32,
                                              class_mode='binary',
                                              subset='training')

valid_data = valid_datagen.flow_from_directory(VALID_DIR,
                                              target_size=IMAGE_SIZE,
                                              batch_size=32,
                                              class_mode='binary',
                                              subset='validation')

# Отримання кількості даних для гістограми
train_class_counts = train_data.classes
valid_class_counts = valid_data.classes

# Виведення гістограм для тренувальних даних
plt.figure(figsize=(6, 6))
plt.hist(train_class_counts * 1.0, 5, width=0.2, alpha=1, color='skyblue', edgecolor='black')
plt.title('Гістограма тренувальних даних за класами')
plt.xlabel('Клас')
plt.ylabel('Кількість')
plt.show()

# Виведення гістограми для валідаційних даних
plt.figure(figsize=(6, 6))
plt.hist(valid_class_counts * 1.0, 5, width=0.2, alpha=1, color='skyblue', edgecolor='black')
plt.title('Гістограма тестових даних за класами')
plt.xlabel('Клас')
plt.ylabel('Кількість')
plt.show()

# Виведення випадкових зображень з тренувальної вибірки
plt.figure(figsize=(10, 10))
for images, labels in train_data:
    labels = list(labels.astype('int'))
    for i in range(6):
```

```

ax = plt.subplot(2, 3, i + 1)
plt.imshow(images[i])
if labels[i] == 1:
    title = 'Здоровий'
else:
    title = 'Хворий'
plt.title(title)
plt.axis("off")
break

```

Виведення випадкових зображень з тестової вибірки

```

plt.figure(figsize=(10, 10))
for images, labels in valid_data:
    labels = list(labels.astype('int'))
    for i in range(6):
        ax = plt.subplot(2, 3, i + 1)
        plt.imshow(images[i])
        if labels[i] == 1:
            title = 'Здоровий'
        else:
            title = 'Хворий'
        plt.title(title)
        plt.axis("off")
    break

```

```
plt.show()
```

Побудова моделі

```
epochs = 2
```

```

model = Sequential([
    efn.EfficientNetB0(
        input_shape=(224, 224, 3),
        weights='imagenet',
        include_top=False,
    ),
    Flatten(),
    Dense(1, activation='sigmoid')
])

```

```
model.compile(optimizer=Adam(learning_rate=0.0001), loss='binary_crossentropy',
metrics=['accuracy'])
history = model.fit(train_data, validation_data=valid_data, epochs=epochs, verbose=1)

model.save('model_name.keras')

# Побудова графіків точності та втрат для тренувальної та тестової вибірок
acc = history.history['accuracy']
val_acc = history.history['val_accuracy']

loss = history.history['loss']
val_loss = history.history['val_loss']

epochs_range = range(epochs)

plt.figure(figsize=(8, 8))
plt.subplot(1, 2, 1)
plt.plot(epochs_range, acc, label='Training Accuracy', color='black', lw=1.5)
plt.plot(epochs_range, val_acc, label='Validation Accuracy', color='red', lw=1.5)
plt.ylim(0, 1)
plt.grid(True)
plt.legend(loc='lower right')
plt.title('Training and Validation Accuracy')

plt.subplot(1, 2, 2)
plt.plot(epochs_range, loss, label='Training Loss', color='black', lw=1.5)
plt.plot(epochs_range, val_loss, label='Validation Loss', color='red', lw=1.5)
plt.ylim(bottom=0) # Встановити нижню межу графіка втрат
plt.grid(True) # Додати сітку
plt.legend(loc='upper right')
plt.title('Training and Validation Loss')
plt.show()

print('End')
```

Додаток Б
Підсумкова таблиця точності отриманої моделі з використанням методів обробки
зображень

Таблиця Б.1 - Таблиця точності отриманої моделі на основі EfficientNetV0 з використанням методів обробки зображень

	Точність
Дані без обробки без еквалізації	75%
Дані без обробки з еквалізацією	83%
Оператор Кенні	60%
Оператор Собеля	76%
Оператор Лапласа	68%
Оператор Превітта	80%