

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

Приладобудівний
Комп'ютерно інтегрованих оптичних та навігаційних систем

До захисту допущено:

Завідувач кафедри

_____ Н. І. Бурау

« ____ » _____ 2022 р.

Дипломний проект

на здобуття ступеня бакалавра

**за напрямком підготовки 151 «Автоматизація та комп'ютерно-інтегровані
технології»**

**на тему: «Розробка алгоритмічного та програмного забезпечення для
виявлення облич на основі нейронних мереж»**

Виконав:

студент II курсу, групи ПО-11мп

Проскурін Вадим Сергійович _____

Керівник:

д. т. н., проф.,

Колобродов Валентин Георгійович _____

Рецензент:

к. т. н., доцент, _____

Засвідчую, що у цьому дипломному
проекті немає запозичень з праць інших
авторів без відповідних посилань.

Студент (-ка) _____

**Національний технічний університет України
«Київський політехнічний інститут
імені Ігоря Сікорського»**

Рівень вищої освіти –другий (магістерський)

Спеціальність – 151 – Автоматизація та комп'ютерно-інтегровані технології

Освітньо-професійна програма - Комп'ютерно-інтегровані системи та технології в приладобудуванні

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ Надія БУРАУ

«__» _____ 20__ р.

**ЗАВДАННЯ
на магістерську дисертацію студента
Проскурін Вадим Сергійович**

1. Тема дисертації «Розробка алгоритмічного та програмного забезпечення для виявлення облич на основі нейронних мереж»,
науковий керівник дисертації Колобродов Валентин Георгійович, д. т. н.,
проф., затверджені наказом по університету від «__» _____ 20__ р.
№ _____

2. Термін подання студентом дисертації «_10_» грудня 2022 р.

3. Об'єкт дослідження _ виявлення облич на основі нейронних мереж.

4. Предмет дослідження: розробка алгоритмічного та програмного забезпечення для виявлення облич.

5. Перелік завдань, які потрібно розробити

5.1. Огляд існуючих методів та технологій виявлення облич. Визначення недоліків таких систем.

5.2. Обґрунтування запропонованого підходу розробки нейронної мережі для виявлення облич. Огляд існуючих методів зменшення дефектів зображення

5.3 Дослідження технологій зменшення дефектів гаусового шуму.

5.4. Дослідження технологій збільшення різкості розмитих зображень.

5.5. Розробка алгоритмічного та програмного забезпечення.

6. Орієнтовний перелік ілюстративного матеріалу презентація в програмі PowerPoint: схеми і теоретичні співвідношення, завдяки яким викладаються зміст і основні висновки результатів дослідження при виконанні магістерської дисертації.

7. Орієнтовний перелік публікацій:

- стаття в збірнику праць науково-практичної конференції студентів, аспірантів та молодих вчених «Ефективність та Автоматизація інженерних рішень у приладобудуванні».

8. Консультанти розділів дисертації*

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

9. Дата видачі завдання 31.10. 2022р_____

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації	Примітка
1	Огляд існуючих методів та технологій виявлення облич. Визначення недоліків таких систем.	05.11.22р	05.11.22р
2	Обґрунтування запропонованого підходу розробки нейронної мережі для виявлення облич. Огляд існуючих методів зменшення дефектів зображення.	08.11.22р	08.11.22р
3	Дослідження технологій зменшення дефектів гаусового шуму.	16.11.22р	16.11.22р
4	Дослідження технологій збільшення різкості розмитих зображень.	22.11.22р	22.11.22р
5	Розробка алгоритмічного та програмного забезпечення.	02.12.22р	02.12.22р
6	Підготовка до захисту магістерської дисертації	10.12.22р	10.12.22р

Студент

(підпис)

(ініціали, прізвище)

Науковий керівник дисертації

(підпис)

(ініціали, прізвище)

* Консультантом не може бути зазначено наукового керівника магістерської дисертації.

Пояснювальна записка
до дипломного проекту
на тему: «Розробка алгоритмічного та програмного
забезпечення для виявлення облич на основі нейронних мереж»

Київ – 2022 року

АНОТАЦІЯ

Магістерська дисертація на тему : «Розробка алгоритмічного та програмного забезпечення для виявлення облич на основі нейронних мереж». В ході роботи були досліджені типи нейронних мереж, проведений аналіз NN для зчитування облич, були розглянуті різні типи фільтрації. На основі проведених досліджень, були запропоновано використання деконволюції Вінера для вирішення проблем пошкоджених зображень. Була розроблена власна CNN для зчитування облич та застосовані реалізовані технології на шумних зображеннях. На основі власних напрацювань був реалізований стартап проект.

Дисертаційна робота містить 4 розділи, налічує 90 сторінок друкованого тексту без додатків. Робота включає в себе 22 таблиць, 44 рисунки, 5 додатки, та список використаних джерел у кількості 46 одиниці літератури.

В першому розділі проекту було виконано огляд існуючих нейронних мереж. Були описані за якими принципами вони працюють, історія їх розробки, та з якими перешкодами можуть зіткнутися дані системи.

В другому розділі були описані як задачі виконує комп'ютерний зір. розглянуті популярні методи фільтрації вхідного зображення. Проведений аналіз фільтру різкості та маски нечіткості для підвищення якості зображення. Фільтр гаусу, розмиття по медіані та швидкі NІ для зменшення гаусового шуму. Були розглянуті оператори Щару, Собеля та метод Оцу для виявлення границь зображення

В третьому розділі на основі проведених досліджень, була розроблена власна нейронна мережа для виявлення облич. Реалізовані технології зменшення шуму та підвищення якості розмитих зображень на основі Вініровской деконволюції. Були порівнянні розроблені методи фільтрації з представленими в другому розділі аналогами

В четвертому розділі був розроблений стартап проект та проаналізовано можливості і стратегії його реалізації.

Ключові слова: нейронні мережі, графічна фільтрації, згорткова нейронна мережа, деконволюція Вінера, каскади Хара, комп'ютерний зір.

ABSTRACT

Master's thesis on the topic: "Development of algorithmic and software for revealing appearances based on neural networks". In the course of the robots, there were completed types of neural networks, analysis of NN for reading appearances, and different types of filtering were examined. On the basis of the conducted research, we proposed the deconvolution of Wiener for the solution of the problems of advanced images. The bula was broken up by CNN to read the denunciation and stalemate of the implementation of technology on noisy images. On the basis of the authoritative guidelines for the implementation of a start-up project.

The dissertation work was divided into 4 divisions, there are 90 sides of the written text without supplements. The work includes 22 tables, 44 drawings, 5 appendices, and a list of quotations from 46 pieces of literature.

In the first part of the project, an overview of the essential neural networks was made. If they are described according to some principles, they work, the history of their developments, and these systems can be closed with some kind of transitions.

In another section, they described how the tasks of viconuing the computer space were described. reviewed popular methods of filtering the input image. Conducting an analysis of the sharpness filter and the fuzziness mask to improve the sharpness of the image. Gaussian filter, adjusted by median and NI to change Gaussian noise. We looked at Sharu and Sobel operators and Otsu's method for revealing the image boundary

In the third section, on the basis of the investigations, the bule was divided into a neural network for revealing the appearance. Implementation of technologies for noise reduction and image brightness enhancement based on Venerian deconvolution. Boulli por_vnyanni razrobleni methods and filtratsii s presented in another razdil_ analogues

In the fourth division, the start-up project was analyzed in terms of the feasibility and strategy of its implementation.

Key words: neural networks, graphic filtering, hump neural array, Wiener deconvolution, Hare cascades, computer sir.

ЗМІСТ

СПИСОК СКОРОЧЕНЬ.....	5
ВСТУП.....	6
РОЗДІЛ 1. ОГЛЯД ТА АНАЛІЗ ЛІТЕРАТУРИ.....	8
1.1. Нейронні мереж.....	8
1.2 Історія нейронних мереж.....	11
1.3 Принципи роботи нейронних мереж для обробки зображення	13
1.4 Сучасні нейронні мережі для обробки зображення	14
1.5 Технології для обробки відео потоку	15
1.6 Технології зменшення шумів зображення	16
1.7Висновки до першого розділу	17
 РОЗДІЛ 2. ОБҐРУНТУВАННЯ ФУНКЦІОНАЛЬНОГО ПІДХОДУ ТА	
ПРИНЦИП РОБОТИ НЕЙРОННОЇ МЕРЕЖІ	19
2.1 Нейронні мережі.....	19
2.2 Задачі комп'ютерного бачення	26
2.3 Технології підвищення якості	28
2.4 Технології зменшення шумів.....	30
2.5 Технології виявлення країв об'єкту.....	36
2.6 Висновки до другого розділу	41

АЛГОРИТМИ ТА ТЕХНІЧНІ ХАРАКТЕРИСТИКИ НЕЙРОННОЇ МЕРЕЖІ.....	43
3.1 Згорткова нейронна мережа	43
3.2 Принцип роботи CNN для розпізнавання обличчя	46
3.3 Підготовка шуканих зображень. Каскади Хаара	47
3.4 Навчання моделі для нашої NN	51
3.5 Зменшення шумів та ефекту розмиття.....	47
3.6 Розроблена CNN	57
3.3 Висновки до розділу	61
РОЗДІЛ 4. РОЗРОБКА СТАРТАП-ПРОЕКТУ «CNN ДЛЯ РОЗПІЗНАВАННЯ ОБЛИЧ З МОЖЛИВІСТЮ РОБОТИ ЗІ ШУМНИМИ ТА РОЗМИТИМИ ЗОБРАЖЕННЯМИ».....	63
4.1 Опис ідеї проекту	63
4.2 Технологічний аудит ідеї проекту	65
4.3 Аналіз ринкових можливостей запуску стартап-проекту	66
4.4 Розроблення ринкової стратегії проекту	72
4.5 Розроблення ринкової стратегії проекту	75
4.6 Висновки	76
ЗАГАЛЬНІ ВИСНОВКИ.....	79
СПИСОК ЛІТЕРАТУРИ	80

СПИСОК СКОРОЧЕНЬ

NN – Neural networks (нейронні мережі)

MV - Machine vision (машинне бачення)

HSV – Hue, saturation, value

CNN – Convolutional neural network (конволюційна нейронна мережа)

RNN – Recurrent neural networks (повторна нейронна мережа)

LSTM – Long short-term memory (довга короткострокова пам'ять)

GAN – Generative adversarial networks (генеральна змагальна мережа)

ВСТУП

Бажання науковців симулювати роботу мозку та створити систему, яка зможе вчитися без участі розробника, з'явилося одразу після появи перших комп'ютерів, проте технологічний прогрес тоді ще не досягнув теперішніх можливостей. Перші прототипи моделей нейронних мереж були створені не більше ніж 50 років тому. Вони використовували примітивні алгоритми навчання та не мали фізичних уявлень, але це надало ґрунт для появи машинних систем у майбутньому.

Використання нейронних мереж здобуло велику популярність в теперішні часи, хоча алгоритми зробили величезний технологічний крок, проте вони й досі далекі від ідеалу. Нинішні системи є дуже трудомісткими в розробці та потребують багато часу на навчання та виправлення помилок, але можливість доводити до автоматизму обробку великої кількості даних з корегуванням самого процесу зробило ці машини дуже популярними у нинішніх компаніях ІТ гігантів.

Нейронні мережі – це комп'ютерне уявлення розумових процесів, послідовності нейронів, з'єднаних між собою синапсами. Завдяки такій структурі машина має здатність аналізувати і навіть запам'ятовувати різну інформацію. Нейронні мережі також здатні не лише аналізувати вхідні дані, але й відтворювати їх зі своєї пам'яті. Розробник розписує реакції програми на вхідні дані та корегує її підсумкові вихідні дані. Вони мають широку галузь застосування, починаючи з класифікації даних та закінчуючи машинним зором.

Нині найбільш поширені нейронні системи використовуються для машинного зору чи обробки зображення або відео потоків. Все частіше і частіше ми бачимо як машині мережі можуть обробляти або створювати ілюстрації, малювати картини, знаходити об'єкти. Ці всі можливості зробили

їх дуже потужним інструментом у військовій та соціальній сферах. Головна перевага нейронних мереж — це здатність розпізнавати глибші, іноді несподівані закономірності даних та виконувати потужні математичні перетворення.

Коли нейронна мережа отримує зображення або відео потік, вона розбиває вхідну інформацію на пікселі та пропускає через цифрові шари нейронів. Кожен шар виконує певні математичні згортки інформації, це дозволяє виявити різні характеристики вхідного сигналу. На якість результуючого зображення впливає глибина нейронної мережі. Чим більше шарів між входом та виходом, тим більш детально буде опрацьований результат.

При роботі з нейронними мережами для обробки зображення можна виділити дві основні проблеми:

- 1) При роботі з відео потоками при поганій освітлюваності або при пошкодженому обладнанні, вихідне зображення піддається великій кількості шумів, що не дозволяє машинному зору знаходити об'єкти.
- 2) При спотворенні спостережуваного об'єкту нейронна мережа часто не може знайти збіги, що погіршує результат.

РОЗДІЛ 1. ОГЛЯД ТА АНАЛІЗ ЛІТЕРАТУРИ

1.1. Нейронні мережі

Нейронні мережі — це послідовність нейронів пов'язаних між собою синапсами, серією алгоритмів, які намагаються розпізнавати зв'язки в наборі вхідних даних, за допомогою процесу, який імітує роботу людського мозку...

Для більш детального розуміння чим є нейронні мережі, ми повинні звернути увагу на їх людський прототип – мозок, бо в основі роботи NN лежить імітація наших розумових здібностей. Людський мозок – це дуже складна паралельна і нелінійна система обробки інформації. Він має здатність організовувати свої нейронні зв'язки таким чином, щоб виконувати обробку даних у лічені секунди, набагато швидше, ніж найпотужніший цифровий комп'ютер. Для більшого розуміння потужностей мозку розберемося як працює людський зір, який є дуже явним прикладом обробки інформації. При надходженні світла до нашого ока, зоровий нерв поглинає вхідну інформацію, та відправляє її до нашого центру прийняття рішень. Мозок регулярно виконує завдання розпізнавання знайомих образів у незнайомих обставинах приблизно через 100-200 мс, тоді, як завдання набагато меншої складності, може зайняти кілька днів для звичайного комп'ютера.

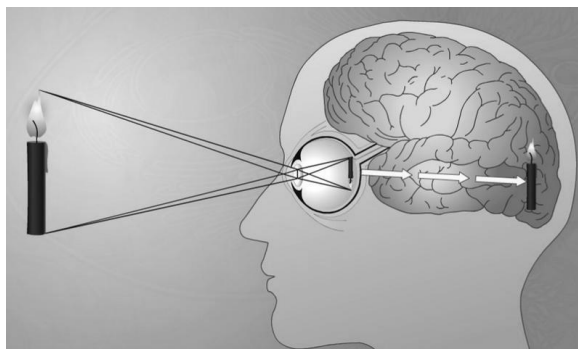


Рис 1.1.1 Робота людського зору [1]

Нейронна мережа схожа з мозком з декількох точок зору:

- 1) Знання надходять у нейронну мережу з навколишнього середовища та використовуються в процесі навчання.
- 2) Для накопичення знань застосовуються зв'язки між нейронами, звані синаптичними вагами [2]

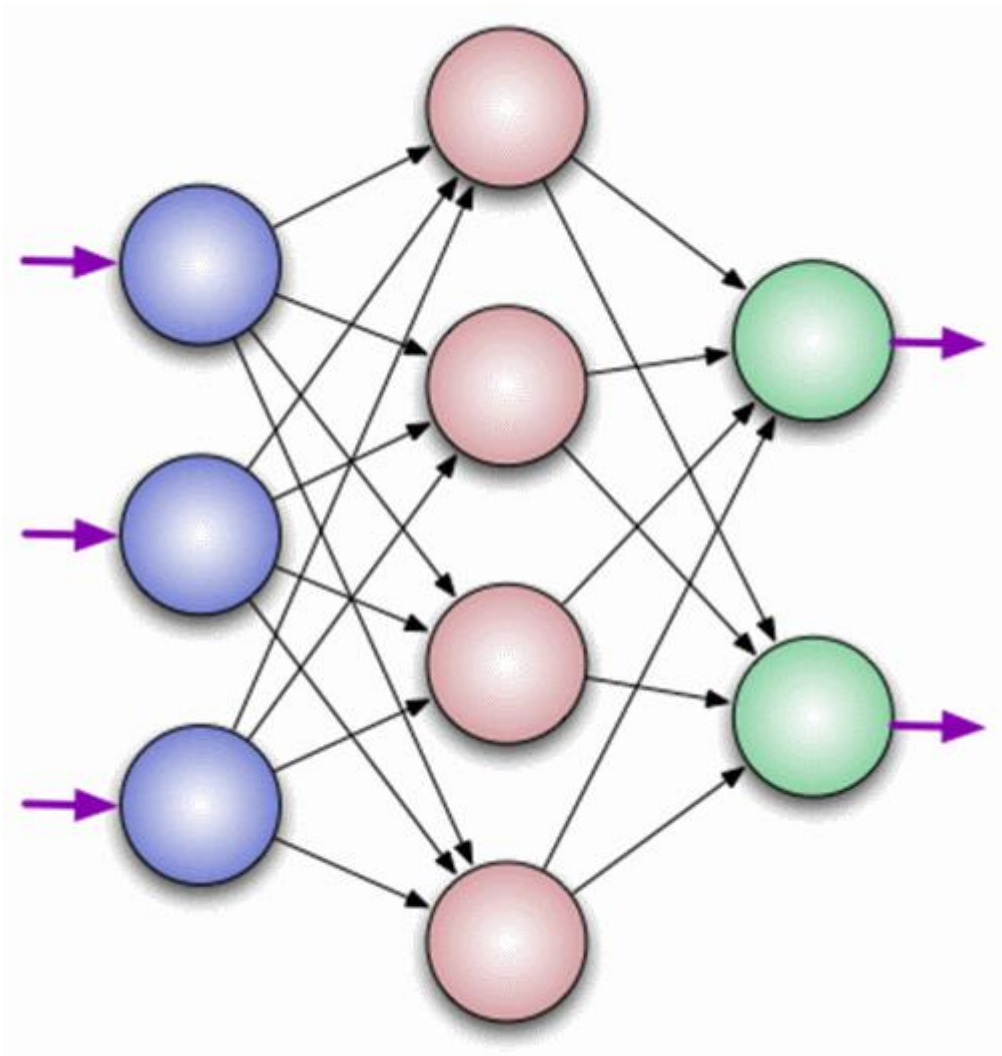


Рис 1.1.2 Схема нейронів [3]

Нейрон - це обчислювальна одиниця, яка отримує інформацію, здійснює над нею прості обчислення і передає її далі. Вони поділяються на три основні типи: вхідний (синій), прихований (червоний) та вихідний (зелений). У разі, коли нейронна мережа складається з великої кількості нейронів, вводять термін

шару. Відповідно є вхідний шар, який отримує інформацію, n прихованих шарів), які її обробляють і вихідний шар, який виводить результат. У кожного з нейронів є 2 основні параметри: вхідні дані (input data) та вихідні дані (output data). Що стосується вхідного нейрона: $\text{input} = \text{output}$. В інших, в поле input потрапляє сумарна інформація всіх нейронів з попереднього шару, після чого вона нормалізується за допомогою функції активації (поки що просто представимо її $f(x)$) і потрапляє в поле output

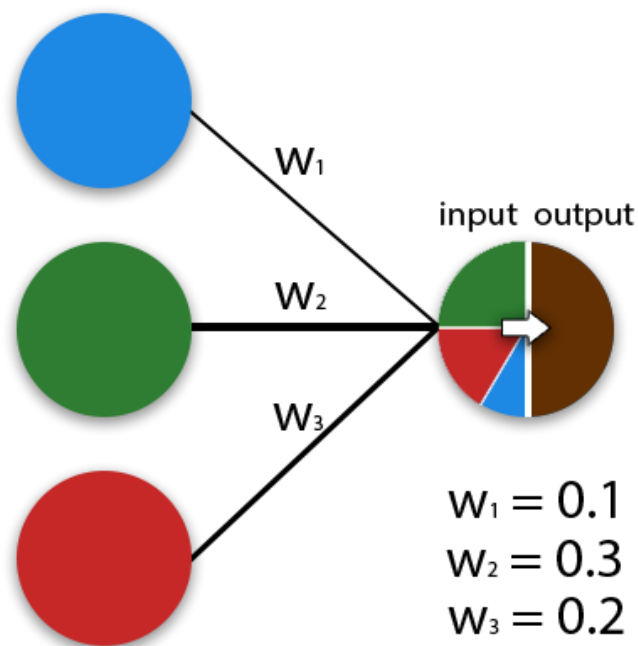


Рис 1.1.3 Схема роботи синапсу [4]

Синапс це зв'язок між двома нейронами. У синапсів є один параметр - вага. Завдяки йому вхідна інформація змінюється, коли передається від одного нейрона до іншого. Припустимо, є 3 нейрони, які передають інформацію наступному. Тоді у нас є 3 ваги, які відповідають кожному з цих нейронів. У того нейрона, у якого вага буде більшою, та інформація і буде домінуючою в наступному нейроні (приклад - змішання кольорів). Насправді, сукупність терезів нейронної мережі або матриця терезів - це своєрідний мозок всієї системи. Саме завдяки цим вагам, вхідна інформація обробляється та перетворюється на результат.

1.2 Історія нейронних мереж

Таке поняття як NN з'явилося у середині XX століття. У 1943 році два математика Мак-Каллок Уоррен та Уолтер Пітс, працювали над розробкою першої комп'ютерної моделі нейронної мережі на основі математичних алгоритмів, що надало революційне уявлення про мозок як про комп'ютер. В їх теорії вони зробили припущення щодо уявлення нейронів як пристроїв, які взаємодіють з двійковими числами – це дало життя моделі, яку назвали порогова логіка. Як і людський мозок, прототип Мак-Каллока-Пітса, був здатний навчатися, розпізнавати образи та виконувати логічні чи числові операції. Завдяки цій моделі було підвищено інтерес до вивчення біологічних процесів у людському мозку та застосування нейронних мереж як методу штучного інтелекту.

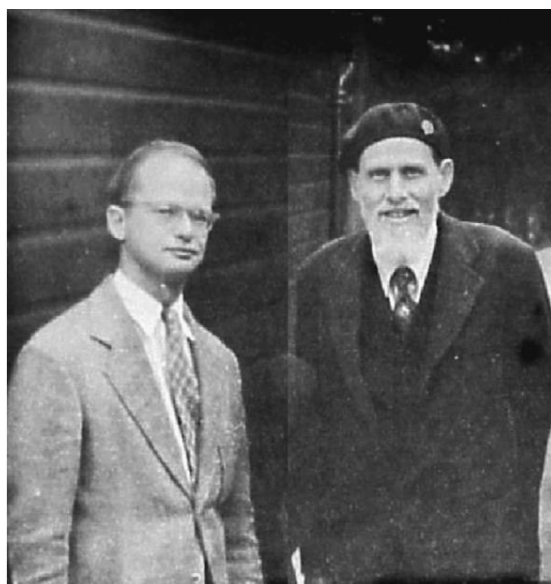


Рис 1.1.1 Мак-Каллок Уоррен та Уолтер Пітс [5]

Після праці Каллока та Пітса, у 1949 році канадський нейробіолог Дональд Гебб висунув припущення, що машинне навчання насамперед полягає у змінах ваги синаптичних зв'язків. За його теорією, система

повинна навчатися спонтанно, без втручання. Пізніше теорія Гебба лягла в основу явища довгострокової потенціації.

У 1957 році Розенблатом були розроблені математична та комп'ютерна моделі сприйняття інформації мозком на основі двошарової нейронної мережі, що навчається. Під час навчання дана мережа використовувала арифметичні дії додавання та віднімання. Розенблатт описав таку схему як перцептрон. У 1958 році їм була запропонована модель електронного пристрою, який повинен був імітувати процеси людського мислення, а через два роки була продемонстрована перша діюча машина, яка могла навчитися розпізнавати деякі з літер, написаних на картках, які підносили до її "очей", що нагадує кінокамери.[6]

У 1969 році вийшла книга Мінського та Пейперта, в якій увага була акцентована на обмеженості простих перспертонів. Він вивів наступні основні вади:

- Алгоритми навчання таких систем еквівалентні знаходженню лінійної розділяючої функції, тобто при відсутності лінійного розділення алгоритм не знайде коефіцієнт лінійної розділяючої функції
- Помилки в експериментах з розпізнаванням
- Відсутність додавання за модулем XOR (Виключає АБО)
- Малі обчислювальні потужності комп'ютерів, що робить неефективним обчислення великої кількості даних

Після руйнівної книги Мінського-Пейперта дослідження нейронних мереж уповільнилися до того моменту, поки комп'ютери не досягли високих обчислювальних потужностей. Одним із важливих кроків, що стимулювали подальші дослідження, стала розробка в 1975 Вербосом методу зворотнього поширення помилки, який дозволив ефективно вирішувати завдання із

навчанням багатошарових мереж і вирішити проблему зі "складанням по модулю 2".

Алгоритм паралельної розподіленої обробки даних у середині 1980-х років став популярним під назвою «конективізм». У 1986 році в роботі Руммельхарта та Мак-Клелланда конективізм був використаний для комп'ютерного моделювання нейронних процесів.

1.3 Принципи роботи нейронних мереж для обробки зображення

Комп'ютерний зір зосереджується на тому, щоб комп'ютери «бачили» світ так, як це роблять люди, тобто змушує їх розуміти зображення. У наш час з'являться все більше і більше технологій комп'ютерного зору: QR-сканери, розпізнавання обличчя, пошук хвороби по рентгенівські знімкам, та інше.

Один із найрозповсюдженіших прикладів комп'ютерного зору — це розпізнавання обличчя. Біометрія використовується для оплати в кафе, реєстрації у ДІІ, підтвердження банківських операцій.

Працює така система за таким алгоритмом:

1. Програма аналізує зображення, що надходить із камер, на наявність осіб у кадрі.
2. Коли особа виявлена, запускається алгоритм її відстеження у відеопотоці — програма визначає, на якому з кадрів відео обличчя відображено у найкращій якості та ракурсі.

3. У цьому кадрі за допомогою алгоритму програма знаходить ключові точки особи, які дозволяють визначити його орієнтацію на зображенні. Перетворений до потрібного стандартного положення портрет обличчя, надсилається на сервіс розпізнавання.
4. Зі стандартного формату JPEG він перетворюється на дескриптор – набір незмінних параметрів обличчя, який використовується для порівняння знімка з іншим зображенням. Далі програма порівнює два дескриптори і видає відповідь, чи внесена в базу людина, яка потрапила в кадр.

В основі сучасних систем комп'ютерного зору лежать алгоритми машинного навчання, як правило, це нейронні мережі. Для їхнього навчання потрібна велика кількість зображень з мітками того, які об'єкти на них знаходяться. У процесі навчання мережа сама визначає елементи, які шукатиме в інших зображеннях, щоб розпізнати їх із мінімальною кількістю помилок.[7]

	Granny Smith	85.6%
	iPod	0.4%
	library	0.0%
	pizza	0.0%
	toaster	0.0%
	dough	0.1%
	Granny Smith	0.1%
	iPod	99.7%
	library	0.0%
	pizza	0.0%
	toaster	0.0%
	dough	0.0%

Рис.1.2.1 Приклад похибки машинного зору OpenAI [8]

1.4 Сучасні нейронні мережі для обробки зображення

Щоб зрозуміти актуальність нейронних мереж для обробки зображень або комп'ютерного зору варто розглянути найпопулярніші продукти.

DALL-E и DALL-E 2

Нейронна мережа від некомерційного проекту OpenAI – на цей час одна із найпотужніших програм по генерації зображень, має широкий інструментарій по текстовій генерації, редагуванню зображення, створювати детальні картинки. Попри широкий спектр можливостей цієї нейронної мережі, робота з картинками, які завантажені в поганій якості, дає не дуже задовільний результат.

RawNeRF

Нова розробка працівників Google, яка була представлена під час конференції CVPR 2022. Вони показали технологію в роботі на прикладі кількох знімків, зроблених увечері та вночі. Нейронна мережа націлена на те, щоб прибирати шуми з фотографій. Поки є лише прототипом



Рис.1.3.1 Приклад роботи машинного RawNeRF [9]

1.5 Технологія для обробки відеопотоку та зображень

Для роботи з відеопотоком або зображенням, яке буде надходити до нейронної мережі, в основному використовується бібліотека OpenCV.

OpenCV – це бібліотека алгоритмів комп'ютерного зору, обробки зображень та чисельних алгоритмів загального призначення з відкритим кодом, яка може вільно використовуватися в академічних та комерційних цілях – поширюється за умов ліцензії BSD.

Плюсами даної бібліотеки є проста робота зі зчитуванням зображенням, наявність на всіх популярних мовах програмування, можливість накладання фільтрів, та попіксельне перетворення зображення.

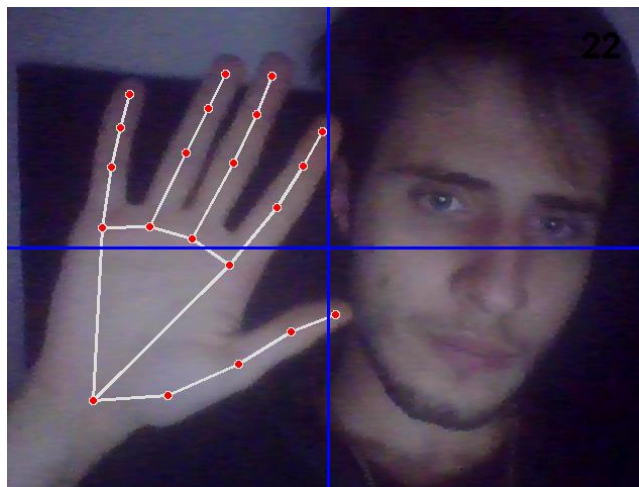


Рис.1.4.1 Приклад роботи захвату руху написаний у програмі OpenCV

1.6 Технології зменшення шумів зображення

Найголовніша проблема комп'ютерного зору, це праця з шумними зображеннями, які дають велику похибку, що приводить до погіршення результату. Для уникнення таких проблем є наступні способи зменшення шуму

Перехід у координати яскравість-колір.

Найпростіший з представлених фільтрів може бути здійснюваний багатьма способами (HSV), $L * a * b$. Так як шум у колірній компоненті, навпаки, набагато вищий, ніж у яскравості проста фільтрація колірної компоненти та зворотне відновлення, як правило, роблять картинку значно кращою.

Медіанний фільтр.

Один із видів фільтрів, що широко використовується в цифровій обробці сигналів та зображень для зменшення рівня шуму. Спочатку усі значення пікселів країв сортуються у певному порядку (зростання) та вибирається медіанне значення, яким замінюється центральний піксель. Медіанна фільтрація - ефективна процедура обробки сигналів, схильних до впливу імпульсних перешкод.

Фільтри, що використовують спектральне подання сигналу

За таким принципом, працює Photoshop. Суть цього методу полягає в тому, щоб зробити навколо кожного пікселя перетворення Фур'є, потім стерти високі частоти і зробити зворотне перетворення.

Фільтри, що виділяють домінантний напрямок

Ці фільтри в кожній точці спочатку знаходять домінантний напрямок (напрямок градієнта яскравості), а потім усереднюють сигнал лише в перпендикулярному напрямку. Таким чином, лінії та дрібні деталі залишаються чіткими.

1.7 Висновки до першого розділу

1. В результаті інтернет-досліджень було виявлено два основні класи нейронних мереж для обробки зображення:

- Конволюційна нейронна мережа (CNN)
- Повторна нейронна мережа (RNN)

Вони відрізняються типом навчання та шарової структури.

2. Зробивши огляд на популярні нейронні мережі можна виявити 3 основні перешкоди у розробці:

- Швидкодійність
- Можливість праці з шумними зображеннями
- Можливість надавати результат без похибок

Метою даної роботи буде розробка NN яка зможе уникнути даних перешкод

РОЗДІЛ 2. ОБҐРУНТУВАННЯ ФУНКЦІОНАЛЬНОГО ПІДХОДУ ТА ПРИНЦИП РОБОТИ НЕЙРОННОЇ МЕРЕЖІ

2.1 Нейронні мережі

Нейронна мережа є групою алгоритмів, які намагаються розпізнати основні набори відносин в даних за допомогою процесу, що схожий з роботою людського мозку. Нейронні мережі відносяться до певних систем нейронів (органічних або штучних). Нейрон представляє собою, певну математичну функцію яка виконує збір та класифікацію даних (відповідно до закладеної архітектури).

Особливістю нейронні мережі є можливість адаптуватись до зміни вхідних даних, це дозволяє нам отримати найкращий можливий результат без потреби переробляти алгоритм виводу. Концепція нейронних мереж, які використовують штучний інтелект та комп'ютерний зір, стрімко набирає популярність в поточні дні.

Штучна нейрона мережа складається з певної кількості з'єднаних вузлів(штучних нейронів). Дана система схожа з будовою біологічних нейронів у головному мозку тварин). Кожне таке з'єднання штучних нейронів може передавати сигнал від одного до іншого. Штучний нейрон який одержує сигнал може спершу обробляти його а потім сигналізувати штучним нейронам які з'єднані з ним.

Нейронна мережа містить шари взаємозалежних вузлів(перцептрон). Кожен такий вузол передає сигнал, отриманий в результаті множинної лінійної регресії, у функцію активації, яка може бути нелінійною

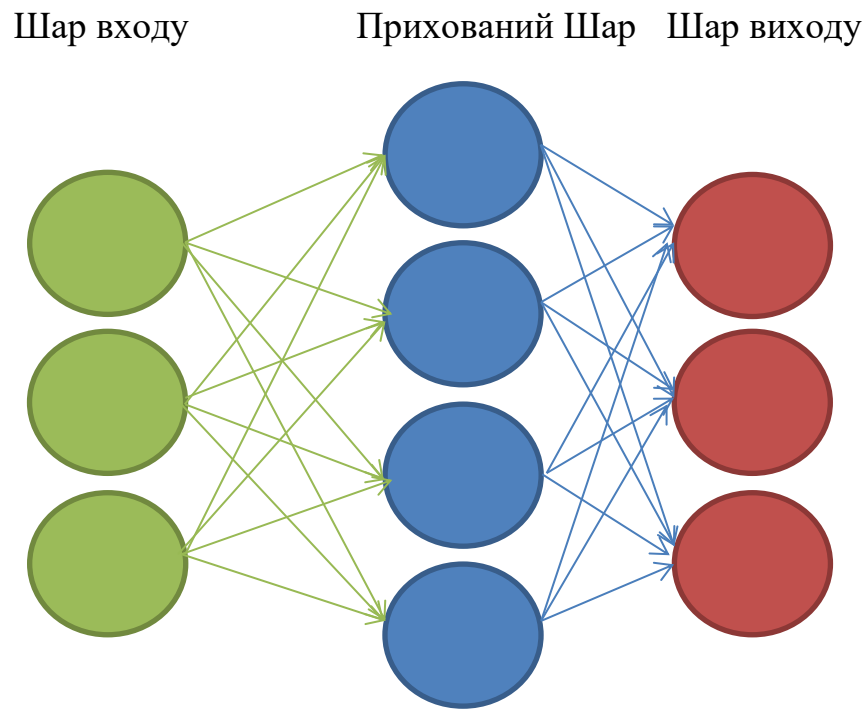


Рис 2.1 схема нейронної мережі

В поширених реалізаціях штучних нейронних мереж сигнал на з'єднанні між штучними нейронами є дійсним числом, а вихід кожного обчислюється нелінійною функцією (сумою його входів). Штучні нейрони та їх з'єднання мають вагу, яка корегується в процесі навчання. Вага може збільшувати або зменшувати силу сигналу на з'єднанні. Штучні нейрони можуть мати такий поріг, при якому сигнал відправляється лише тоді, якщо сукупний сигнал перетинає цей ваговий поріг. Штучні нейрони зазвичай складаються з себе шари. Кожен такий шар може виконувати різні види перетворень своїх входів. Сигнали проходять від першого до останнього шару.

Первинною метою підходу ШНМ було симуляція розв'язання задач як це робив би людський мозок. Штучні нейронні мережі використовували в ряді різноманітних задач, включно з машинним перекладом, розпізнаванням мовлення, комп'ютерним баченням, соціально-мережовим фільтруванням.

2.2 Типи нейронних мереж

Є багато різних типів штучних нейронних мереж, що відрізняються за складністю. У них основна задача – зімітувати функцію людського мозку для вирішення складних проблем або завдань. Архітектура кожного типу штучної нейронної мережі певною мірою являє собою нейрони і синапси. Однак вони відрізняються за складністю та варіантами використання.

Якщо відокремити їх в групи можна виділити п'ять основних груп:

- Штучні нейронні мережі з прямим зв'язком
- Багатошаровий перцептрон
- Радіальні базисні нейронні мережі
- Рекурентні нейронні мережі
- Модульні нейронні мережі

Далі для більш детального розуміння проведемо короткий огляд на дані нейронні мережі

Штучні нейронні мережі з прямим зв'язком

Нейронна мережа із прямим зв'язком (FNN) — це штучна нейронна мережа, у якій вузли не утворюють цикл[10]. Це головна відмінність від їхнього нащадка: рекурентних нейронних мереж.

Нейронна мережа з прямим зв'язком була першим і найпростішим типом розробленої штучної нейронної мережі [11]. Інформація, яка надходить в нашу NN, рухається тільки в одному напрямку (вперед). Від вхідних через приховані та до вихідних вузлів.

Найпростіший нейронна мережа – це одношарова мережа перцептрона, тобто складається з одного шару вихідних вузлів. В даній мережі входи подаються безпосередньо на виходи через серію ваги. Вага і вхідні дані підсумовуються та обчислюється в кожному вузлі. Якщо вони набувають

значення вище деякого порога (зазвичай 0), то нейрон спрацьовує та змінюється на активоване значення (зазвичай 1). В іншому випадку він стає неактивним, тобто його значення дорівнює негативному (зазвичай -1).

Хоча обчислювальна потужність однієї порогової одиниці досить обмежена, було показано, що мережі паралельних порогових одиниць можуть апроксимувати будь-яку безперервну функцію з компактного інтервалу дійсних чисел в інтервал $[-1,1]$. [12].

Багатошаровий перцептрон

Багатошаровий перцептрон є повнозв'язним класом штучних нейронних мереж з прямим зв'язком.

Багатошаровий перцептрон складається як мінімум із трьох шарів:

- вхідного шару
- прихованого шару
- вихідного шару.

За винятком вхідних вузлів, кожен вузол є нейроном, що використовує нелінійну функцію активації. Дана нейронна мережа використовує метод навчання з учителем, званий зворотним розповсюдженням.[13] Його багатошаровість та нелінійна активація відрізняють її від лінійного перцептрона. Він може розрізняти дані, які є лінійно роздільними.[14]

Навчання відбувається після обробки кожного фрагменту даних, що призводить до зміни ваг у перцептроні, яка обумовлена залежністю від кількості помилок у вихідних даних порівняно з очікуваним результатом. Це приклад навчання з учителем, яке здійснюється за допомогою зворотного розповсюдження, узагальнення алгоритму найменших середніх квадратів у лінійному перцептроні.

Радіальні базисні нейронні мережі

Радіальні базисні функції представляють собою спеціальний тип нейромереж із прямими зв'язками. Основне їхнє призначення - апроксимація багатовимірних функцій[15].

Їхню математичну основу складає теорія апроксимації багатовимірних функцій. RBF-мережі мають лише один шар прихованих нейронів.

Структура RBF-мереж містить один вхідний шар, один прихований шар нейронів, число яких зазвичай відповідає числу елементів навчальної послідовності, і один вихідний шар з одного (на рис. 1 представлений саме цей випадок) або декількох нейронів[16].

Т.о. за своєю побудовою RBF-мережі - це двошарові FF-мережі 1-го порядку, причому обидва шари з'єднані ваговою матрицею C .

Вхідний вектор X передається на нейрони прихованого шару. При цьому кожен нейрон прихованого шару отримує повну інформацію про вхідний вектор X .

Радіальні базисні функції h_i (1) виконують попередню обробку вхідних векторів, визначаючи їхню близькість (подібність) до центрів X_i . Вихідні значення $h_i(X)$ – це ступеня подібності між вхідним вектором X та окремими центрами X_i . На основі значень $h_i(X)$ визначається виважена сума.

Функція активації або виходу прихованих нейронів приймає великі значення лише у випадках, коли вхідний образ знаходиться поблизу опорної точки. Для входів поза областю, "покритою" образами навчальної послідовності, мережа формує лише невеликі значення на своїх виходах, причому для мереж із сигмоїдними функціями активації або виходу (наприклад, у багатошарових перцептронах) ці виходи можуть бути взагалі не передбачувані[17],

Проста структура RBF-мереж, що містять лише один шар прихованих нейронів, уможлиблює інший (не ітеративний) розрахунок ваги мережі. У

цьому їхня велика перевага в порівнянні з іншими типами нейромереж, які в більшості випадків використовують трудомісткі рекурентні алгоритми навчання[18]. Заслуговує на увагу також можливість ініціалізації RBF-мереж на основі прямого розрахунку ваг з подальшим їх донавчанням на основі алгоритмів навчання з заохоченням .

Рекурентні нейронні мережі

Повторювана нейронна мережа (RNN) — це клас штучних нейронних мереж, яка може будуватися з циклічними зв'язками між вузлами, що дозволяє виводу з деяких вузлів впливати на наступний вхід до тих самих вузлів. Це дозволяє йому демонструвати тимчасову динамічну поведінку. Похідні від нейронних мереж прямого зв'язку, RNN можуть використовувати свій внутрішній стан (пам'ять) для обробки послідовностей вхідних даних змінної довжини.[19][20] Це робить їх застосовними для таких завдань, як несегментоване, пов'язане розпізнавання рукописного тексту[21] або розпізнавання мовлення.[22][23] Рекурентні нейронні мережі теоретично повні за Тьюрінгом і можуть запускати довільні програми для обробки довільних послідовностей вхідних даних.[24]

Відмінність рекурентної нейронної мережі від згорткової нейронної мережі проявляється у імпульсній характеристиці, коли тільки перша є нескінченною. Обидва класи мереж демонструють часову динамічну поведінку.[25] Скінченна імпульсна рекурентна мережа — це орієнтований ациклічний граф, який можна розгорнути та замінити прямою нейронною мережею, тоді як нескінченна імпульсна рекурентна мережа — це орієнтований циклічний граф, який не можна розгорнути.

Як кінцеві імпульсні, так і нескінченні імпульсні рекурентні мережі можуть мати додаткові збережені стани, і зберігання може бути під прямим контролем нейронної мережі. Пам'ять також можна замінити іншою мережею чи графіком, якщо це включає часові затримки або має петлі

зворотного зв'язку. Такі контрольовані стани називаються стробованим станом або стробованою пам'яттю, і вони є частиною довготривалих мереж короткочасної пам'яті (LSTM) і стробованих рекурентних блоків. Це також називається нейронною мережею зворотного зв'язку (FNN).

Повністю рекурентні нейронні мережі (FRNN) з'єднують виходи всіх нейронів із входами всіх нейронів. Це найзагальніша топологія нейронної мережі, оскільки всі інші топології можна представити, встановивши деякі вагові коефіцієнти з'єднань на нуль, щоб імітувати відсутність зв'язків між цими нейронами. [26]

Модульна нейронна мережа

Модульна нейронна мережа — це штучна нейронна мережа, що характеризується серією незалежних нейронних мереж, модерованих деяким посередником. Можливо, незалежний нейронний набір служить модулем і працює з окремими вхідними даними для виконання певної підзадачі, яку мережа хоче виконати[27]. Вчитель нейронної мережі використовує вихідні дані кожного модуля, виконуючі певні процеси, обробляє їх для отримання вихідних даних мережі. Посередник приймає тільки вихідні дані модулів та не відповідає і не сигналізує модулям яким-небудь іншим способом.

За результатами дослідження штучних нейронних мереж цілком ймовірно, щоб штучні нейронні мережі продовжували опиратися на своє біологічне введення та емулювати сегментацію та модульність, виявлені в мозку. Мозок, наприклад, поділяє складну задачу зорового сприйняття на безліч підзадач.[28] У частині мозку, званої таламусом, розташоване латеральне колінчасте тіло (LGN), яке розділено на шари, які окремо обробляють колір і контраст: обидва основні компоненти зору

2.3 Задачі комп'ютерного бачення

Комп'ютерний зір, у його базисному розумінні, являє собою певну симуляцію біологічному зору. Біологічні моделі роботи таких систем

навчаються на базі зорового сприйняття людей і тварин. Системи комп'ютерного зору виконуються апаратно або програмно.

До підрозділів комп'ютерного зору входять: виявлення, стеження, розпізнавання та відновлення зображень.

Перш ніж перейти до обґрунтовуванню функціонального підходу, розглянемо принципи роботи та задачі CV. На початку NN отримує вхідне зображення за якимось із обраних способів (камера у звичайному або інфрочервоному спектрі, рентгенівські знімки , 3-D модель) . Інформація, яка знаходить до нашої нейронної мережі аналізується та виконує одну з поставлених задач.

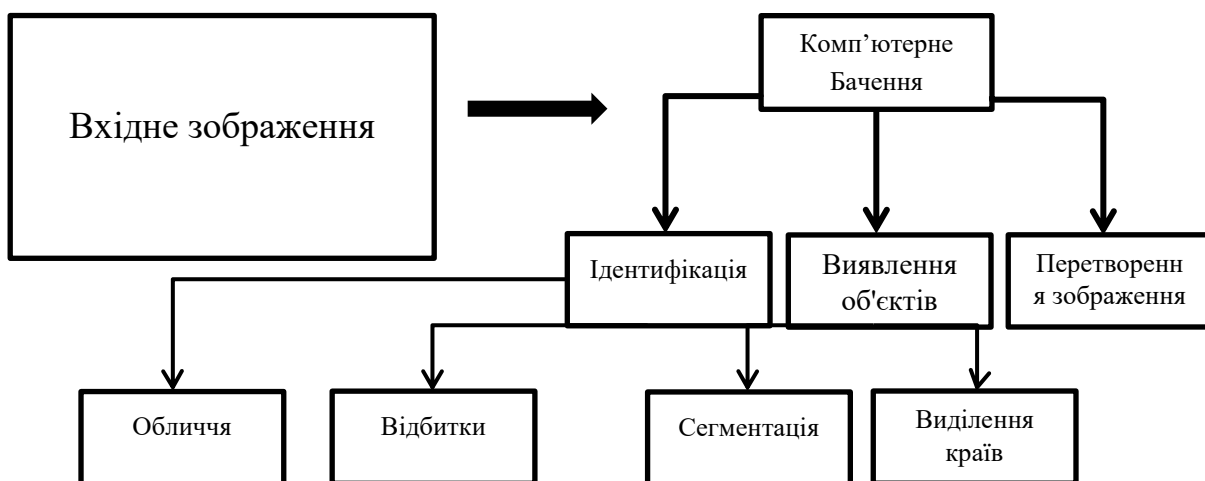


Рис. 2.1.1 Задачі комп'ютерного зору

Як ми бачимо з рис 2.1.1 , задачі комп'ютерного зору найрізноманітніші, проте вони всі зводяться до аналізу зображення або відеопотоку. Якщо нейронна мережа знаходить збіг між вхідним зображенням та даними з БД та відсоток цього збігу більш ніж заданий, вона виділяє фрагмент, що містить необхідну інформацію. Для виділення зазвичай

використовують або прямокутну область, яка обмежує вихідний фрагмент, або просто виділяють пікселі, що належать йому.

Головне розуміти, що під час процесу передачі вхідних даних до нашої системи, ми можемо застосувати дуже потужні механізми для спрощення роботи нашої NN. Зазвичай цими функціями є графічні фільтри (матриці, які використовуються для збільшення, або зменшення вхідної інформації, деякі з яких ми розглянемо в наступних розділах).

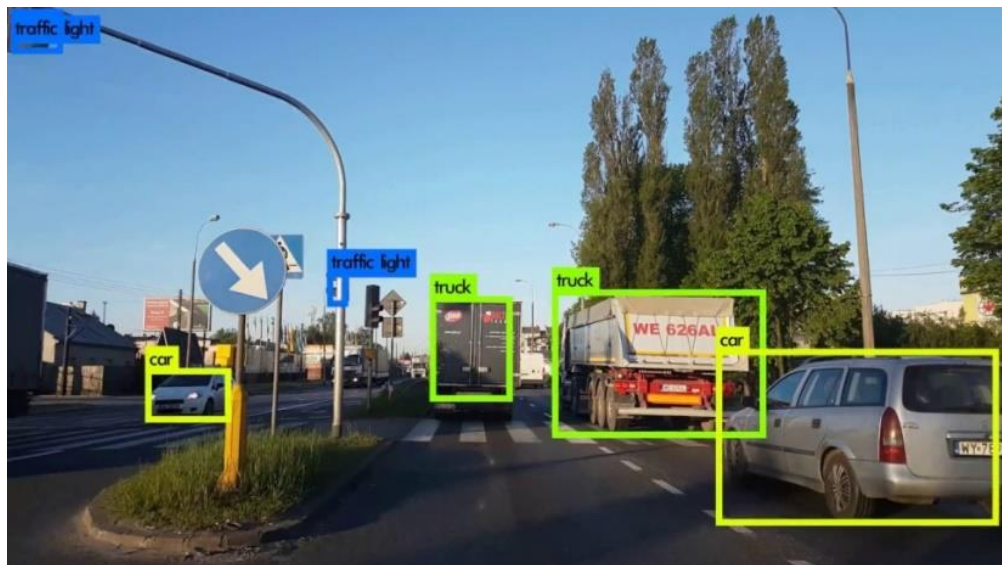


Рис. 2.1.2 Приклад комп'ютерного бачення[29]

2.4 Технології підвищення якості

Головною проблемою праці з CV є надходження до NW розмитих зображень. Такі вхідні дані дають велику похибку і роблять неможливим наступну працю з виявленням країв об'єктів. Однак зі збільшенням кількості інформації, завжди виникають труднощі. Частково вирішити цю проблему можна за допомогою додаткових фільтрів, які можуть видалити розмиття зображення.

Видалення розмиття зображення – це тип відновлення зображення, який зосереджується на відновленні чистих зображень шляхом усунення спотворень, які можуть з'явитись при трясінні камери або руху об'єкта. Артефакти розмиття є одним із видів спотворення.[30]. Дана технологія широко привернула увагу в галузі обробки зображень та комп'ютерного зору.

Важливість цієї технології обумовлена можливістю отримання максимуму інформації в умовах, коли отримання зображення без розмиття є дорогим або навіть неможливим.

Далі ми розглянемо та перевіримо найвідоміші функції та їх ефективність, використовуючи фільтри на фотографії, яка навмисне зроблена у нечитаній якості для більш видимого ефекту обробки зображення.

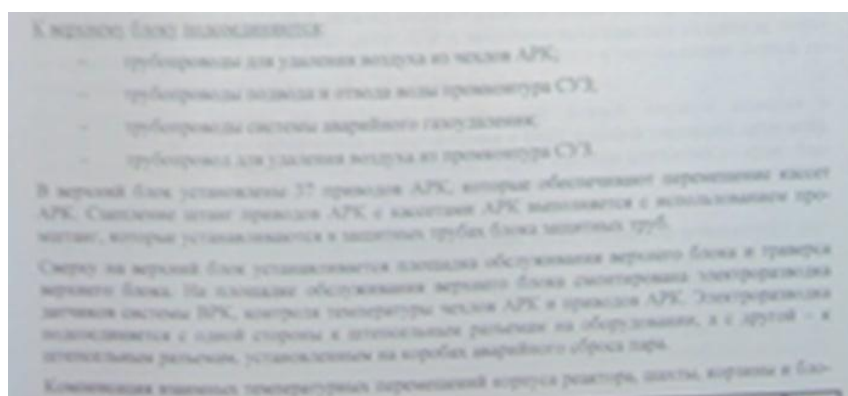


Рис 2.2.1 Зображення в поганій якості.

Фільтр різкості

Дана функція являє собою матрицю, де центральна комірка є позитивною, а її суміжні осередки є негативною сумою цього члену – 1(табл. 2.2.1) . Як ми бачимо на рис 2.2.2 , даний фільтр хоч і робить зображення

більш чітким, проте після його використання воно починає ставати більш гранульованим. Ми можемо і далі підвищувати центральний елемент, що трішки буде покращувати якість, проте велике збільшення призведе до дуже великого спотворення.

-1	-1	-1
-1	9	-1
-1	-1	-1

Рис 2.2.1 Матриця різкості

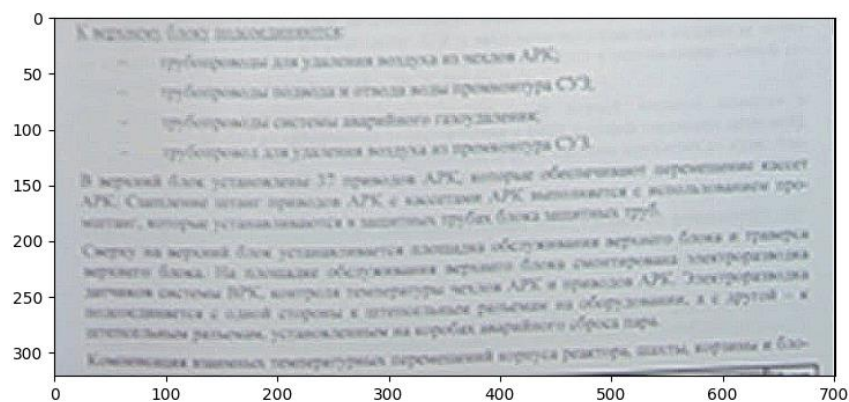


Рис 2.2.2 Зображення після фільтру різкості.

Фільтр нерізке маскуваня

Нерізке маскуваня — це техніка підвищення різкості зображення, яка вперше була застосована у фотозйомці в темних кімнатах, але зараз широко використовується в програмному забезпеченні для обробки цифрових зображень[31]. Його назва походить від того факту, що техніка використовує розмите, або «нерізке», негативне зображення для створення маски вихідного зображення. Потім нерізка маска поєднується з оригінальним позитивним зображенням, створюючи зображення, яке є менш розмитим, ніж оригінал. Отримане зображення, хоч і більш чітке, може бути менш точним відображенням об'єкта зображення.

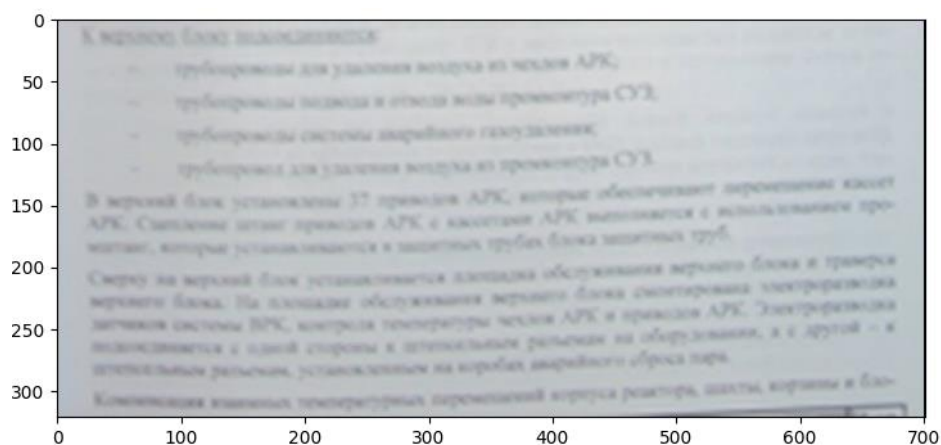


Рис 2.2.3 Зображення після фільтру маски нечіткості.

2.4 Технології зменшення шумів

Шум на зображеннях може виникнути, якщо вони були зроблені недорогими камерами, або під впливом зовнішніх збуджувачів. Простіше кажучи, метою шумозаглушення є відновлення «справжнього» сигналу (або його найкращого наближення). Всі такі методи засновані на розумінні та використовуючи відмінності між властивостями сигналу та шуму. Формально вирішення проблеми шумозаглушення спираються на три фундаментальні компоненти: модель сигналу, модель шуму і, нарешті, міра точності сигналу (широко відома як цільова функція), яка має бути зведена до мінімуму[32].

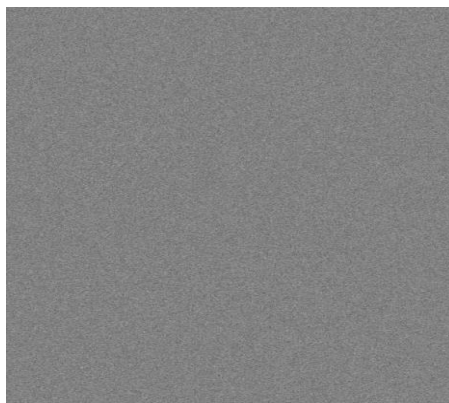


Рис 2.3.1 Гаусів шум

Якщо зображення було зроблено при поганій освітленості або у високій температурі це може приводити до появи Гаусового (сенсорного) шуму. Гаусів шум - це статичний шум, тобто його значення мають гаусів розподіл. Розглянемо фільтри, які можуть знизити дані дефекти на зображенні з великою кількістю сенсорного шуму. Розглянемо найпопулярніші фільтри, які зазвичай використовуються при роботі з нейронними мережами.



Рис 2.3.2 Зображення з Гаусовим шумом

Фільтр Гауса

Фільтр Гауса — це фільтр низьких частот, який використовується для зменшення шуму (високочастотних компонентів) і розмиття областей зображення[33]. Даний фільтр являє собою симетричне ядро непарного розміру, яке проходить по зображенню. Ядро не піддається різкій зміні країв зображення, завдяки більшій вазі в центрі ядра. Даний фільтр може бути розглянутий через наближення функції Гауса[34].

В двох вимірах дана функція являє собою:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad 2.3.1'[35]$$

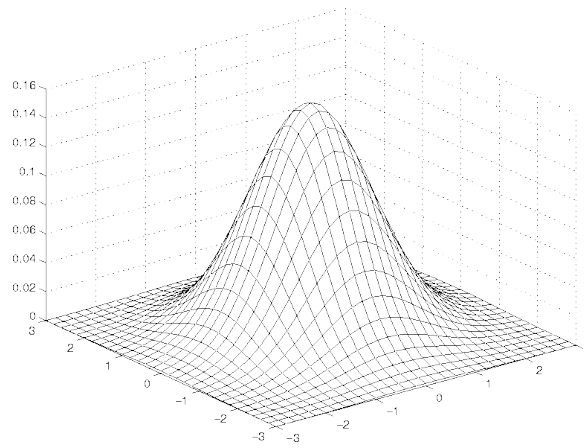


Рис 2.3.2 2D розподіл гаус[36]

За допомогою функції 2.3.1' ми можемо обчислити гаусове ядро будь-якого розміру:

$$\frac{1}{16}$$

1	2	1
2	4	2
1	2	1

Рис 2.3.3 3x3 двовимірна апроксимація ядра Гауса

Використаємо розраховане ядро Гауса на нашому зображенні (рис 2.3.1), щоб перевірити якість вихідного зображення.



Рис 2.3.4 Зображення після фільтру Гауса

Як ми бачимо (рис 2.3.4), після застосування стандартного фільтру Гауса з ядром 3x3, кількість шуму на зображенні дійсно стало меншою, проте зображення почало подаватись зернистості, це непомітно при невеликій кількості доданих шумів, проте при їх збільшені картинка буде спотворюватись сильніше.

Розмиття по медіані

Іншим популярним фільтром, який широко використовується для зменшення шумів, є медіанний фільтр.

Медіанний фільтр — це техніка нелінійної цифрової фільтрації, яка часто використовується для видалення шуму із зображення чи сигналу. Таке зменшення шуму є типовим етапом попередньої обробки для покращення результатів подальшої обробки[37].

Він бере значення інтенсивності пікселів у невеликій області в межах розміру фільтра, і обирає для центрального елементу даного проміжку середнє значення заданої інтенсивності (рис 2.3.4).

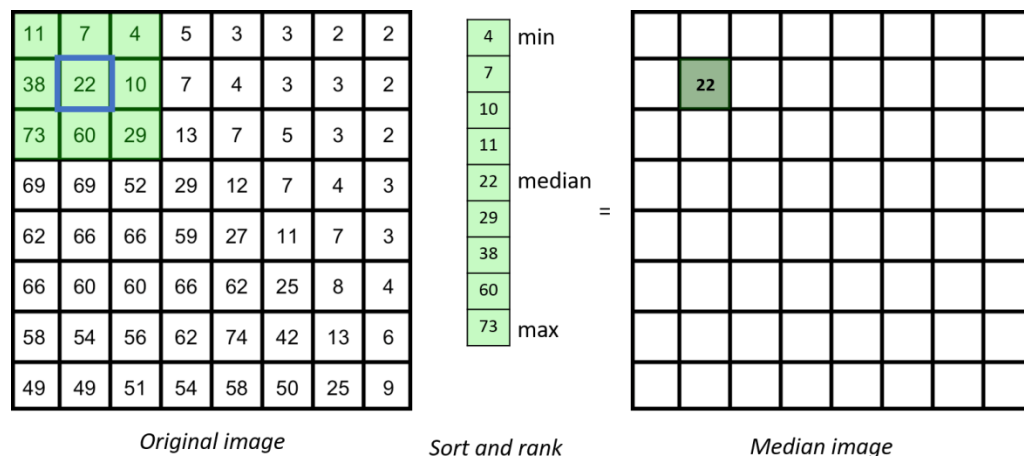


Рис 2.3.5 Робота фільтру по Медіані [38]



Рис 2.3.3 Зображення після фільтру по Медіані

Як ми бачимо, видалення шуму за допомогою медіанного фільтра не зменшує різницю в яскравості зображень, оскільки значення інтенсивності відфільтрованого зображення беруться зі вихідного зображення. Крім того, медіанний фільтр не зміщує края зображень, як це може статися з лінійним фільтром. В порівнянні з фільтром Гауса, він краще працює з зображеннями, на яких можуть бути високі спади і збільшення інтенсивності, проте при намаганні видалити з зображення велику кількість шумів він, як і фільтр Гауса, буде сильно зернити зображення.

Швидкий NI

Білатеральний фільтр – нелінійний фільтр, що виконує просторове усереднення в межах своєї маски, представлений Томазі і Мандуччі в 1998 році [39]. Він розширює поняття "згладжування Гауса", збільшуючи показники фільтра відповідно до відносної інтенсивності пікселів. Пікселі, які сильно відрізняються по інтенсивності від центрального пікселя, збільшуються в меншій мірі.

$$I^{filtered}(x) = \frac{1}{Wp} \sum_{x_i \in \cap} I(x_i) f_r(||I(x_i) - I(x)||) g_s(||x_i - x||) [40]$$

$$Wp = \sum_{x_i \in \cap} f_r(||I(x_i) - I(x)||) g_s(||x_i - x||) [41]$$

$I^{filtered}$ – вихідне зображення

x – координати поточного пікселя підлягаючому фільтрації

\cap – вікно з центром x , тому $x_i \in \cap$ це інший піксель

f_r – ядро діапазону для згладжування

g_s – просторове ядро для згладжування



Рис 2.3.4 Зображення після швидкої фільтрації

Як ми бачимо, після швидкої фільтрації шумів стало набагато менше, ніж при використанні медіанної фільтрації та фільтрації Гауса. Проте, незважаючи на сильне зменшення шумів, ми можемо спостерігати сильне спотворення дрібних деталей. Нажаль, пікселі зі схожою світловою гамою можуть бути затерті при використанні даного фільтру.

2.5 Технології виявлення країв об'єкту

При роботі з нейронними мережами треба розуміти, що комп'ютерний зір відрізняється від людського. Коли ми бачимо якийсь об'єкт, ми одразу можемо виявити його краї. Комп'ютерний зір шукає певний градієнт (межі з заданого об'єкта) , які іноді буває складно знайти на фотографіях без обробки. Для спрощення цього процесу використовуються розмиваючі фільтри для виділення контурів.

Під контуром зображення розумітимемо точки, між якими відбувається зміна значень яскравості.

Зміна яскравості характеризується висотою стрибка, кутом нахилу і координатою центру похилої ділянки. Перепад яскравості вважається контуром, якщо його висота та кут нахилу перевершують деякі порогові значення.

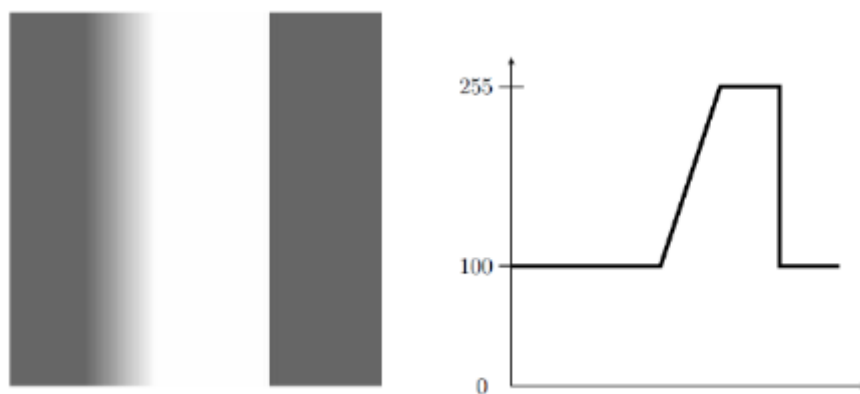


Рис.2.4.1 Перепад яскравості. Виявлення країв[42]

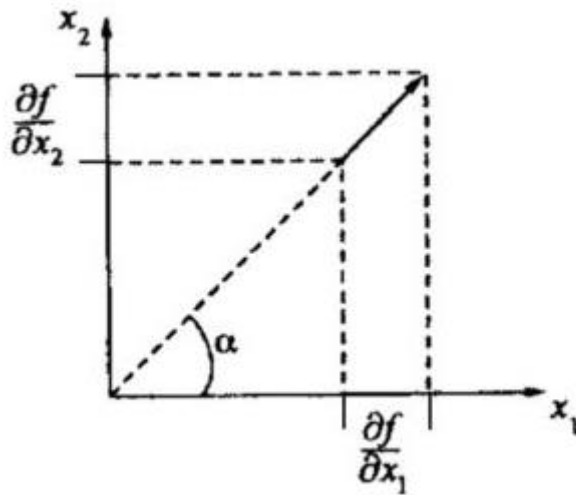


Рис.2.4.2 Графічне уявлення градієнту

Модуль градієнту $|Vf(x_1, x_2)|$ та кут його орієнтації α розраховується за формулою

$$|Vf(x_1, x_2)| = \sqrt{\left(\frac{\partial f}{\partial x_1}\right)^2 + \left(\frac{\partial f}{\partial x_2}\right)^2} \quad 2.4.1` [43]$$

$$\alpha = \arctg\left(\frac{\partial f}{\partial x_1} / \frac{\partial f}{\partial x_2}\right) \quad 2.4.2` [43]$$

Даний фільтр працює за наступним принципом:

- 1) Береться певний оператор (маски фільтру) коефіцієнтів (задається статично)
- 2) Маска проходиться по зображенню
- 3) В кожній точці зображення рахується відклик фільтру
- 4) Після отримання відклику фільтру вираховується градієнт

Зазвичай всі оператори мають схожі принципи. Центральні лінії представляють собою нульові елементи, а її краї позитивну і негативну суму, яка дорівнює 0. Далі ми розглянемо найвідоміші маски та їх використання на даному зображенні (рис. 2.4.3)



Рис 2.4.5 Зображення для виделення країв

Оператор Собеля

Одним із популярних способів виявлення градієнтних країв є оператори Собеля. Вони відрізняються напрямком проходження по зображенню. Існують наступні види: горизонтальний, вертикальний.

1	0	-1
2	0	-2
1	0	-1

а) Горизонтальний

1	2	1
0	0	0
-1	-2	-1

б) Вертикальний

Рис 2.4.4 Горизонтальний та вертикальний оператор Собеля

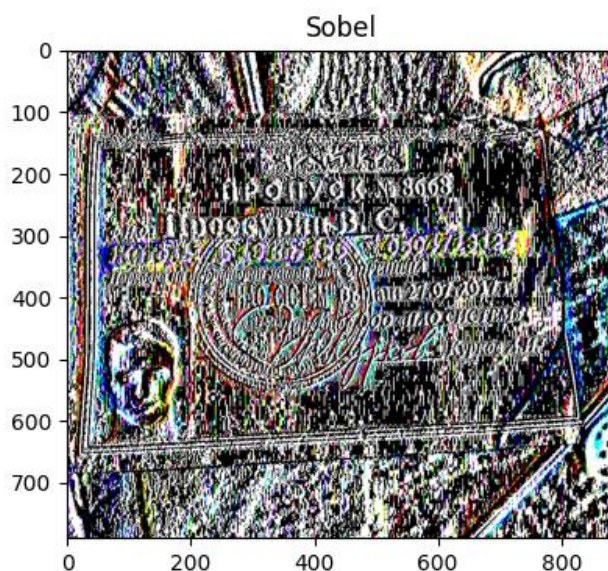


Рис 2.4.5 Зображення після фільтру Собеля

Як ми бачимо з зображення(рис. 2.4.5) після використанні маси Собеля шум замилився, а зони перепадів яскравості (максимальні зміни градієнту) дуже явно виділяються.

Оператор Щарра

Оператор Щарра – працює за тими ж принципами, що і оператор Собеля, за винятком того, що коефіцієнти ядра згортки відрізняються (рис.2.4.6). Це впливає на більш явне виділення центру, що робить його більш чутливий при витягуванні країв.

3	0	-3
10	0	-10
3	0	-3

Рис. 2.4.6 Оператор Щарру

Завдяки збільшенню коефіцієнтів ядра, збільшується кількість можливих країв, проте з їх збільшенням зростає і кількість можливого шуму. Як ми можемо бачити з фотографії (рис.2.4.5), краї зображення в порівнянні з оператором Собеля (рис. 2.4.5) стали більш явними, проте кількість шумів збільшується експоненціально. Дану проблему можна вирішити за допомогою попередньої фільтрації

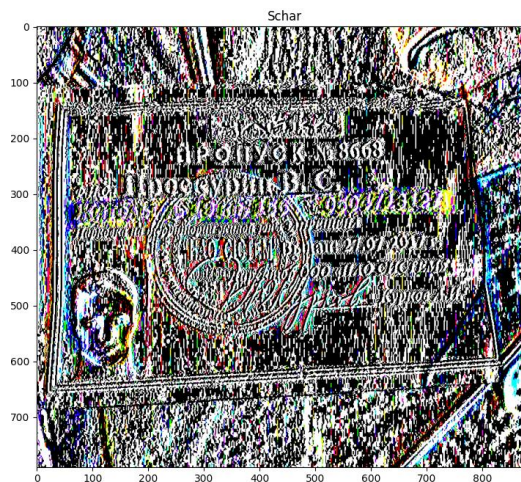


Рис 2.4.7 Зображення після фільтру Щарра

Метод Оцу

Метод Оцу на відміну операторів Щару та Собеля, не проходиться по зображенню з метою виявлення градієнтних частин. Оцу виявив закономірність, що на гістограмі зображення кількість точок однієї та другої групи яскравості розподілені локалізовано у двох областях. Тому для того, щоб фон відокремити від зображення, він запропонував статичний метод максимізації міжгрупової дисперсії, а фон занулити. Даний метод шукає центри цих груп і все, що відноситься до пікселів яскравості об'єкту робить еквівалентним одиниці. Все, що не є об'єктом, тобто має невелику яскравість, він робить еквівалентним нулю. Іншими словами він робить певну бінарізацію зображення.

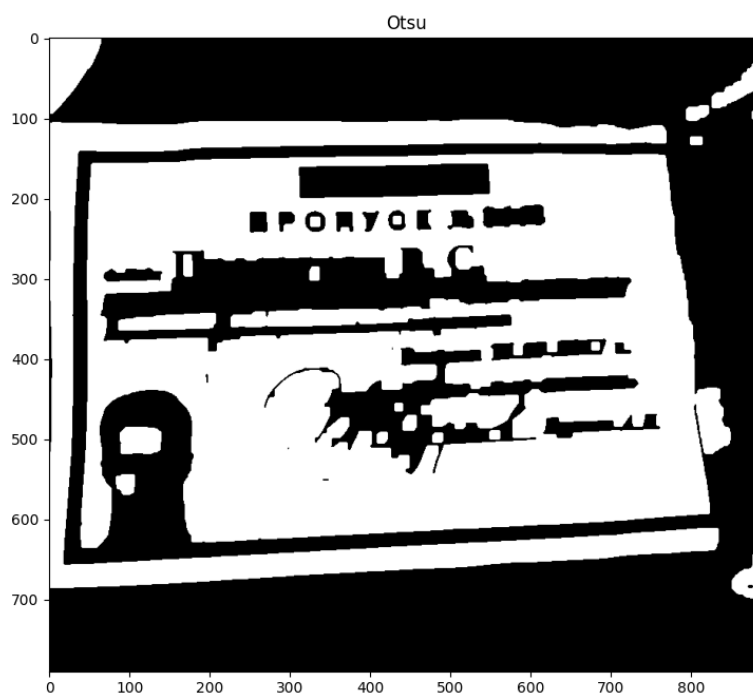


Рис 2.4.8 Зображення після фільтру Оцу

Як ми можемо спостерігати зі зображення (рис.2.4.8), даний фільтр замазав непотрібні нам деталі, зробивши нашу карточку більш видимою. Даний фільтр не є найкращим, іноді при його застосуванні краї можуть виділятися не повністю, тому, хоч і даний фільтр є гарним для застосування, він не вирішує всі наші поставлені задачі.

Метод Оцу на відміну операторів Щару та Собеля, не проходиться по зображенню.

2.6 Висновки до другого розділу

В розділі 2 було розглянуто різні типи нейронних мереж, були описані їх характеристики та сфери їх застосування. Було з'ясовано які задачі виконує комп'ютерне бачення та з якими проблемами може зіткнутися NN.

Було виявлено дві основні проблеми: шуми та розмиття зображення. Були розглянуті найпопулярніші методи боротьби з шумними та розмитими зображеннями, описані їх плюси та мінуси , наведені зображення після обробки даними фільтрами.

Також були розглянуті технології виявлення країв для написання функції розпізнавання обличчя.

Як результат, були обрані орієнтири, які ми повинні будемо покращити в наступному розділі.

РОЗДІЛ 3. АЛГОРИТМИ ТА ТЕХНІЧНІ ХАРАКТЕРИСТИКИ НЕЙРОННОЇ МЕРЕЖІ

3.1 Згорткова нейронна мережа

Згорткова нейронна мережа (CNN) — це алгоритм глибокого навчання, який сприймає вхідне зображення та призначає вагові значення різним типам об'єктів зображення. Дана нейронна мережа використовується для розпізнавання різних типів об'єктів, які знаходяться в нашому даних.

CNN реалізована як версія багатошарових перцептронів. Це означає, що кожен нейрон одного шару з'єднаний з усіма нейронами наступного шару. «Повна зв'язність» цих мереж робить їх схильними до переобладнання даних. На відміну від типових способів регуляризації при навчанні (зниження або з'єднання ваги нейронів) CNN використовують інший підхід. Вона збирає візерунки більшої складності завдяки об'єднанні простих візерунків.

Архітектура CNN аналогічна структурі підключення нейронів у людському мозку та нагадує організацію зорової кори. Окремі нейрони реагують на стимули лише в обмеженій області поля зору, відомій як рецептивне поле. Набір таких полів перекривається, щоб охопити всю візуальну область. Рецептивні поля різних нейронів частково перекриваються так, що охоплюють усе поле зору. Розглянемо архітектуру нашої нейронної мережі більш детально (рис. 3.1.1)

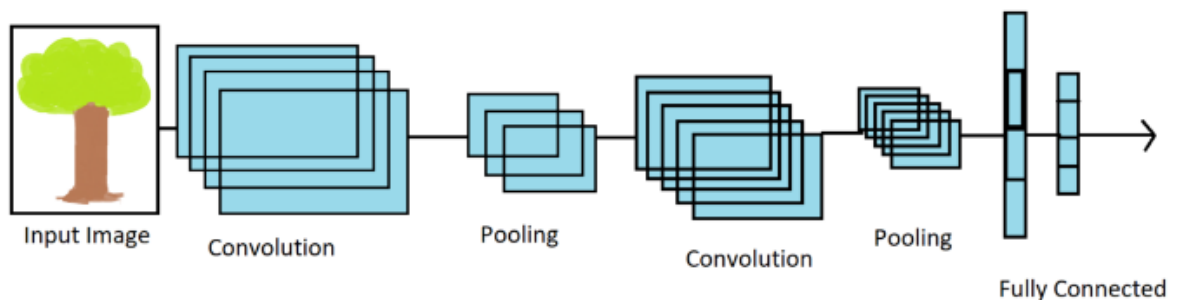


Рис. 3.1.1 Архітектура CNN [44]

Згорткова нейронна мережа складається з трьох основних рівнів (рис. 3.1.1):

- Вхідний рівень (зображення, яке надходить до нашої NN)
- Приховані шари, які складаються з певної кількості згорткових шарів (convolution) та об'єднуючих шарів (pooling)
- Вихідного рівня, тобто результуюче зображення (Fully Connected)

Згорткові шари

У CNN вхідний сигнал являє собою тензор форми який дорівнює множенню W, D, k

Де:

- W – матриця форми
- D – матриця входу

- k — кількість вхідних каналів

Після проходження через згортковий шар зображення абстрагується до карти функцій, яка також називається картою активації, з формою: (кількість входів) \times (висота карти характеристик) \times (ширина карти характеристик) \times (канали карти функцій).

Після обробки частини зображення згортковими шарами, вхідні дані передаються наступному шару (даний процес схожий за діяльністю із реакцією нейрона в корі головного мозку). Кожен згортковий нейрон обробляє дані тільки для свого рецептивного поля.

Об'єднання шарів

Згорткові мережі можуть включати локальні та глобальні рівні, які об'єднують разом із традиційними згортковими рівнями. Рівні об'єднання виконують функцію зменшення розмірів даних, це відбувається завдяки об'єднанню вихідних даних кластерів нейронів з одного рівня в наступний рівень. Локальний пул об'єднує невеликі кластери, зазвичай використовуються мозаїки розміром 2×2 . Глобальне об'єднання діє на всі нейрони карти функцій[45][46]. Існує два поширених типи об'єднання: максимальне та середнє. При максимальному об'єднанні використовується максимальне значення кожного локального кластера нейронів на карті функцій тоді як при об'єднанні середніх береться середнє значення.

Повністю зв'язані шари

Повністю зв'язані шари з'єднують кожен нейрон одного шару з кожним нейроном іншого шару. Працює за тим же принципом як і традиційна багат шарова нейронна мережа перцептронів.

Рецептивне поле

У нейронних мережах кожен нейрон отримує вхідні дані з певної кількості місць на попередньому рівні. У згортковому шарі кожен нейрон

отримує вхідні дані лише від обмеженої області попереднього шару, яка називається рецептивним полем нейрона. Зазвичай площа має квадрат (наприклад, 5 на 5 нейронів). У той час як у повністю зв'язаному шарі сприйнятливим полем є весь попередній шар. Таким чином, у кожному згортковому шарі кожен нейрон отримує вхідні дані з більшої області входу, ніж попередні шари. Це пов'язано з постійним застосуванням згортки, яка враховує значення пікселя, а також пікселів навколо нього. При використанні розширених шарів кількість пікселів у сприйнятливому полі залишається постійною, але поле заповнюється рідше, оскільки його розміри зростають при поєднанні ефекту кількох шарів.

Ваги

Для обчислювання вихідного значення, кожен нейрон нейронної мережі застосовує певну функцію до вхідних значень, отриманих із шару попереднього рівня. Вектор ваг – це функція, яка застосовується до вхідних значень. Навчання полягає в ітеративному коригуванні цих упереджень і ваг.

Фільтрами у CNN називають вектори ваг і зміщень, що представляють характеристики вхідних даних (наприклад, певну форму). Відмінною рисою згорткових нейронних мережах є те, що багато нейронів можуть використовувати один і той самий фільтр. Даний алгоритм зменшує обсяг пам'яті, оскільки для всіх рецептивних полів використовується єдиний зсув і один вектор ваг, які поділяють цей фільтр, на відміну від кожного рецептивного поля, що має власне упередження та векторне зважування.

3.2 Принцип роботи CNN для розпізнавання обличчя

Для більш детального розуміння CNN розглянемо її принцип роботи використовуючи бібліотеку *tensorflow* як основу даної нейронної мережі.

Перш ніж перейти до алгоритму та функціонального підходу, розіб'ємо логіку роботи NN на маленькі модулі та директорії (рис.3.2.1)

- dataset – директорія для зберігання шуканих зображень
- images – директорія для обробки вхідних зображень
- build_model – модуль для побудови нашої моделі
- train_model – модуль для навчання нашої моделі
- init – головний файл запуску CNN
- functions – директорія допоміжних функцій для підвищення якості та зменшення шуму

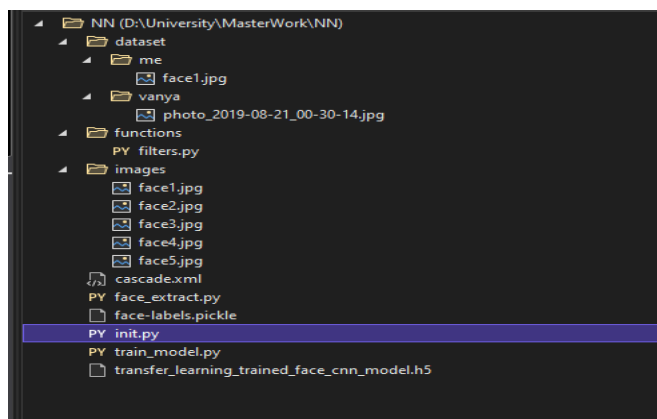


Рис.3.2.1 Директорії CNN

3.3 Підготовка шуканих зображень. Каскади Хаара

Перш ніж перейти до розпізнавання об'єктів ми повинні сформувати шукані зображення. Для підготовки таких зображень використовуються фільтри виокремлення країв. Як було розглянуто в п. 2.4 дані фільтри являють собою певні маски градієнтів, які проходять по зображенню та роблять краї більш видимими.

Одним з найпотужнішим інструментів задля виокремлення країв обличчя є Каскади Хаара.

Каскади Хаара являють собою деякі часні випадки методів Кені та операторів Собеля. Ознаки Хаара – це певна кількість суміжних прямокутників точок мінімумів і максимумів пікселів. Вони позиціонуються на зображенні, далі підсумовуються інтенсивності пікселів в областях, після чого обчислюється різниця між сумами. Ця різниця і буде значенням певної ознаки. Кожна ознака може показати наявність (або відсутність) будь-якої конкретної характеристики зображення, як зміна текстур, межі або країв між світлими та темними регіонами. Ознаки Хаара можуть бути як прямими так і нахиленими (рис. 3.3.1)

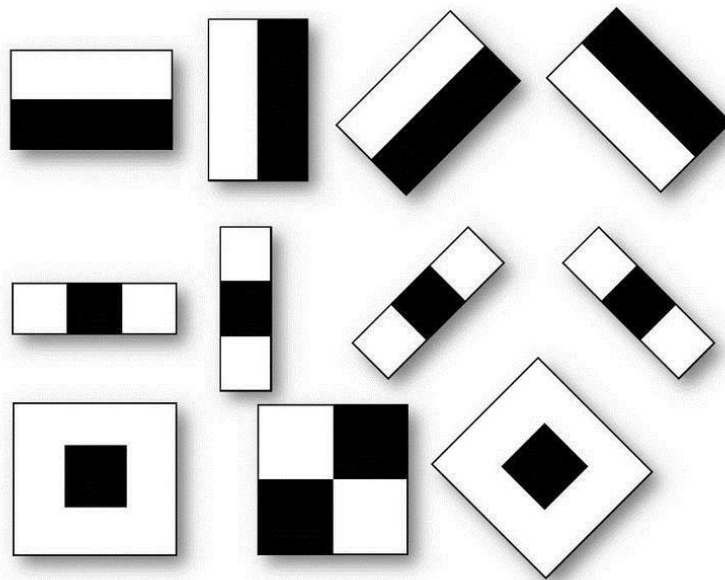


Рис. 3.3.1 Каскади Хаара

Знаходження обличчя за каскадами Хаара являє собою дуже потужну та швидку функцію (Класифікатори рис.3.3.2), фактично вирішуючих дерев, заснованих на розташуванні примітивів (ліній та точок) у зображенні, він працює за наступним алгоритмом:

- Зображення надходить до функції класифікатора
- Зі збільшеного зображення виділяється основоположний каскад

- Якщо цей каскад знаходиться, то функція починає дивитись на певні характерні ознаки, від загального виду нашого об'єкту до більш часних (адаптивний бустінг)
- Адаптивний бустінг проходиться по всім деревам та збирає інформацію
- Після збору інформації ми отримуємо певні краї шуканих нами зображень

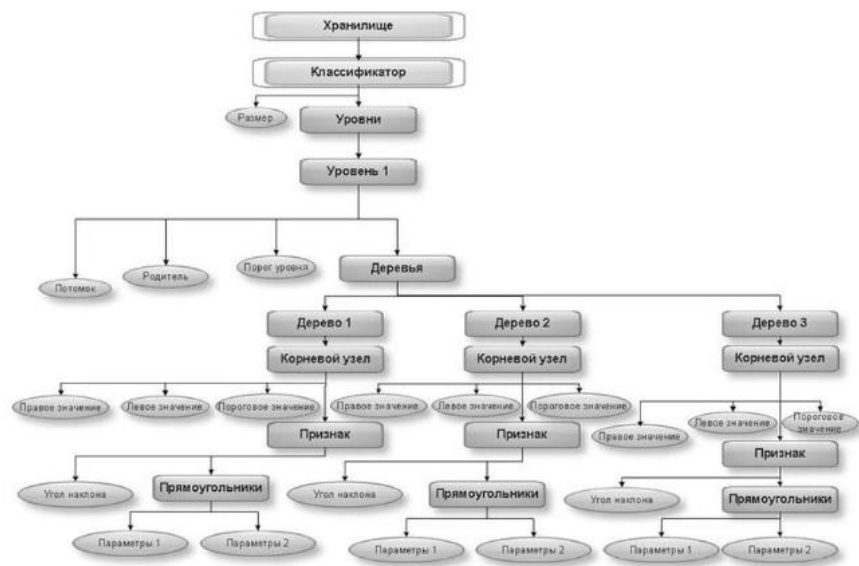


Рис. 3.3.2 Классификаторы Хаара

Даний метод в деяких випадках працює швидше ніж CNN, завдяки меншій кількості шарів згорток, які вже об'єднані в оптимальні каскади.

Розглянемо реалізацію нашого алгоритму підготовки шуканих облич (рис 3.3.3)

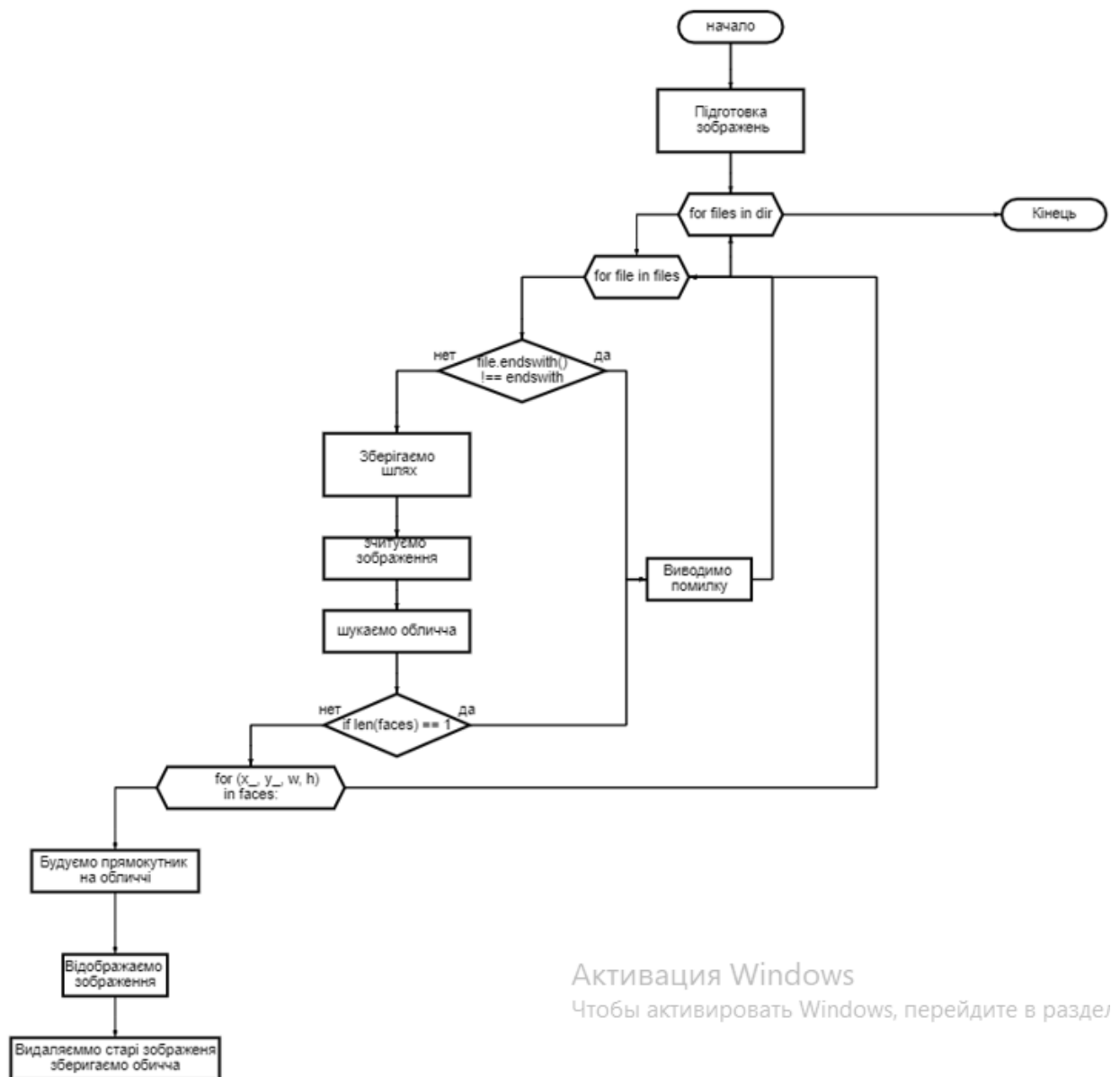


Рис. 3.3.3 Алгоритм збереження облич

Як ми бачимо з нашого алгоритму (рис 3.3.3) :

- Спочатку ми готуємо зображення, записуємо ім'я нашої директорії, зберігаємо каскади Хаара.
- Проходимо циклом по директорії з шуканими нами людьми.
- Далі в цих директоріях ми проходимо по всім зображенням, що знаходяться в них.

- Перевіряємо чи обраний нами файл відноситься до типу зображень; якщо ні, то виходимо з циклу та пишемо про помилку.
- Якщо файл існує, то зберігаємо шлях до зображення
- Зчитуємо дане зображення та формуємо з нього масив
- Шукаємо обличчя на даних зображеннях, формуємо з них масив
- Перевіряємо чи були знайдені зображення; якщо ні, то виходимо з циклу та пишемо про помилку.
- Далі проходимо циклом по знайденим обличчям, записуємо стартові пікселі по x, y та висоту(h) і ширину (w) нашого зображення
- За допомогою x, y, h , будуємо прямокутник, який відобразить знайдене обличчя
- Відображаємо зображення з прямокутником на знайденому обличчі
- Видаляємо старі зображення та зберігаємо нові зображення (обличчя, по яким наша модель буде шукати збіги)

3.4 Навчання моделі для нашої NN

Перш ніж наша нейронна мережа почне працювати, ми повинні створити модель (певний набір правил), по якій наша система буде давати відповідь чи існує на зображенні шукане обличчя.

Модель описує її архітектуру та конфігурацію, використовувані алгоритми навчання нашої нейронної мережі.

Конфігурація конкретизує структуру мережі у межах заданої архітектури: число нейронів, число входів і виходів мережі, використовувані активаційні функції.

Розглянемо алгоритм навчання нашої моделі (рис 3.4.1)

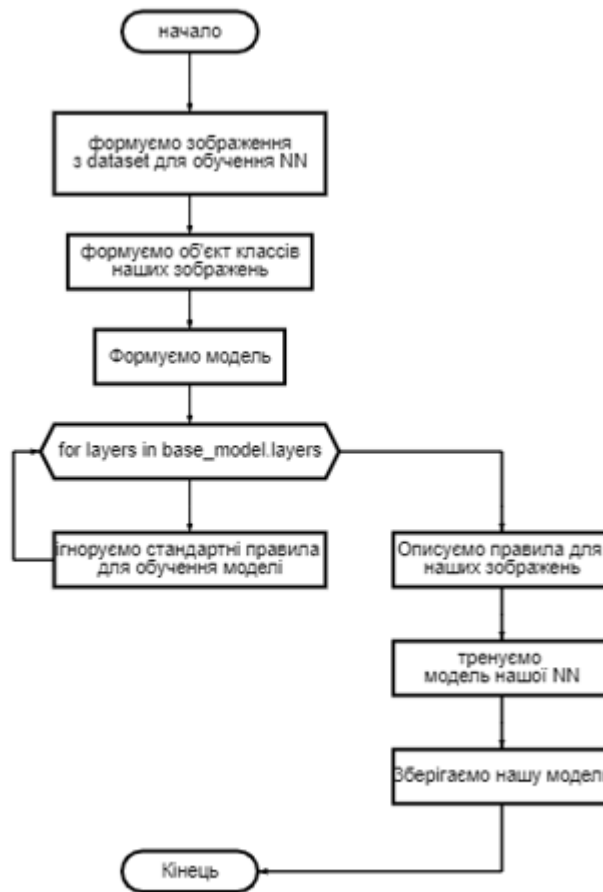


Рис. 3.4.1 Алгоритм навчання нашої NN

Розберемо як працює наша блок-схема :

- Спочатку ми формуємо алгоритм навчання нашої нейронної мережі з зображень, які знаходяться в “./dataset”; вказуємо в цій функції розмір та кольор зображень, які будуть зберігатися, розмір групи, тип формування та інше
- Після формування функції, заповнюємо об’єкт класів, який пізніше передамо в модель
- Формуємо модель за типом «vgg16», виключаємо з неї верхні слої за ненадійністю та передаємо розмір наших навчаних зображень
- Далі проходимося по стандартним правилам навчання та ігноруємо їх, щоб прискорити навчання моделі

- Додаємо нові правила з нашими зображеннями
- Тренуємо нашу модель
- Зберігаємо модель, щоб не витратити кожен раз час на її навчання.

3.5 Зменшення шумів та ефекту розмиття

Перш ніж перейти до запропонованих рішень для вирішення проблем пов'язаних із дефектами зображення, розглянемо реалізацію алгоритму, який зможе аналізувати зображення на наявність шумів.

Зменшення шуму. Деконволюція Вінеру

Даний фільтр являє собою потужний інструмент для вирішення проблем шуму, властивих деконволюції. Він працює в частотній області, намагаючись мінімізувати їх вплив. Метод деконволюції Вінера може широко використовуватися в програмах деконволюції зображень, оскільки частотний спектр більшості візуальних зображень можна досить легко оцінити.

$$G(u, v) = \frac{H^*(u, v)S(u, v)}{|H(u, v)|^2S(u, v) + N(u, v)}$$

- $G(u, v)$ – Перетворення Фур'є спотвореного зображення та оператору шуму
- $H(u, v)$ – спотворююча функція
- $H^*(u, v)$ – комплексне спряження спотворюючої функції
- $S(u, v)$ – енергетичний спектр оригінального зображення
- $N(u, v)$ – енергетичний спектр шуму

Даний фільтр можна використовувати як ефективний метод боротьби проти гаусового шуму, а з додаванням до формули PSF буде здатний

зменшувати розмитість вхідного зображення. Розглянемо і розберемо як працюють алгоритми до цих функцій

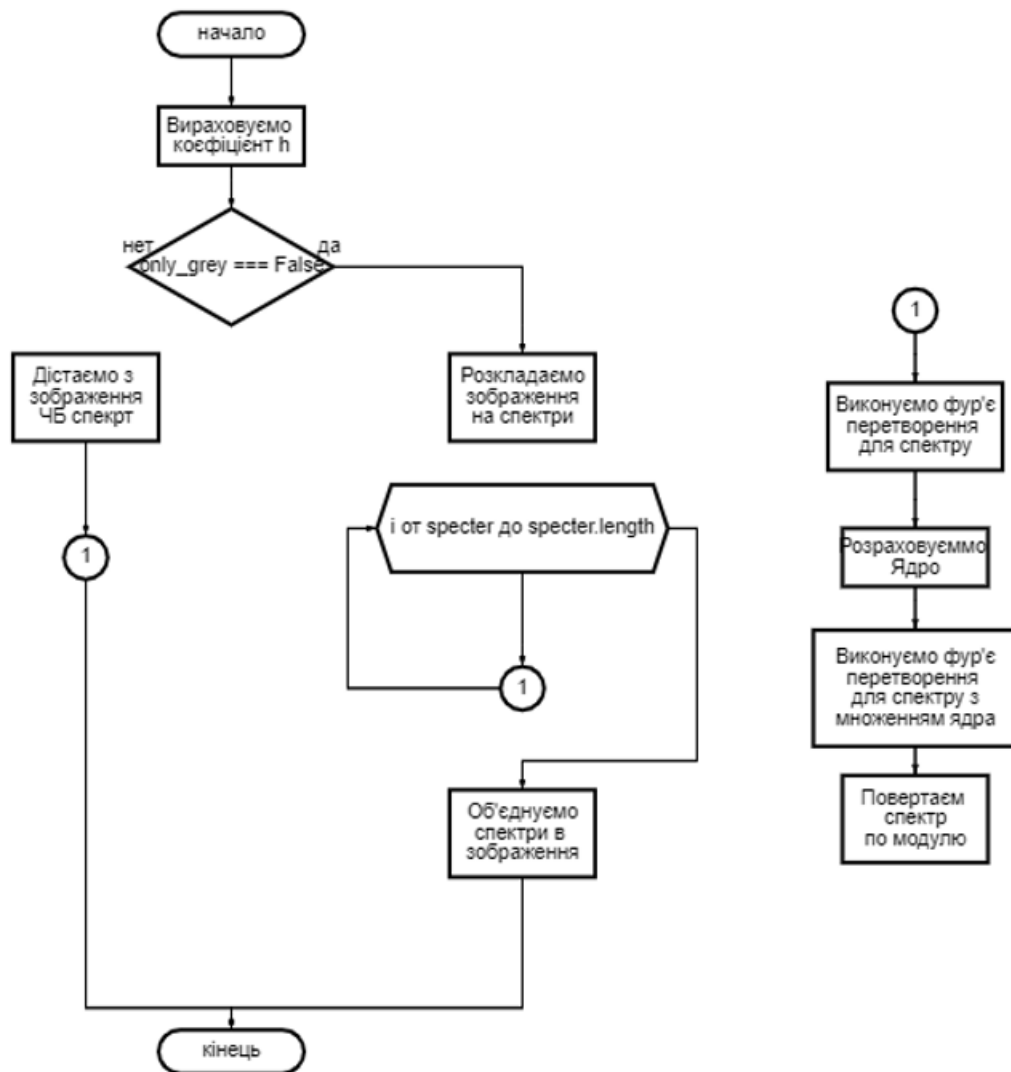


Рис. 3.5.1 Алгоритм Деконвульції Вінера для зменшення шумів.

Розглянемо більш детально як працює цей алгоритм:

- Розраховуємо коефіцієнт H за допомогою “*gaussian*” функції з бібліотеки “*scipy.signal*” та простих математичних перетворень
- Якщо нам потрібно розрахувати зображення в чорно-білому спектрі ми вилучаємо даний спектр та передаємо його в функцію

- При отриманні зображення з RGB спектром ми розкладаємо його на одномірні масиви, та передаємо у функцію
- У функції ми, спершу, копіюємо спектр, потім робим перетворення Фур'є для даного спектру
- Далі за допомогою знайденого коефіцієнту H розраховуємо ядро та виконуємо з ним перетворення Фур'є
- Робимо поелементне комплексне сполучення та ділимо на ядро в другому ступені.
- Виконуємо перетворення Фур'є для спектру помноженого на ядро
- Повертаємо спектр по модулю, якщо в нас зображення не лише в чорно-білому спектрі, то об'єднуємо їх та повертаємо зображення після деконволюції Вінера

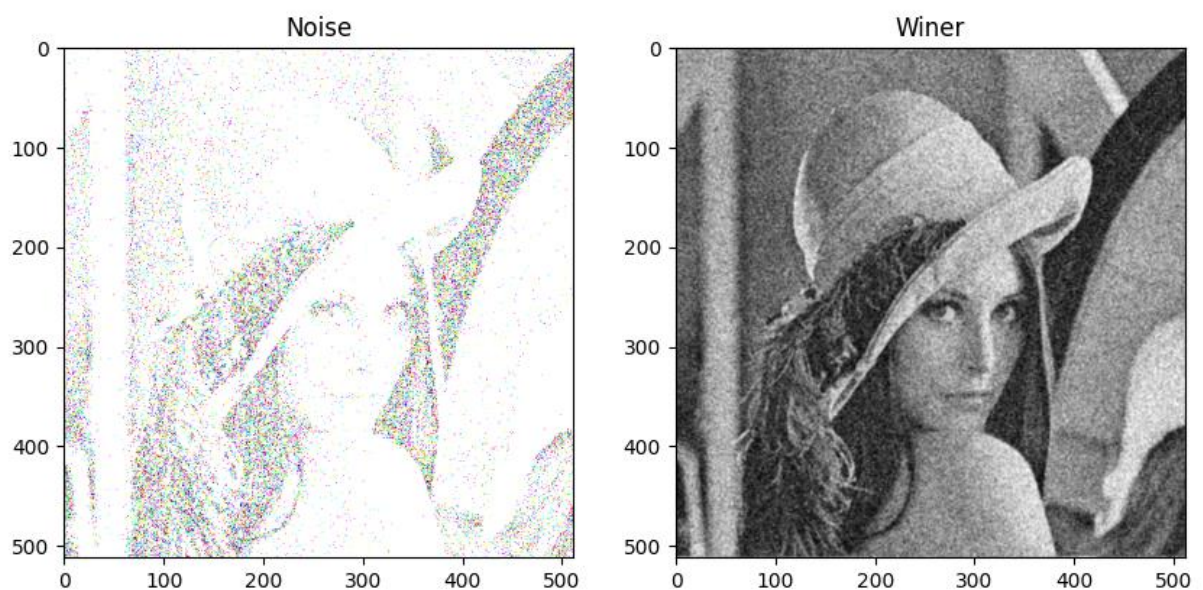


Рис. 3.5.2 Приклад роботи фільтру Деконволюції Вінера.

Підвищення якості. Деконволюція Вінера з PSF

Далі розглянемо деконволюцію Вінера з використанням PSF (Функція розсіювання точки), яка використовується для опису зображення. У багатьох ситуаціях ФРТ має вигляд витягнутої плями, що накладається на

зображення об'єкта. Тобто PSF можна розглядати як просторову версією частотно-контрастної характеристики. Розглянемо реалізацію цієї функції(рис.3.5.3.)

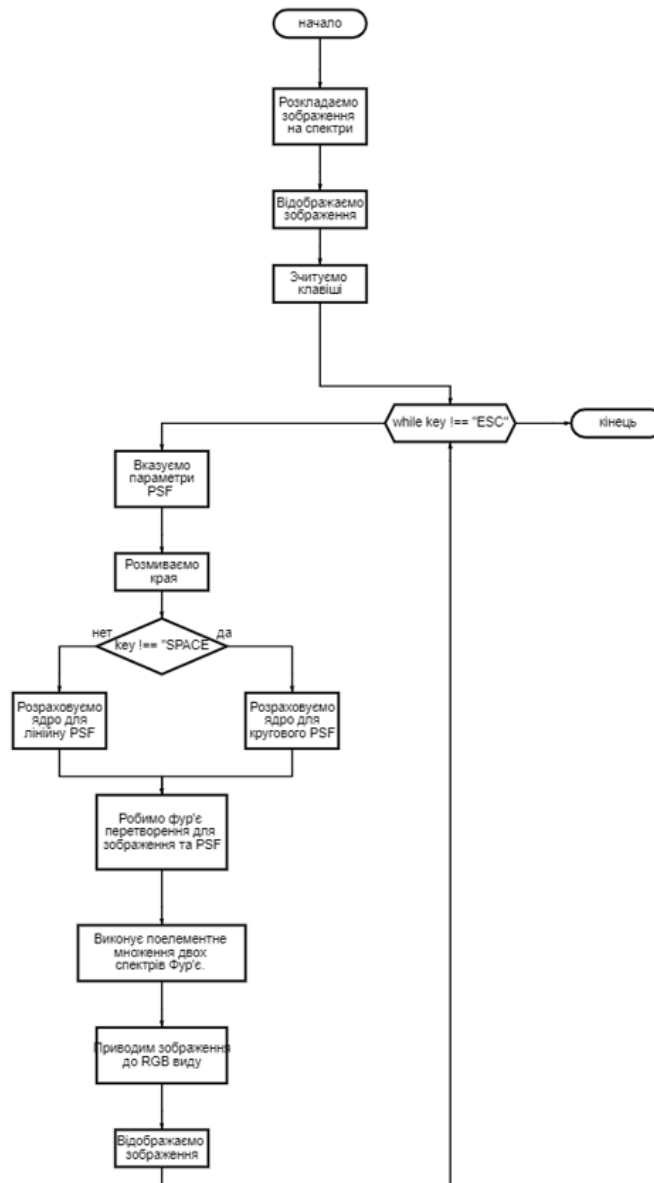


Рис. 3.5.3 Алгоритм деконволюції Вінера з PSF.

Перейдемо до аналізу алгоритму:

- Спочатку ми отримуємо зображення, воно може бути або чорно-білим або кольоровим
- Далі ми розкладаємо це зображення на його спектри та всі наступні перетворення будемо робити з кожним із цих спектрів

- Відображаємо інтерфейс. Запускаємо зчитувач клавіш, який буде завершувати програму, якщо користувач натисне «ESC»
- В цьому інтерфейсі вводимо значення: кута, позиції та SNR
- Далі розмиваємо края для кожного зі спектрів
- Далі розраховуємо ядро для кругового або лінійного типу PSF
- Робимо перетворення Фур'є для зображення та PSF
- Розраховуємо PSF2, яке далі використовуємо для знаходження IPSF, яке дорівнює $PSF / PSF2 + \text{шум}$. З цим значенням виконуємо поелементне помноження спектрів нашого зображення та отриманного IPSF
- Об'єднуємо R,G,B в RGB
- Відображаємо зображення

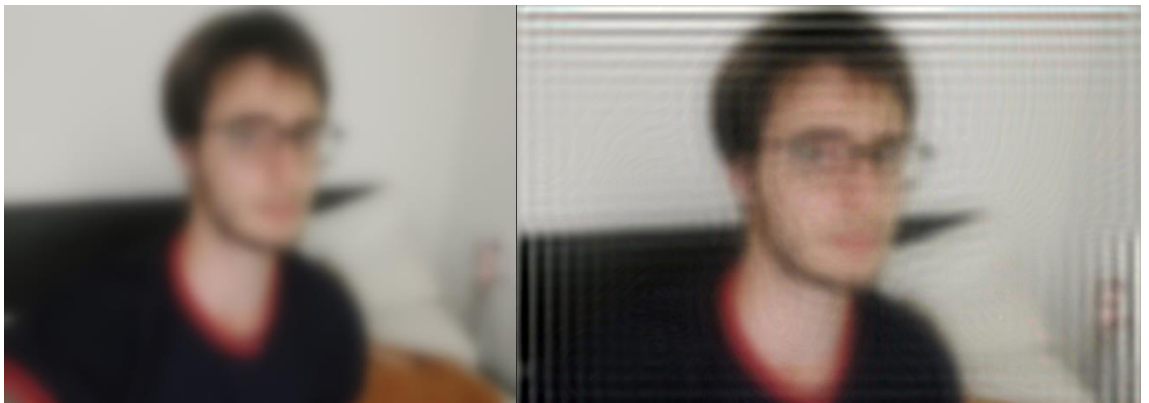


Рис. 3.5.3 Приклад роботи фільтру деконволюції Вінера з PSF.

3.6 Розроблена CNN

Розглянемо розроблену нейронну мережу та покажемо її принципи роботи.

Почнемо з алгоритму роботи нашої NN :

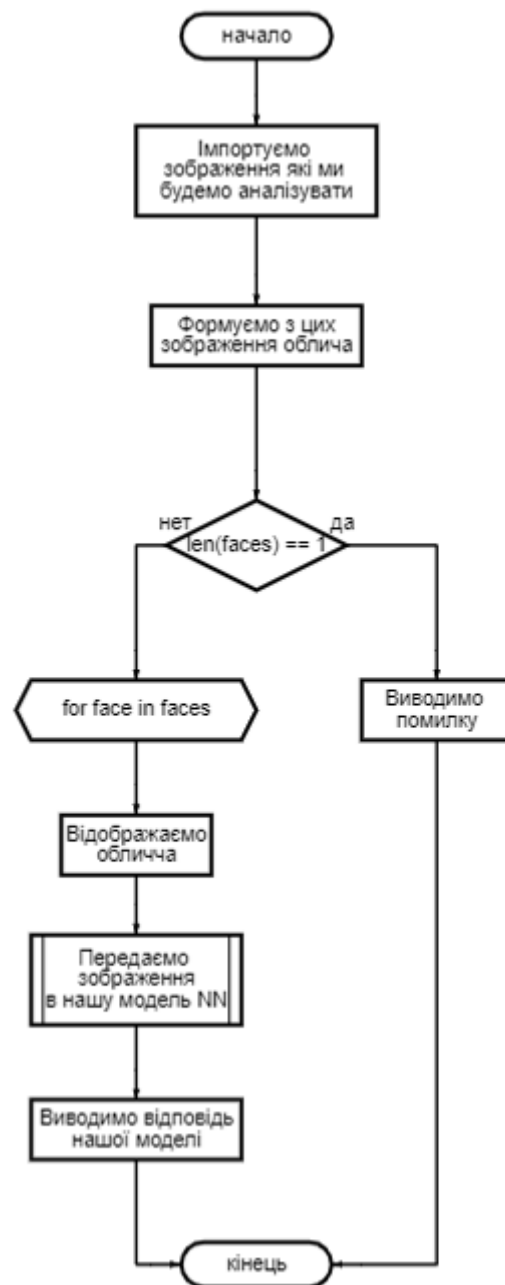


Рис. 3.5.4 Алгоритм розробленої NN.

Розглянемо як працює даний алгоритм :

- Спочатку ми імпортуємо всі наші розроблені функції

- Імпортуємо зображення, на яких ми і шукатимемо обличчя в нашій моделі; якщо на даних зображеннях присутні шуми, то робимо попередню обробку.
- Формуємо з цих зображень обличчя
- Якщо на зображеннях не були знайдені обличчя, то виводимо повідомлення про помилку та зупиняємо роботу програми
- В іншому випадку проходимо по всім зображенням
- Малюємо прямокутник на знайдених обличчях та відображаємо зображення.
- Передаємо обличчя в нашу модель, яка проводить аналіз на наявність їх в її базі
- Виводимо результат

Перевіримо як працює розробленна NN зі звичайними, шумними та розмитими зображеннями

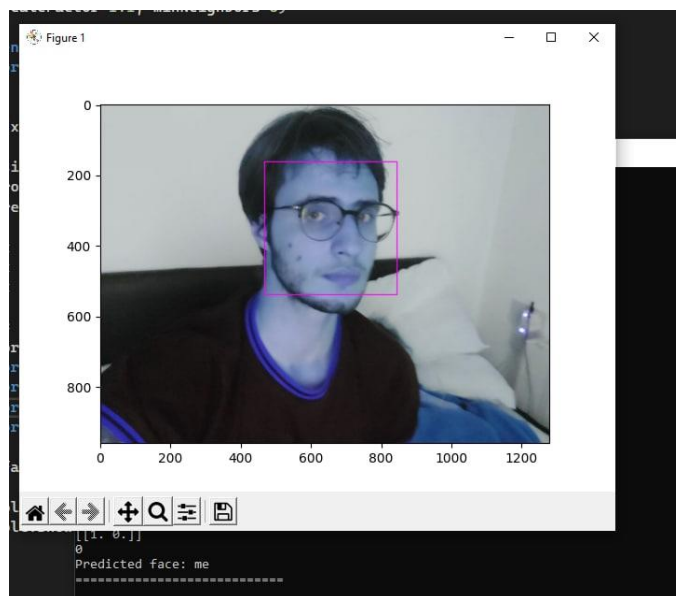
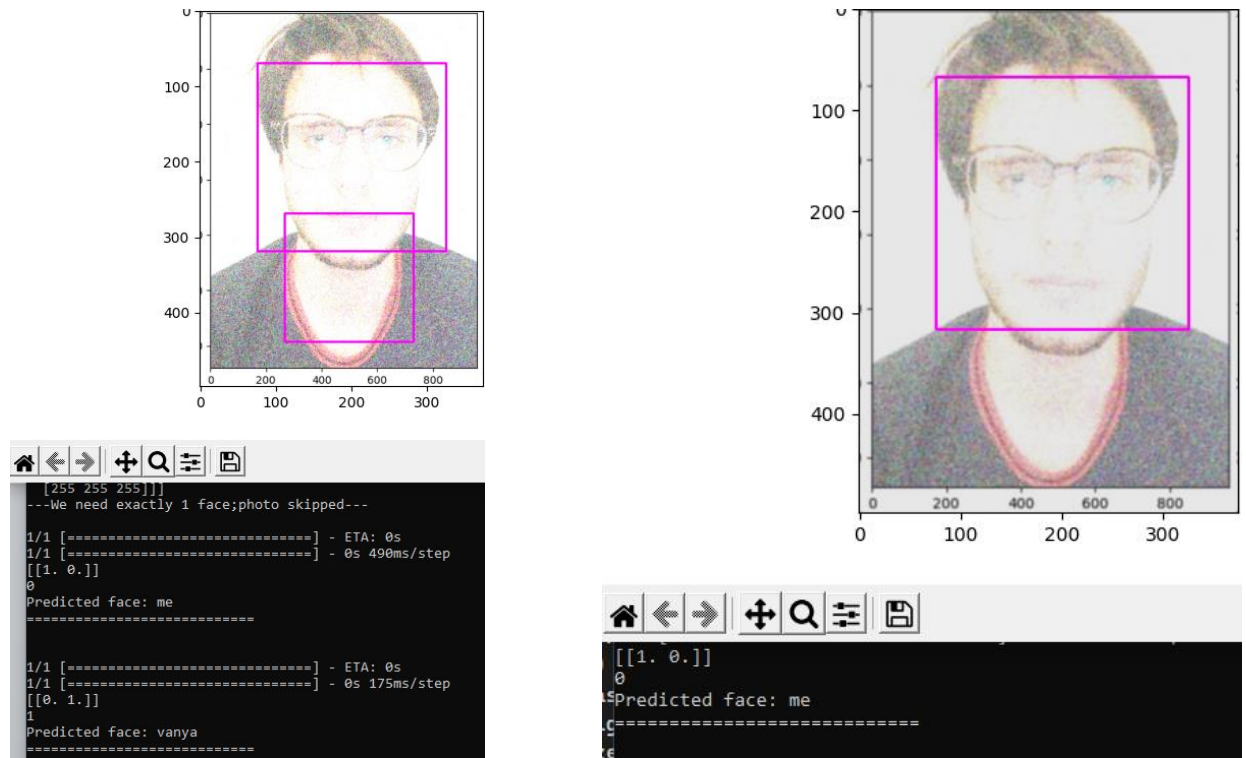


Рис. 3.5.5 Робота NN зі звичайним зображенням.

Як ми бачимо (рис 3.5.5) розроблена NN дала коректну відповідь, відносно звичайного зображення. Тепер додамо певну кількість гаусового

шуму, та перевіримо спершу без використання деконволюції Вінера, а потім з нею



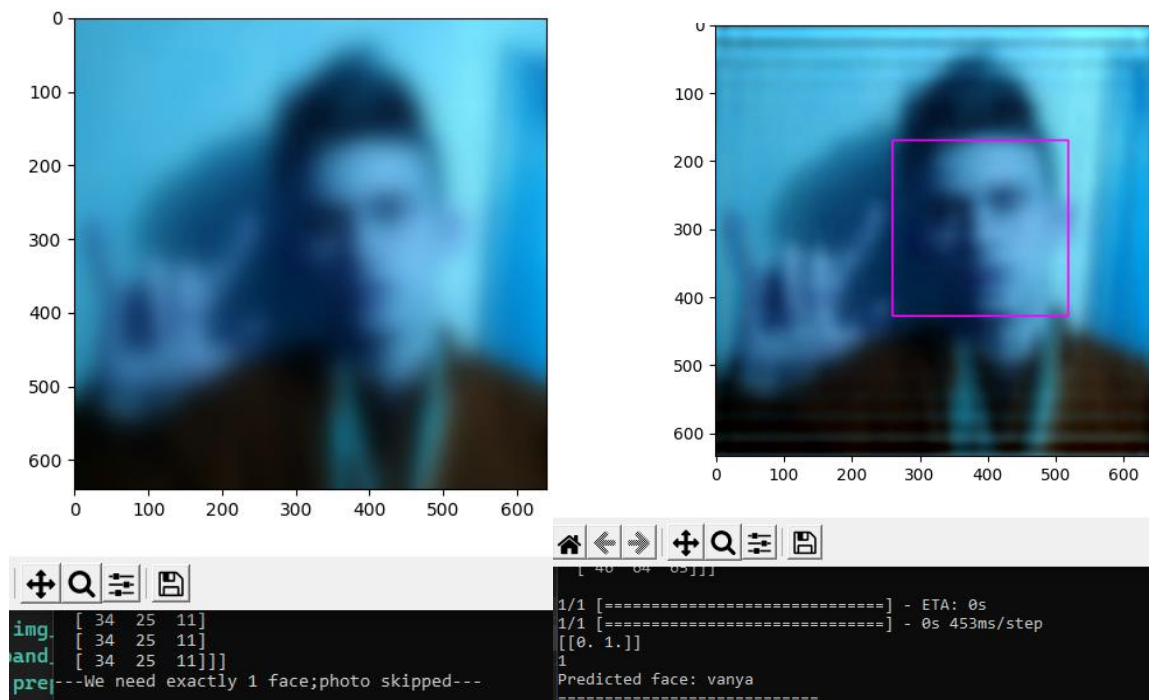
а) Зображення без фільтрації

б) Зображення після деконволюції Вінера

Рис. 3.5.6 Робота NN зі зображеннями, на які ми додали гаусів шум .

Як ми бачемо з наших зображень (рис 3.5.6), після надходження шумного зображення до нашої NN (рис 3.5.6 а.) вона не змогла дати коректну відповідь і відобразила обидва обличчя, на відміну від зображення після деконволюції Вінера (рис. 3.5.6. б.), де нейронна мережа дала відповідь без похибок.

Тепер перейдемо до аналізу роботи нашої NN із заблюреним зображенням



а) Зображення без фільтрації

б) Зображення після PSF Деконвульції Вінера

Рис. 3.5.7 Робота NN із розмитим зображенням

Як ми бачимо (рис. 3.5.7) , коли розмите зображення потрапило до нашої системи, вона не змогла знайти на ньому обличчя тому пропустила його. При використанні PSF Деконвульції Вінера зображення стало достатньо чітким для визначення.

3.7 Висновки до розділу

В цьому розділі було розглянута та розроблена основна робота магістерської дисертації. Були розглянуті запропоновані рішення для вирішення проблем шумних зображень.

Перш за все було описано принцип роботи обраної нейронної мережі, її особливості та складові частини. Як результат було реалізовано згорткову нейронну мережу, були розроблена її структура, модель та функціонал. Як

метод для зчитування облич було обрано каскади Хаара та реалізована з ними функція зчитування облич з зображень.

Другим етапом роботи була мета обрати та розробити найрезультуючий метод фільтрації зображення з метою зменшення гаусового шуму та розмиття зображення. Було обрано деконволюцію Вінера як спосіб боротьби з шумами та деконволюцію Вінера з PSF для зменшення розмитості.

Як результат, були протестовані розроблені методи для боротьби з неякісними зображеннями. Реалізовані методи дали задовільний результат, при якому робота з пошкодженими зображеннями стала можливою.

РОЗДІЛ 4. РОЗРОБКА СТАРТАП-ПРОЕКТУ «CNN ДЛЯ РОЗПІЗНАВАННЯ ОБЛИЧ З МОЖЛИВІСТЮ РОБОТИ ЗІ ШУМНИМИ ТА РОЗМИТИМИ ЗОБРАЖЕННЯМИ»

4.1 Опис ідеї проекту

В 21 сторіччі прогрес не стоїть на місці, і все більше і більше компаній намагаються використовувати комп'ютерний зір для автоматизації людських процесів, проте незважаючи на технологічний прогрес у нинішніх систем є деякі вади, які були виправлені в даному проекті. Отже після розглядання технічної характеристики роботи перейдемо до написання стартап-проекту табл. 4.1.

Таблиця 4.1 Опис ідеї стартап-проекту

Зміст ідеї	Напрямки застосування	Вигоди для користувачів
Система розпізнавання обличчя з можливістю роботи з шумними та розмитими зображеннями	Системи розпізнавання обличчя, які можуть використовуватись на станціях метро, вокзалах, аеропортах.	Використання даної системи дозволить зробити ці місця більш безпечними, а також дасть можливість розробити швидкодійну методологію уникнення злочинів навіть при виникненні пошкодженого зображення.
	Пошук цілей у війсьній чи оборонній промисловості	Мета військових технологій – це забезпечення роботи в будь-який час та при будь-яких погодних умовах, тому нерідко бувають випадки, коли на об'єкти потрапляє забагато світла, що може приводити до непередбачуваних наслідків, тому дуже важливо мати методи для боротьби з шумом.
	Автентифікація у банківських	Головне при роботі з персональними даними – це

	системах та персональних акаунтах	безпека, яка досягається за допомогою ідентифікації обличчя.
--	-----------------------------------	--

Основна ідея даного проекту полягає у використанні графічної фільтрації для зменшення шумів та підвищення якості неякісних зображень. Завдяки цій технології ми зможемо уникнути неточних відповідей нашої системи при пошкодженому обладнанні чи засвітленому об'єктиві. А технологія покращення якості дозволить використовувати застаріле обладнання.

В порівнянні з існуючими аналогами та заміниками ми можемо виділити дві основні переваги нашої розробленої системи:

- 1.) Вартість продукту та його переваги. В теперішні часи з'являється все більше систем комп'ютерного зору, проте дані розробки належать великим ІТ-гігантам, що впливає на їх ціну, та робить ці розробки неможливими до копіювання іншими компаніями.
- 2.) Велика кількість розробленого продукту більш орієнтована на саму технологію комп'ютерного бачення та не працює з шумозаглушенням та підвищенням якості.

Таблиця 4.2. Визначення сильних, слабких та нейтральних характеристик ідеї проекту

№ n/n	Техніко-економічні характеристики ідеї	(потенційні) товари/концепції конкурентів				W	N	S
		Мій проект	Google	IBM	Kairos			
1	Вартість програмного забезпечення	15.000\$	150.000\$	100.000\$	80.000\$	-	-	+
2	Екологічність	Не впливає на середовище	Не впливає на середовище	Не впливає на середовище	Не впливає на середовище	-	+	-
3	Надійність	4 років	7 років	6 років	4 років	-	-	+

4	Економічність (інвестиції)	+	-	-	-	+	-	-
5	Підтримка	Кожні півроку	Кожні два роки	Кожен рік	Кожен рік	-	-	+
6	Торгова марка	Невідома	Відома	Відома	Відома	+	-	-

Як ми бачимо з (табл. 4.2) однією з найсильніших сторін проекту є невелика ціна відносно існуючих конкурентів. Даний плюс обумовлений наявністю невеликої кількості працівників на проекті та відсутності відомості, як у зазначених компаній, що приводить до зменшення ціни. Нейтральною стороною є підтримка, так як написано нами програмне забезпечення нове, перший час воно буде потребувати постійної підтримки. Поганою стороною є відсутність торгової марки та довіри клієнтів, але після перших продажів це питання буде вирішене якістю розробленого продукту.

4.2 Технологічний аудит ідеї проекту

В межах даного підрозділу проводимо аудит технології, за допомогою якої можна реалізувати ідею створення проекту. Основні існуючі технології комп'ютерного зору зазвичай не використовують методи графічної фільтрації, або не підходять комплексно до вирішення даної проблеми. Одні з найпопулярніших технологій являють собою методи медіанного та гаусового фільтру для боротьби з шумами. Проект передбачає використання комплексного підходу для вирішення цієї проблеми. Дану технологію ми будемо використовувати для обробки статичних зображень в подальшому планується використовувати це для обробки відеопотоку.

Перейдемо до аналізу складових, які зазначені в табл. 4.3

Таблиця 4.3. Технологічна здійсненність ідеї проекту

№ п/п	Ідея проекту	Технології її реалізації	Наявність технологій	Доступність технологій
1	Комплексний метод графічної фільтрації з метою зменшення шумів і підвищення якості	Графічний фільтр Вінера	Наявні, але треба доробити	Доступно, але застаріло
		Комплексний підхід	Наявні	Доступно
		Нейронна мережа	У розробці	Доступно
Обрана технологія реалізації ідеї проекту: Використання комплексного підходу для підвищення якості вхідних даних зображення				

Як ми бачимо з табл. 4.3 найкращий варіант для реалізації проекту – це комплексний підхід до даної роботи, бо включає в себе найбільш можливий функціонал вирішення поставлених задач. Даний метод не потребує багато часу, звідки можна зробити висновок, що реалізація проекту можлива.

4.3 Аналіз ринкових можливостей запуску стартап-проекту

Визначимо ринкові можливості та ринкові загрози, які можуть виникнути під час впровадження проекту. Можливість провести даний аналіз проекту дозволяє виявити потенціал та спланувати розвиток проекту на ринку. Спочатку проведемо аналітику попиту табл. 4.4

Таблиця 4.4 Попередня характеристика потенційного ринку стартап-проекту

<i>№ п/п</i>	<i>Показники стану ринку (найменування)</i>	<i>Характеристика</i>
1	Кількість головних гравців, од	3
2	Загальний обсяг продажів, грн/ум.од	8000 ум.од.
3	Динаміка ринку (якісна оцінка)	Зростає на 53% у рік (станом на 2019)
4	Наявність обмежень для входу (вказати характер обмежень)	Нерозвинуті технології або нестача open source проєктів
5	Специфічні вимоги до стандартизації та сертифікації	Оцінка відповідності
6	Середня норма рентабельності в галузі (або по ринку), %	60%

Провівши аналіз ринку ми побачили, що динаміка ринку у 2019 році виросла на 53%. Попередній аналіз ринку показав, що у 2019 динаміка виросла майже вдвічі, в порівнянні з 2018 зріст є позитивним, це означає що нема обмежень та специфічних вимог до стандартизації та специфікації. З цього можна зробити висновок, що даний ринок є задовільною платформою для розвитку цієї ідеї.

Після аналізу ринку, визначимо потенційні групи клієнтів та їх характеристики, попередньо оформивши це в таблицю (табл. 4.5).

Таблиця 4.5. Характеристика потенційних клієнтів стартап-проекту

№ п/п	Потреба, що формує ринок	Цільова аудиторія (цільові сегменти ринку)	Відмінності у поведінці різних потенційних цільових груп клієнтів	Вимоги споживачів до товару
	Комп'ютерно-інтегрована система виявлення облич для служб безпеки	Воєнні чи державні підприємства, сайти з лицьовою автентифікацією	Робота з різними видами шумів	Доступність, якість, ціна, швидкість.

Зробивши аналіз цільової аудиторії можна побачити, що головними клієнтами програмного забезпечення є компанії, які потребують надійної роботи даної системи, навіть при спотворених вхідних даних.

У зв'язку з відсутністю відомості на ринку, у клієнтів ще нема довіри до продукту, тому існують певні ризики, які описанні в (табл. 4.6)

Таблиця 4.6. Фактори загроз

№ п/п	Фактор	Зміст загрози	Можлива реакція компанії
1	Збуту	Пошук іноземних та вітчизняних компаній	Реклама, участь у стартап конференціях

2	Науково-технічний	Потреба у покращенні технічної бази	Підтримка проекту
3	Соціальний	Погіршення репутації	Виставки, конференції та реклама
4	Технологічний	Робота з специфікованими шумами	Майбутні оновлення функціоналу
5	Попиту	Відсутність довіри клієнтів	Зменшення цін та реклама

Деякі з головних загроз пов'язані зі стандартними для нових проектів факторами: відсутність репутації, недовіра ринку, потреба в залученні інвестицій. Це все на відміну від нерозвиненості технологій, хоч і буде вирішуватись з часом, проте має великий вплив на перших етапах розвинення проекту. Але незважаючи на загрози, даний проект може дати великий спектр можливостей (табл. 4.6)

Таблиця 4.7. Фактори можливостей

№ п/п	Фактор	Зміст можливостей	Можлива реакція компанії
1	Збуту	Залучення інвестицій	Зустрічі, відкриття іноземного ринку

2	Науково-технічний	Розробка нових видів фільтрації	Отримання більш якісного результату
3	Соціальний	Здобуття популярності	Проведення рекламних компаній
4	Технологічний	Розробка більш якісної кодової бази	Більш швидкий відклик нейронної мережи
5	Міжнародний	Залучення іноземних інвесторів	Прискорення розвитку проекту

Можливості для проекту відкриваються у тому, що він є відносно новим на ринку, тому ним можуть зацікавитися сторонні компанії, що займаються інвестиціями у інноваційні проекти. Відносна новизна проекту та якісна технічна база проекту дає великі можливості на ринку для зацікавлення та залучення інвестицій сторонніх компаній. Основними інвесторами можна вважати компанії, які займаються безпекою та комп'ютерним зором.

Після аналізу факторів загроз перейдемо до аналізу конкуренції на ринку (табл. 4.8)

Таблиця 4.8. Ступеневий аналіз конкуренції на ринку

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)
Олігополістична конкуренція	ІТ гіганти розділили ринок	Реклама компанія направлена на зовнішній ринок
Міжнародна конкуренція	Невелика кількість продукту призводить до зростання конкуренції	Участь у міжнародних тендерах
Внутрішньогалузева конкуренція	Задовольняє потреби своєї вузької галузі	Реклама компанії направлена на внутрішній ринок
Товарно-родова	Існування різних підходів та технологій реалізації	Розвиток технологій
Цінова	Висока цінова політика	Баланс між ціною та технічною якістю продукту

Ступеневий аналіз конкуренції на ринку показав, що, незважаючи на високі її показники закордоном, реалізувати проект в Україні не є проблемою. Після здобуття впливовості на вітчизняному ринку, даний проект може бути покращений та відправлений на іноземний ринок.

Після аналізу конкуренції проведемо більш детальний аналіз умов конкуренції в галузі.

Таблиця 4.9. Аналіз конкуренції в галузі за М. Портером

Складові аналізу	Прямі конкуренти в галузі	Потенційні конкуренти	Постачальники	Товари-замінники
	IBM, Google, Kairos	Наукові команди різних країн	ІТ компанії	Більш дешеві і неякісні реалізації
Висновки:	На ринку України нема прямих конкурентів в цій галузі	Потенційні конкуренти, ще не заявили про бажання вийти на ринок.	Не присутні на ринку.	Є обмеження для роботи на ринку через існування товарів замінників.

Провівши аналіз слабких та сильних сторін, ми можемо прийти до висновку, що незважаючи на можливий важкий старт в маркетинговій та експлуатаційній галузі (у порівнянні з іноземними конкурентами), наявність невисокої ціни, собівартість продукту, просте технологічне обслуговування і можливість вийти на вітчизняний ринок робить цю роботу дуже актуальною в нашій країні.

Після зазначення всіх аналізів обґрунтуємо перелік факторів конкурентоспроможності. Дивлячись на зроблені вище висновки, ми можемо зрозуміти, що так як цей продукт буде випускатись в Україні, він повинен відповідати певним вимогам споживання.

Таблиця 4.10. Обґрунтування факторів конкурентоспроможності

№	Фактор	Обґрунтування (наведення чинників, що роблять
---	--------	---

п/п	конкурентоспроможності	фактор для порівняння конкурентних проектів (значущим)
1	Вартість	Прийнятна ціна та якість, дають гарну відомість нашому продукту на ринку України
2	Підтримка продукції	Довгостроковий термін технічної підтримки
3	Довіра споживачів	Формується після продажу перших версій
4	Висока якість	Завдяки безперервному поліпшенню продукту та його постійній підтримці, даний продукт зможе досягти високої якості та довіри клієнтів

В результаті розглядання факторів конкурентоспроможності, можемо спостерігати необхідність розробки проекту, який міг би конкурувати з факторами ціни та якісністю технічного обслуговування. Тобто вихідна якість продукту повинна бути не нижча ніж у конкурентів, а його ціна залишитися на задовільному для клієнтів рівні. Для більш детального розуміння якості розробленого проекту, перейдемо до аналізу його слабких та сильних сторін (табл.4.8).

Таблиця 4.11. Порівняльний аналіз сильних та слабких сторін проекту

№ п/п	Фактор конкурентоспроможності	Бали 1-20	Рейтинг товарів-конкурентів у порівнянні з проектом						
			-3	-2	-1	0	+1	+2	+3
1	Вартість	17	0	0	0	0	+	0	0
2	Технічна підтримка	16	0	0	0	0	0	+	0
3	Довіра споживачів	13	+	0	0	0	0	0	0
4	Висока якість	19	0	0	0	+	0	0	0

Засновуючись на табл. 4.10 та 4.11 можна зробити висновок, що фактори конкурентоспроможності мають позитивний внесок, що дає нам можливість впровадити нове програмне забезпечення. Головними плюсами є невелика ціна відносно конкурентів, висока якість технічної складової та постійна підтримка даного продукту. Для більш детального розуміння перейдемо до SWOT-аналізу стартап-проекту

Таблиця 4.12. SWOT- аналіз стартап-проекту

Сильні сторони: <ul style="list-style-type: none"> • Вартість • Технічна підтримка • Якість 	Слабкі сторони: <ul style="list-style-type: none"> • Довіра споживачів • Новизна товару на ринку • Залежність від інвестицій
---	--

Можливості: <ul style="list-style-type: none"> • Розробка технічно досконалого вузькоспрямованого проекту, який зможе потягатися із західними аналогами • Відкритість до інвестицій • Можливість отримання державних пільг • Підвищення рівня безпеки в країні • Відсутність конкурентів на вітчизняному ринку 	Загрози: <ul style="list-style-type: none"> • Важке існування в перші часи без інновації • Недовіра з боку клієнтів • Потреба в удосконаленні проекту при появі нових технологій • Потреба в технічній підтримці клієнтів, що призводить до зростання працівників • Можлива поява конкурентів на внутрішньому ринку
--	---

Проведений нами аналіз допомагає нам виділити сильні та слабкі сторони. До сильних сторін відноситься технічна якість нашого проекту, відкритість ринку в Україні. Слабкі сторони пов'язані з новизною проекту та відсутністю стартового капіталу. Засновуючись на їх аналізі ми можемо розробити альтернативні ринкові поведінки (таблиця 4.13).

Таблиця 4.13. Альтернативи ринкового впровадження стартап-проекту

№ п/п	Альтернатива (орієнтовний комплекс заходів) ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації
1	Стратегія використання сильних сторін для уникнення ринкових загроз	60%	2-3 роки
2	Стратегії використання ринкових можливостей для компенсації слабких сторін	80%	2-5 роки
3	Стратегія виходу із ринку	20%	пів року

На основі проведеного SWOT-аналізу були розроблені та оцінені альтернативні ринкові поведінки для впровадження стартапу. Був отриманий їх оптимальний час реалізації. Зі зазначених альтернатив вирішено використовувати сильні сторони для уникнення загроз.

4.4 Розроблення ринкової стратегії проекту

Перш ніж розробити стратегію охоплення ринку розглянемо групи потенційних споживачів (табл. 4.14).

Таблиця 4.11. Вибір цільових груп потенційних споживачів

№ п/п	Опис профілю цільової групи потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтовний попит в межах цільової групи (сегменту)	Інтенсивність конкуренції в сегменті	Простота входу у сегмент
1	Державні установи	Готові	100 одиниць	Середня	Середня
2	Офіси		25 одиниць	Висока	Доволі просто
3	Об'єкти підвищеної безпеки		>500 одиниць	Середня	Складно
Які цільові групи обрано: Державні установи, офіси, об'єкти підвищеної безпеки. Під час аналізу груп було вирішено, що компанія буде працювати з об'єктами, які потребують забезпечення високого рівня безпеки					

Цільовими групами потенційних клієнтів були обрані об'єкти, які потребують забезпечення високого рівня безпеки. Орієнтований попит від 25 одиниць на маленькі установи до більше ніж 500 одиниць на об'єкти підвищеної безпеки. Інтенсивність конкуренції в рамках України є незначною, а завдяки співпраці з державними об'єктами є великі можливості долучення великої кількості інвестицій.

Для роботи в обраному сегменті ринку необхідно сформулювати базову стратегію розвитку (табл.4.14)

Таблиця 4.14. Визначення базової стратегії розвитку

№ п/п	Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентоспроможні позиції відповідно до обраної альтернативи	Базова стратегія розвитку*
1	Стратегія використання сильних сторін для уникнення ринкових загроз	Спрощений недиференційований маркетинг	Якість продукту, постійна технічна підтримка, відносно низька цінова політика	Стратегія диференціації

Базовою стратегією розвитку проекту було обрано стратегію диференціації, альтернативою – спрощений недиференційований маркетинг.

Така стратегія є найбільш доцільною, дозволяє знизити наявність прямих конкурентів та підвищити рентабельність даного продукту.

Наступним кроком є вибір стратегії конкурентної поведінки (табл. 4.16)

Таблиця 4.16. Визначення базової стратегії конкурентної поведінки

№ п/п	Чи є проект «першопрохідцем» на ринку?	Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?	Чи буде компанія копіювати основні характеристики товару конкурента, і які?	Стратегія конкурентної поведінки*
1	На вітчизняному ринку – так	Буде забирати у конкурентів та намагатися знаходити нових.	Так, бо продукт є вдосконаленням існуючих прототипів.	Стратегія наслідування лідера

При визначенні базової стратегії конкурентної поведінки було обрано «стратегія наслідування лідера». Оскільки завдяки наявності переваг лідера та невеликої ціни даний продукт може почати на внутрішньому ринку, набираючи відомість та вдосконалюючись, а через деякий час перейти на зовнішній.

Базуючись на основних вимогах споживачів обраного сегменту до продукту і постачальника розробимо стратегію позиціонування (табл. 4.17).

Таблиця 4.17. Визначення стратегії позиціонування

№ п/п	Вимоги до товару цільової аудиторії	Базова стратегія розвитку	Ключові конкурентоспроможні позиції власного стартап-проекту	Вибір асоціацій, які мають сформувати комплексну позицію власного проекту (три ключових)
1	Вдосконалення алгоритмів фільтрації, швидкодії	Стратегія диференціації	Висока якість, низька ціна, довгострокова підтримка	Якість. Ціна. Технічна підтримка.

На основі результатів даного підрозділу, була сформована система рішень щодо ринкової поведінки компанії. Позиції конкурентоспроможності

дають можливість виконати ці вимоги у повному обсязі. Ключові конкурентоспроможні позиції дають можливість виконати ці вимоги у повному обсязі, тому і було обрано асоціації на базі вимог цільової аудиторії, які мають сформулювати комплексну позицію власного проекту - простота, якість, технічне обслуговування.

4.5 Розроблення маркетингової програми стартап-проекту

Під час розробки маркетингової програми ми повинні почати з визначення ключових переваг концепції потенційного товару. Підсумуємо дані результати в табл. 4.18

Таблиця 18. Визначення ключових переваг концепції потенційного товару

№ п/п	Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами (існуючі або такі, що потрібно створити)
1	Якість отриманого зображення	Використання фільтрації	Комплексний підхід фільтрації до вхідних даних
2	Якість	Простота отримання результату	Збільшення якості та захист від пошкоджених зображень
3	Економічність	Можливість довготривалого використання	При ціні нижчій, ніж у конкурентів клієнт отримує довгий термін технічної підтримки

Проаналізувавши ключові переваги проекту, вигідними маркетинговими сторонами проекту визначено співвідношення ціни до якості. За відносно невелику ціну клієнти отримують швидко дієвий продукт з багатofункціональними алгоритмами фільтрування зображення.

Розробимо трирівневу маркетингову модель товару (табл.4.19).

Таблиця 4.19. Опис трьох рівнів моделі товару

Рівні товару	Сутність та складові		
I. Товар за задумом	Комп'ютерний зір з алгоритмами зменшення шуму.		
II. Товар у реальному виконанні	Властивості/характеристики	М/Нм	Вр/Тх /Тл/Е/Ор
	1.Довговічність 2.Ціна	Нм М	Е Е

	3.Швидкодія 4.Коефіцієнт пропускання 5.Надійність	Нм Нм Нм	Тх Тх Е
	Якість: відповідає нормам ДСТУ 2504:2017		
	Пакування: програмне забезпечення записане на компакт диск.		
	Марка: DNVision (DNV)		
III. Товар із підкріпленням	До продажу : немає вимог		
	Після продажу: підтримка 5 років		
Продукт буде захищений від копіювання за допомогою цифрового ключа.			

Опис даної моделі показав, що основна ідея стартап-проекту полягає в створенні нейронної мережі для комп'ютерного зору, яка зможе працювати з неякісними вхідними зображеннями. Перевагою даної моделі є можливість уникнути неточних відповідей при надходженні неякісної картинки, що надає приладам з даним ПЗ можливість працювати в місцях великої освітлюваності та виконувати обов'язки при замиленні та некритичних поломках. До продажу виробникам потрібно буде створити технічну документацію, в якій буде описано використання даної системи. Проект буде захищений за допомогою цифрового ключа.

Наступним кроком є визначення цінових меж товару (табл. 4.20).

Таблиця 4.20. Визначення меж встановлення ціни

№ п/п	Рівень цін на товари-замінники	Рівень цін на товари-аналоги	Рівень доходів цільової групи споживачів	Верхня та нижня межі встановлення ціни на товар/послугу
1	2000-2500\$	5000-50000\$	4-7 млн. \$ на рік	2000-2500\$

Зробивши аналіз цін товару ми можемо побачити, що ціна на нашому товару нижче, ніж у конкурентів, а верхня та нижня межа лежить в межах 2000-2500\$.

Після цього сформуємо систему збуту (табл.4.21).

Таблиця 4.21. Формування системи збуту

№ п/п	Специфіка поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту
1	Покупка від виробника	Аналіз ринку, реклама	Канал нульового рівня	Продаж через інтернет та власний магазин.

Зазначені функції збуту формують оптимальну систему через розробника та онлайн-магазин. Завдяки цьому ми можемо легко контактувати зі споживачем та надавати їм підтримку.

Перейдемо до концепції маркетингових комунікацій

Таблиця 4.22. Концепція маркетингових комунікацій

№ п/п	Специфіка поведінки цільових клієнтів	Канали комунікацій, якими користуються цільові клієнти	Ключові позиції, обрані для позиціонування	Завдання рекламного повідомлення	Концепція рекламного звернення
1	Клієнти зацікавлені у пошуках нових технологій обробки інформації.	Інтернет, спеціалізовані сайти.	Інтернет-маркетинг.	Зацікавити клієнта.	«Зможеш побачити усе»

В теперішні часи клієнти отримують інформацію про нові продукти завдяки інтернету, тому більша частина рекламної компанії буде проводитись в різних соціальних мережах таких, як Facebook та Instagram. Також планується залучення державних інвестицій.

4.6 Висновки

На основі проведеного аналізу можна стверджувати, що даний стартап-проект має великі шанси на успіх. На даний фактор впливають деякі причини: якісна кодова структура, наявність більш комплексних технологій,

вузькоспеціалізована сфера використання, невелика ціна продукту, відсутність конкурентоспроможності у внутрішньому ринку, відсутність потреби у спеціальному обладнанні.

. Основні напрямки застосування даного продукту – міста, які потребують більш якісних технологій спостереження, бо даний проект має можливість підвищувати якість та зменшувати шуми зображень, що робить використання даного проекту привабливим для таких структур, як державні установи, воєнні об'єкти, аеропорти та інші місця, які потребують упровадження підвищення безпеки.

Відсутність конкурентів на внутрішньому ринку дає велику можливість у реалізації даного продукту для різних охоронних компаній, а оборонна направленість дає великий ґрунт для залучені державних інвестицій. При цьому невелика ціна розробленого стартапу відносно іноземних конкурентів дає можливості реалізувати даний проект на вітчизняному ринку, а потім перейти на зовнішній.

Перспективність ринку та відсутність конкурентів дає можливості залучення великої кількості клієнтів на внутрішньому ринку. Після деякого часу на внутрішньому ринку даний проект зможе зрости, як і в технічності, так й у відомості. Звернувши увагу на конкурентів, ми можемо побачити, що даний проект не відстає від конкурентів, а в деяких аспектах є більш якісним продуктом. Хоч і з рекламною компанією можуть виникнути проблеми, проте можливість залучення державних інвестицій повинна усунути це, а невелика ціна привабити покупця.

ВИСНОВКИ

В даній магістерській дисертації був проведений огляд існуючих нейронних мереж, були досліджені технології для зменшення шуму та підвищення якості, та розроблений власний аналог для вирішення поставлених задач

В першому розділі : було проведено загальний огляд теми, наведені основні етапи дослідження та методи їх реалізації. Перш за все оглянули загальну структуру нейронних мереж, з якими проблемами вони можуть зіткнутись. Як одну з найголовніших проблем було обрано працю з неякісними зображеннями.

В другому розділі Були проведені дослідження існуючих методів для зменшення шуму, підвищення якості, та виявлення країв. Були реалізовані ці методи для оцінки їх ефективності.

В третьому розділі: були реалізовані власні методи для зменшення шуму та підвищення якості , які дали набагато кращі результати ніж запропоновані рішення які ми розглядали в другому розділі. Була реалізована власна NN яка показала добрий результат як при роботі з звичайними так і шумними зображеннями

В четвертому розділі був розроблений стартап по реалізованим напрацюванням, зроблений огляд на можливості реалізації даного продукту, його потенціал та економічні можливості

1. Робота людського зору [Електронний ресурс] // What this. – 2022. – Режим доступу до ресурсу:
https://www.what-this.ru/people/five_human_senses/
2. Neural Networks a comprehensive foundation // Simon Haykin // Prentice hal, Inc . 1999
3. Схема нейронів [Електронний ресурс] // Habr – 2016. – Режим доступу до ресурсу: <https://habr.com/ru/post/312450/>
4. Схема роботи синапсу [Електронний ресурс] // Habr – 2016. – Режим доступу до ресурсу: <https://habr.com/ru/post/312450/>
5. McCulloch (right) and Pitts [Електронний ресурс] // HistoryofInformation – 2015. – Режим доступу до ресурсу:
<https://www.historyofinformation.com/detail.php?entryid=782>
6. Rosenblatt, F. // The Perceptron: A Probabilistic Model For Information Storage And Organization In The Brain. Psychological Review // 1958. 386–408. PMID 13602029
7. Jiao, Licheng; Zhang, Fan; Liu, Fang; Yang, Shuyuan; Li, Lingling; Feng, Zhixi; Qu, Rong // "A Survey of Deep Learning-Based Object Detection" 2019 128837–128868. 1907.09408.
8. Приклад похибки машинного зору OpenAI [Електронний ресурс] // 20Bits – 2016. – Режим доступу до ресурсу:
<https://www.20minutos.es/noticia/4613438/0/openai-muestra-lo-sencillo-que-es-enganar-a-una-inteligencia-artificial/>
9. Приклад роботи RawNeRF [Електронний ресурс] // WE – 2016. – Режим доступу до ресурсу:
<https://weproject.media/articles/detail/razrabotchiki-google-sozdali-neyroset-kotoraya-udalyaet-shum-na-fotografyakh/>
10. Chmidhuber, Jürgen // "Deep learning in neural networks: An overview". Neural Networks. // 2015 85–117. ISSN 0893-6080

11. Auer, Peter; Harald Burgsteiner; Wolfgang Maass // "A learning rule for very simple universal approximators consisting of a single layer of perceptrons" // 2008 786–795. PMID 18249524.
12. Tahmasebi, Pejman; Hezarkhani, Ardeshir // "Application of a Modular Feedforward Neural Network for Grade Estimation". // 2011 25–32. S2CID 45997840.
13. Collobert and S. Bengio. // Links between Perceptrons, MLPs and SVMs.// 2004 1 — 5 ICML 04
14. Hastie, Trevor. Tibshirani, Robert. Friedman, Jerome // The Elements of Statistical Learning: Data Mining, Inference, and Prediction // 2009. 124— 204 New York, NY,
15. J. Moody and C. J. Darken // Fast learning in networks of locally tuned processing units," Neural Computation // 1989 281-294
16. T. Poggio and F. Girosi // Networks for approximation and learning, // 1990 1484-1487 Proc. IEEE 78
17. Roger D. Jones, Y. C. Lee, C. W. Barnes, G. W. Flake, K. Lee, P. S. Lewis, and S. Qian, // Function approximation and time series prediction with neural networks, // 1990 17–21 I-649.
18. Yee, Paul V. & Haykin, Simon // Regularized Radial Basis Function Networks: Theory and Applications. John Wiley.// 2001 12 – 14 ISBN 0-471-35349-3.
19. Dupond, Samuel // "A thorough review on the current advance of neural network structures" // 2019 200–230.
20. Abiodun, Oludare Isaac; Jantan, Aman; Omolara, Abiodun Esther; Dada, Kemi Victoria; Mohamed, Nachaat Abdelatif; Arshad, Humaira // "State-of-the-art in artificial neural network applications: A survey" // 2018 11-19 PMID 30519653.
21. Graves, Alex; Liwicki, Marcus; Fernandez, Santiago; Bertolami, Roman; Bunke, Horst; Schmidhuber, Jürgen. // "A Novel Connectionist System

- for Improved Unconstrained Handwriting Recognition" // 2009 855–868. S2CID 14635907.
- 22.Sak, Haşim; Senior, Andrew; Beaufays, Françoise // "Long Short-Term Memory recurrent neural network architectures for large scale acoustic modeling" // 2014 20–59 14_0338.
- 23.Li, Xiangang; Wu, Xihong // "Constructing Long Short-Term Memory based Deep Recurrent Neural Networks for Large Vocabulary Speech Recognition". 2014 14–29 arXiv:1410.4281 [cs.CL].
- 24.Hyötyniemi, Heikki // "Turing machines are recurrent neural networks". // 1996 13–24 SteP 96
- 25.Miljanovic, Milos // Comparative analysis of Recurrent and Finite Impulse Response Neural Networks in Time Series Prediction // 2012 4 – 10 Indian Journal of Computer and Engineering. 3
- 26.Li, Shuai; Li, Wanqing; Cook, Chris; Zhu, Ce; Yanbo, Gao // . Independently Recurrent Neural Network (IndRNN): Building a Longer and Deeper RNN // 2018. 4 – 24 arXiv:1803.04831 [cs.CV].
- 27.Azam, Farooq // Biologically Inspired Modular Neural Networks. PhD Dissertation, Virginia Tech // 2000 scholar.lib.vt.edu/theses/available/etd-06092000-12150028/unrestricted/etd.pdf
- 28.Hubel, DH and Livingstone // MS. Color and contrast sensitivity in the lateral geniculate body and primary visual cortex of the macaque monkey // 1990 2223—2237 Journal of Neuroscience. 10
- 29.R. Dash and B. Majhi // Motion blur parameters estimation for image restoration, // 2012 1634–1640 Optik, vol. 125, no. 5
30. R. Fergus, B. Singh, A. Hertzmann, S. T. Roweis, and W. T. Freeman // Removing camera shake from a single photograph // 2006 787–794 ACM Transactions on Graphics, vol. 25, no. 3
- 31.Umesh Rajashekar and Eero P. Simoncelli // Multiscale Denoising of Photographic Images // 2016 1—9 New York University

- 32.E. Davies // Machine Vision: Theory, Algorithms and Practicalities // 1990 42—44 Academic Press
- 33.Функція Гауса [Електронний ресурс] //Geeksforgeeks – 2020 – Режим доступу до ресурсу: <https://www.geeksforgeeks.org/apply-a-gauss-filter-to-an-image-with-python/#:~:text=A%20Gaussian%20Filter%20is%20a,to%20get%20the%20desired%20effect.>
- 34.R.A. Haddad and A.N. Akansu // A Class of Fast Gaussian Binomial Filters for Speech and Image Processing // 1991 723–727 IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. 39
- 35.Розподіл Гауса [Електронний ресурс] // apt.cs – 1999 – Режим доступу до ресурсу: <https://apt.cs.manchester.ac.uk/ftp/pub/ai/jls/CS2411/prob97/node18.html>
- 36.Huang, Thomas S.; Yang, George J.; Tang, Gregory Y. // A fast two-dimensional median filtering algorithm // 1979 13–18. TASSP.1979.1163188.
- 37.Робота медіаного фільтру [Електронний ресурс] // neubias – 2022 – Режим доступу до ресурсу: https://neubias.github.io/training-resources/median_filter/index.html
- 38.Ben Weiss // Fast Median and Bilateral Filtering, // 2006 2-4 SIGGRAPH
39. C. Tomasi R. Manduchi // Bilateral Filtering for Gray and Color Images // 1–5
- 40.Zahraa Chaffat Oleiwi // Edge Detection and image sharpening // 2019 1–2 Researchgate
- 41.Jacobs, David // Image gradients // 2005 1 – 5 CMSC 426
- 42.Gonzalez, Rafael; Richard Woods // Digital Image Processing // 2008 2 – 19 ISBN 978-0-13-168728-8.
- 43.Gonzalez, Rafael; Richard Woods // Digital Image Processing // 2008 26 – 29 ISBN 978-0-13-168728-8.

44. Архітектура CNN[Електронний ресурс] // Debuggercafe – 2019–
Режим доступу до ресурсу: <https://debuggercafe.com/convolutional-neural-network-architectures-and-variants/>
45. Ciresan, Dan, Ueli Meier, Jonathan Masci, Luca M. Gambardella, Jürgen Schmidhuber // Flexible, High Performance Convolutional Neural Networks for Image Classification 2011 1237–1242 Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence-Volume Volume Two. 2
46. Ciresan, Dan; Meier, Ueli; Schmidhuber, Jürgen // (June 2012). Multi-column deep neural networks for image classification. // 2012 3642–3649 arXiv:1202.2745

ДОДАТКИ

ДОДАТОК А

ФУНКЦІЯ ВИЛУЧЕННЯ ОБЛИЧ

```
import cv2
import os
import pickle
import numpy as np
from PIL import Image

import matplotlib.pyplot as plt

headshots_folder_name = 'images'
image_width = 224
image_height = 224
facecascade = cv2.CascadeClassifier('cascade.xml')
images_dir = os.path.join(".", "dataset")

for root, _, files in os.walk(images_dir):
    for file in files:
        if file.endswith("png") or file.endswith("jpg") or file.endswith("jpeg"):
            path = os.path.join(root, file)
        else:
            print(file.endswith("png"), '---Not image---\n')
            continue

        imgtest = cv2.imread(path, cv2.IMREAD_COLOR)
        image_array = np.array(imgtest, "uint8")
        faces = facecascade.detectMultiScale(imgtest, scaleFactor=1.1,
minNeighbors=5)

        if len(faces) != 1:
            print(f'---Photo skipped---\n')
            os.remove(path)
            continue

        print(faces)

        for (x_, y_, w, h) in faces:

            face_detect = cv2.rectangle(imgtest,
                (x_, y_),
                (x_+w, y_+h),
                (255, 0, 255), 2)
            plt.imshow(face_detect)
            plt.show()
            size = (image_width, image_height)

            roi = image_array[y_: y_ + h, x_: x_ + w]
            resized_image = cv2.resize(roi, size)
            image_array = np.array(resized_image, "uint8")
            os.remove(path)
            im = Image.fromarray(image_array)

            plt.imshow(im)
            print(im)
            plt.show()
            im.save(path)
```

ДОДАТОК Б

Деконволюція вінера для зменшення гаусового шуму

```
def get_h() :  
    h = gaussian(3, 1).reshape(3, 1)  
    h = np.dot(h, h.transpose())  
    h /= np.sum(h)  
    return h  
  
def wiener_common (image,h,show=False) :  
    K = 10  
    dummy = np.copy(image)  
    dummy = fft2(dummy)  
    kernel = fft2(h, s = image.shape)  
    kernel = np.conj(kernel) / (np.abs(kernel) ** 2 + K)  
    dummy = dummy * kernel  
    dummy = np.abs(iff2(dummy))  
    if show == True :  
        plt.imshow(dummy, cmap = 'gray')  
    return dummy  
  
def wiener (image) :  
    h = get_h()  
    r = np.float32(image[:, :, 2])  
    g = np.float32(image[:, :, 1])  
    b = np.float32(image[:, :, 0])  
    red_i = wiener_common(r,h)  
    green_i = wiener_common(g,h)  
    blue_i = wiener_common(b,h)  
    res_rgb = np.zeros_like(image)  
    res_rgb[:, :, 0] = blue_i  
    res_rgb[:, :, 1] = green_i  
    res_rgb[:, :, 2] = red_i  
    return res_rgb
```

ДОДАТОК В

Створення моделі для виявлення облич

```
import os
import pandas as pd
import numpy as np
import tensorflow.keras as keras

import matplotlib.pyplot as plt
import tensorflow.python.keras.engine
from keras_vggface.vggface import VGGFace

from tensorflow.keras.layers import Dense, GlobalAveragePooling2D

from tensorflow.keras.preprocessing import image
from tensorflow.keras.applications.mobilenet import preprocess_input

from tensorflow.keras.preprocessing.image import ImageDataGenerator

from tensorflow.keras.models import Model
from tensorflow.keras.models import load_model

from tensorflow.keras.optimizers import Adam

train_datagen = ImageDataGenerator(
    preprocessing_function=preprocess_input)

train_generator = train_datagen.flow_from_directory(
    './dataset',
    target_size=(224,224),
    color_mode='rgb',
    batch_size=32,
    class_mode='categorical',
    shuffle=True)

class_dictionary = train_generator.class_indices
class_dictionary = {
    value:key for key, value in class_dictionary.items()
}

NO_CLASSES = len(train_generator.class_indices.values())

base_model = VGGFace(include_top=False,
    model='vgg16',
    input_shape=(224, 224, 3))
for layer in base_model.layers:
    layer.trainable = False

x = base_model.output
x = GlobalAveragePooling2D()(x)

x = Dense(1024, activation='relu')(x)
x = Dense(1024, activation='relu')(x)
x = Dense(512, activation='relu')(x)
x = Dense(NO_CLASSES, activation='softmax')(x)

model = Model(inputs = base_model.input, outputs = x)
model.compile(optimizer='Adam',
    loss='categorical_crossentropy',
    metrics=['accuracy'])
model.fit(train_generator,
```

```
batch_size = 1,  
verbose = 1,  
epochs = 20)  
model.save(  
    'transfer_learning_trained' +  
    '_face_cnn_model.h5')
```

ДОДАТОК В

Деконволюція Вінера для збільшення різкості

```
def winet_psf(_):
    ang = np.deg2rad( cv2.getTrackbarPos('angle', win) )
    d = cv2.getTrackbarPos('d', win)
    noise = 10**(-0.1*cv2.getTrackbarPos('SNR (db)', win))

    if defocus:
        psf = defocus_kernel(d)
    else:
        psf = motion_kernel(ang, d)
    cv2.imshow('psf', psf)

    psf /= psf.sum()
    psf_pad = np.zeros_like(img_bw)
    kh, kw = psf.shape
    psf_pad[:kh, :kw] = psf
    PSF = cv2.dft(psf_pad, flags=cv2.DFT_COMPLEX_OUTPUT, nonzeroRows = kh)
    PSF2 = (PSF**2).sum(-1)
    iPSF = PSF / (PSF2 + noise)[...,np.newaxis]

    RES_R = cv2.mulSpectrums(IMG_R, iPSF, 0)
    RES_G = cv2.mulSpectrums(IMG_G, iPSF, 0)
    RES_B = cv2.mulSpectrums(IMG_B, iPSF, 0)

    res_r = cv2.idft(RES_R, flags=cv2.DFT_SCALE | cv2.DFT_REAL_OUTPUT )
    res_g = cv2.idft(RES_G, flags=cv2.DFT_SCALE | cv2.DFT_REAL_OUTPUT )
    res_b = cv2.idft(RES_B, flags=cv2.DFT_SCALE | cv2.DFT_REAL_OUTPUT )

    res_rgb = np.zeros_like(img_rgb)
    res_rgb[..., 0] = res_r
    res_rgb[..., 1] = res_g
    res_rgb[..., 2] = res_b

    res_rgb = np.roll(res_rgb, -kh//2, 0)
    res_rgb = np.roll(res_rgb, -kw//2, 1)
    cv2.imshow(win, res_rgb)
```