

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

Теплоенергетичний факультет

Кафедра автоматизації проектування енергетичних процесів і систем

"На правах рукопису"

УДК _____

«До захисту допущено»

Завідувач кафедри

_____ О.В. Коваль

(підпис)

(ініціали, прізвище)

“ ____ ” _____ 2018р.

Магістерська дисертація

зі спеціальності - 122 Комп'ютерні науки та інформаційні технології

за спеціалізацією - Геометричне моделювання в інформаційних системах

на тему: Формування сценаріїв розпізнавання інформаційних вкидань на основі нейронних мереж

Виконала: студентка 6 курсу, групи ТР-71мп

_____ Магеримова Ірада Ілгамівна

(прізвище, ім'я, по батькові)

_____ (підпис)

Науковий керівник к.ф – м.н., доцент Коваль О.В

(посада, вчене звання, науковий ступінь, прізвище та ініціали)

_____ (підпис)

Рецензент _____

(посада, вчене звання, науковий ступінь, прізвище та ініціали)

_____ (підпис)

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

Студент _____

(підпис)

Київ - 2018

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

Факультет теплоенергетичний

Кафедра автоматизації проектування енергетичних процесів і систем

Рівень вищої освіти другий, магістерський

зі спеціальності - 122 Комп'ютерні науки та інформаційні технології

за спеціалізацією - Геометричне моделювання в інформаційних системах

ЗАТВЕРДЖУЮ
Завідувач кафедри
Коваль О.В. _____
(прізвище, ініціали) (підпис)
« ____ » _____ 2018р.

**ЗАВДАННЯ
НА МАГІСТЕРСЬКУ ДИСЕРТАЦІЮ СТУДЕНТУ**

Магерамовій Іраді Ілгамівні

(прізвище, ім'я, по батькові)

1. Тема дисертації Формування сценаріїв розпізнавання інформаційних вкидань на основі нейронних мереж

Науковий керівник к.ф – м.н., доцент Коваль О.В.
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від “ ____ ” _____ 2018 року № 4012-с

2. Строк подання студентом дисертації “ ____ ” _____ 2018 року

3. Об'єкт дослідження сценарії розпізнавання інформаційних вкидань на основі нейронних мереж.

4. Предмет дослідження сценарії розпізнавання та ідентифікації облич на основі нейронних мереж.

5. Перелік питань, які потрібно розробити _____

1) проаналізувати сучасні алгоритми розпізнавання облич;

2) проаналізувати сучасні алгоритми ідентифікації облич;

3) проаналізувати сучасні методи попередньої обробки зображень

4) сформулювати сценарій розпізнавання інформаційних вкидань на основі нейронних мереж;

5) реалізувати обрані алгоритми.

6. Орієнтований перелік ілюстративного матеріалу _____

мета, постановка задачі, алгоритм навчання нейронної мережі, архітектура нейронної мережі, результати дослідження.

7. Орієнтований перелік публікацій _____
 «Формування сценаріїв розпізнавання інформаційних вкидань на основі нейронних мереж // Сучасні проблеми наукового забезпечення енергетики: Матеріали XVI Міжнародної науково-практичної конференції аспірантів, магістрантів і студентів, 2018. – С. 240.

8. Дата видачі завдання « ___ » 2017р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації	Примітка
1	Аналіз проблеми формування сценаріїв розпізнавання інформаційних вкидань на основі нейронних мереж	11.09.2017-12.11.2017	
2	Аналіз існуючих реалізацій та шляхів вирішення проблеми розпізнавання облич	13.11.2017-15.01.2018	
3	Аналіз вимог завдання, вибір методів і засобів розв'язання поставленої задачі	16.01.2018-05.03.2018	
4	Підготовка матеріалів магістерської дисертації	06.03.2018-27.06.2018	
5	Аналіз вимог до програмного застосунку	28.06.2018-18.08.2018	
6	Моделювання схеми роботи програми	19.08.2018-30.08.2018	
7	Розробка архітектури програмного забезпечення	01.09.2018-20.09.2018	
8	Розробка дизайну графічного інтерфейсу	21.09.2018-10.10.2018	
9	Розробка програмного застосунку	11.10.2018-30.11.2018	
10	Оформлення документації	01.12.2018-07.12.2018	

Студент

 (підпис)

Магермова І.І.
 (прізвище та ініціали)

Науковий керівник

 (підпис)

Коваль О.В.
 (прізвище та ініціали)

РЕФЕРАТ

Структура й обсяг дипломної роботи. Магістерська дисертація складається зі вступу, 6 розділів, висновку, переліку посилань з 74 найменувань, 2 додатків і містить 27 рисунків, 22 таблиці. Повний обсяг магістерської складає 113 сторінок, з яких перелік посилань займає 6 сторінок, додатки – 6 сторінок.

Актуальність теми. В сучасному світі інформація поширюється дуже швидко, тому важливо вміти виявляти вкидання. Зображення з обличчями людей не є винятком, щоденно поширюється сотні тисяч фото. У зв'язку з цим, спостерігається неабиякий інтерес до проблеми розпізнавання лиць та їх ідентифікації з метою виявити інформацію, що є вкиданням. Такі системи широко застосовуються в охоронній та антитерористичній сферах, криміналістичній експертизі, верифікації, різних мобільних пристроях. Зокрема, системи ідентифікації обличчя не потребують фізичного контакту з людиною, на відміну від використання біометричних показників, що застосовуються для подібних цілей, саме тому вони найбільш підходящими для масового застосування.

Саме розпізнавання та ідентифікація обличчя є одним з напрямків комп'ютерного бачення, який активно розвивається. Правильно побудована система розпізнавання вкидань здатна вирішити широкий спектр проблем, таких як підрахунок унікальних відвідувачів магазину чи контроль безпеки та правопорядку.

Мета дослідження. Мета дослідження полягає у формуванні сценаріїв розпізнавання вкидань на основі нейронних мереж за допомогою аналізу існуючих методів розпізнавання. Необхідно виявити найефективніші методи та засоби, що вирішують проблему розпізнавання вкидань з використанням мінімальних ресурсів.

Для досягнення поставленої мети були сформульовані наступні **завдання дослідження**, що виявили структуру дослідження:

- дослідити існуючі методи розпізнавання обличчя;

- дослідити існуючі методи ідентифікації облич;
- дослідити топології нейронних мереж;
- дослідити методи навчання нейронних мереж;
- адаптувати методи попередньої обробки зображень для поставленої задачі з метою навчання нейронної мережі;

- розробити програмне забезпечення, сформувавши сценарій розпізнавання інформаційних вкидань на основі нейронних мереж;

Об`єктом дослідження є сценарії формування ідентифікації інформаційних вкидань.

Предметом дослідження є методи та засоби ідентифікації та розпізнавання облич на зображеннях, виявлення інформаційних вкидань.

Методи дослідження. Розв'язання поставлених задач виконувалось засобами комп'ютерного бачення та використанням нейронних мереж, зокрема наступними методами:

- нейронномережевий метод;
- метод з використанням згорткової нейронної мережі;
- метод з використанням бібліотеки OpenCV;
- метод з використанням відкритої бібліотеки машинного навчання

Tensorflow;

- методи розпізнавання облич на зображеннях;
- методи ідентифікації облич;
- методи попередньої обробки зображень для отримання даних на основі яких буде відбуватись навчання нейронної мережі.

Наукова новизна одержаних результатів. Найбільш суттєвими науковими результатами магістерської дисертації є:

- формування сценарію розпізнавання та ідентифікації лиць;
- удосконалено процес розпізнавання інформаційних вкидань шляхом

сумісного використання методів розпізнавання та ідентифікації облич, методів попередньої обробки зображень;

- реалізація методів попередньої обробки зображень з використанням параметрів, які є оптимальними для вирішення поставленої задачі;

- визначення архітектури та параметрів нейронної мережі, які є оптимальними для вирішення поставленої задачі та використання локальної бази даних;

Практичне значення одержаних результатів. Програмна система може використовуватись спеціалістами в галузі розпізнавання інформаційних вкидань на основі нейронних мереж для розпізнавання облич та їх подальшої ідентифікації.

Ключові слова. *НЕЙРОННА МЕРЕЖА, ЗГОРТКОВА НЕЙРОННА МЕРЕЖА, СИСТЕМИ РОЗПІЗНАВАННЯ ІНФОРМАЦІЙНИХ ВКИДАНЬ, СИСТЕМИ РОЗПІЗНАВАННЯ ОБЛИЧ, СИСТЕМИ ІДЕНТИФІКАЦІЇ ОБЛИЧ, МЕТОДИ ПОПЕРЕДНЬОЇ ОБРОБКИ ЗОБРАЖЕНЬ, ВАГИ НЕЙРОНІВ, ПОШИРЕННЯ ПОХИБКИ, ПЕРЦЕПТРОН.*

ABSTRACT

Structure and volume of thesis. The master's dissertation consists of an introduction, 6 sections, a conclusion, a list of references from 74 titles, 2 applications and contains 27 drawings, 22 tables. The full volume of the master's dissertation is 113 pages, of which the list of links takes 6 pages, applications – 6 pages.

Actuality of problem. In modern world information is spreading very quickly, in this regard, it is of note to be able to detect fake information. Images with human faces are no exception, hundreds of thousands of photos are distributed daily. In this regard, there is a considerable interest in the problem of identifying and identification faces in order to detect fake information. Such systems are widely used in security and anti-terrorist spheres, forensic expertise, verification, and various mobile devices. In particular, facial identification systems do not require physical contact with a person, in contrast to biometric indicators usage. That is why such systems are most suitable for mass application.

Face recognition and identification is one of the areas of active computer vision. Detection system can solve a wide range of problems such as counting of unique store visitors or security monitoring.

The aim of the study. The aim of the study is to create scenarios of information fakes recognition based on neural networks by analyzing existing recognition methods. It is necessary to identify the most effective methods and tools that solve the problem of fakes detection with using of minimal resources.

To achieve this goal the following objectives were formulated research **following tasks** the logic and structure:

- the process of information fakes recognition has been improved by combined usage of methods of face recognition and identification, methods of image pre-processing;

- realization of image pre-processing methods with using parameters that are optimal for solving a given task;
- defining the architecture and parameters of the neural network that are optimal for solving the problem and using a local database.

Object of research. Identifying information fakes scenarios.

Subject of research. Methods and means of image identification and recognition, detection of information fakes.

Research methods. The solution of the set tasks was carried out by means of computer modeling, in particular by the following methods:

- neural network method;
- a method using a convolutional neural network;
- a method using the OpenCV library;
- a method using an open library of machine learning Tensorflow;
- the methods of recognition faces on images;
- preprocessing methods of data obtaining for the neural network training.

Scientific novelty of the obtained results. The most significant scientific results of the master's thesis are:

- creation of faces recognition scenario;
- the process of information fakes recognition has been improved by the combined usage of face recognition methods and identification, methods of preprocessing images;
- realization of methods of pre-processing of images using parameters that are optimal for solving a given task;
- defining the architecture and parameters of the neural network that are optimal for solving the problem and using a local database.

The practical value of the results. The software system can be used by specialists in the field of recognition of information based on neural networks for face recognition and their subsequent identification.

Keywords. *NEURAL NETWORK, CONVOLUTIONAL NEURAL NETWORK, SYSTEMS FOR IDENTIFICATION OF FAKE INFORMATION, FACE DETECTION SYSTEMS, FACE IDENTIFICATION SYSTEMS, IMAGE PREPROCESSING METHODS, NEURONS WEIGHTS, ERROR PROPAGATION, PERCEPTRON.*

ЗМІСТ

Вступ.....	10
1. Огляд існуючих методів виявлення лиця.....	11
1.1 Емпіричні методи.....	11
1.2 Методи на основі контурних моделей.....	13
1.3 Методи на основі порівняння з шаблоном.....	16
1.4 Методи на основі навчання.....	16
Висновки до розділу 1.....	24
2. Огляд існуючих методів розпізнавання обличчя.....	25
2.1 Метод порівняння еластичних графів.....	25
2.2 Метод Eigenface.....	28
2.3 Метод на основі дерев рішень.....	30
2.4 Приховані Марківські моделі.....	33
2.5 Морфінгові моделі.....	35
2.6 Нейронномережевий метод.....	36
Висновки до розділу 2.....	36
3. Нейронні мережі.....	38
3.1 Загальні відомості про нейронні мережі.....	38
3.2 Модель штучного нейрону.....	40
3.3 Функція активації.....	41
3.4 Види навчання нейронних мереж.....	47
3.5 Типи нейронних мереж.....	52
Висновки до розділу 3.....	70
4. Опис методів реалізації та засобів розробки програмної системи.....	71
4.1 Особливості реалізації модулів системи.....	71

4.2 Структурно – функціональна схема роботи системи.....	72
4.3 Операційна система Windows.....	74
4.4 Середовище розробки PyCharm 2017.....	75
Висновки до розділу 4.....	77
5. Методика роботи користувача з програмною системою.....	78
5.1 Інсталяція та системні вимоги.....	78
5.2 Алгоритми роботи програми.....	78
Висновки до розділу 5.....	81
6. Бізнес-план інноваційного проекту.....	82
6.1. Опис ідеї стартап-проекту.....	82
6.2. Технологічний аудит ідеї проекту.....	83
6.3. Аналіз ринкових можливостей запуску стартап-проекту.....	84
6.4. Розроблення ринкової стратегії.....	94
6.5. Розроблення маркетингової програми стартап-проекту.....	97
Висновки до розділу 6.....	99
Висновки.....	100
Список використаних джерел.....	102
Додаток А.....	108
Додаток Б.....	112

ВСТУП

В мережі інтернет щоденно публікуються сотні тисяч зображень, багато з яких відображують події з новин або є фотографіями людей. Можливість ідентифікації лиць на цих зображеннях дозволить відфільтрувати вкидання та значно розширити можливості пошуку. Сьогодні виявлення вкидань, зокрема ідентифікація лиць, є гострою проблемою у багатьох сферах людської життєдіяльності. Раніше системи розпізнавання в основному застосовувались у сфері правоохорони, але з розвитком інформаційних технологій такі системи застосовуються набагато ширше. В промисловості та корпораціях впроваджуються технології розпізнавання, вони поступово стирають границю між реальним та віртуальним простором.

Формування сценаріїв розпізнавання вкидань на основі нейронних мереж є популярною темою досліджень впродовж останнього десятиліття. Існують різні топології нейронних мереж та алгоритми, які застосовувались для вирішення цієї проблеми, але все ж вона залишається не вирішеною. Кожен метод має свої переваги та недоліки, головними складнощами у вирішенні даного завдання є: зміни у зовнішньому вигляді людини (поява зморшок, бороди, зміна контуру обличчя зі збільшенням чи зменшенням ваги), що була додана до вибірки; зміна ракурсу розпізнавання обличчя; поява перешкод, часткове перекривання лиця предметами; погане освітлення; якість зображення; кольоровий баланс камери. Важливою перевагою в автоматичного розпізнавання облич є те, що для отримання достатньої якості розпізнавання контуру голови та її повороту відсутня необхідність наявності дорогої апаратури [1].

В процесі дослідження потрібно проводити аналітику існуючих засобів та алгоритмів, для розпізнавання зображень, провести аналіз існуючих методів для вирішення поставленої задачі. Сферою застосування програмного продукту є програмні системи, пов'язані з пошуком інформаційних вкидань.

Для формування сценарію розпізнавання інформаційних вкидань необхідно реалізувати алгоритм виявлення і локалізації облич, алгоритм попередньої обробки зображень, з метою подальшого їх використання для навчання нейронної мережі та алгоритм розпізнавання облич. Також важливим завданням є вибір топології нейронної мережі, методу та алгоритму її навчання.

Сценарій розпізнавання інформаційних вкидань може використовуватись в логістичних компаніях з метою відслідкування водія, наприклад, можна встановити його особистість. Також система, що вирішує проблему розпізнавання облич може широко застосовуватись в аеропортах та на вокзалах, банківських установах, правоохоронних закладах, в маркетингових дослідженнях, з метою визначення інтересів людини.

Мета магістерської дисертації полягає в формуванні сценарію розпізнавання інформаційних вкидань на основі нейронних мереж, аналізу та реалізації методу виявлення обличчя на зображенні, аналізу та реалізації методу попередньої обробки зображень, аналізу та реалізації методу розпізнавання обличчя.

1. ОГЛЯД ІСНУЮЧИХ МЕТОДІВ ВИЯВЛЕННЯ ЛИЦЬ

Проблема виявлення обличчя не є новою, тому існує низка методів та алгоритмів для вирішення цієї задачі, кожен з них має свої переваги та недоліки. Розпізнавання обличчя, а саме його локалізація є важливим завданням, адже є першим етапом ідентифікації обличчя, щоб виявити кому належить обличчя і чи є воно в базі даних, спочатку потрібно його локалізувати.

Для рішення даної задачі застосовуються різні підходи серед них: емпіричні методи, метод на основі навчання, метод на основі порівняння з шаблоном, метод на основі контурних моделей.

При розпізнаванні обличчя системі, яка вирішує дану задачу необхідно врахувати сукупність факторів: відмінності облич різних людей; зміна ракурсу обличчя; можливість наявності певних особливостей; зміна виразу обличчя; наявність перешкод на зображенні, що можуть частково перекривати об'єкт; умови зйомки.

1.1 Емпіричні методи

Дані методи базуються на досвіді людини в рішенні задачі розпізнавання облич, їх метою є формалізація та алгоритмізація, на основі яких базується автоматична система розпізнавання. Формується ряд евристик, на основі наявності та відповідності яких, така система може виявити обличчя. Існує два підходи до побудови системи, перший – метод розпізнавання згори до низу та другий – розпізнавання знизу вгору.

В першому випадку будується певний набір правил, яким фрагмент зображення повинен відповідати: лице в більшості є симетричним відносно вертикальної осі; на лиці є наявні певні ознаки; такі особливості лица, як очі, ніс, рот відрізняються від шкіри за забарвленням; особливості лица мають певне розміщення один відносно одного та певні розміри.

В другому випадку, відбувається виявлення певних особливостей, які є характерними для зображення лиця. Межі – різкі переходи яркості, виявлення лиця на зображенні здійснюється за допомогою мапи меж зображення, на якій позначено різкі переходи яркості, з урахуванням характерних ознак людини. Форма особливостей обличчя – характерна симетричність, близькість меж по форм до параболи, результатом є набір точок на зображенні. Колір – надає додаткову інформацію про об'єкт.

Для виявлення пікселів кольору шкіри може застосовуватись метод Байєсівської карти вірогідностей, результатом виконання якого є карта шкіри – напівтонове зображення, на якому інтенсивність кожного пікселя пропорційна мірі близькості кольору пікселя до кольору шкіри. Як правило, інформація про колір не може дати достатньо даних для виявлення обличчя, через помилки кольорової сегментації.

Вчені Янг та Хуанг запропонували систему, в якій було використано ієрархічні знання для виявлення обличчя на зображенні [2]. Дана система складається з трьох правил. Багато масштабна ієрархія зображень створюється шляхом поступового усереднення вхідного зображення, спочатку відбувається сканування зображення на першому рівні, який має найнижче розширення, на наявність потенційних кандидатів. Потім дані передаються на більш високий рівень, де відбувається вирівнювання гістограм і виділення меж зображень, після цього, виділені області передаються на останній рівень, де обличчя кандидатів скануються за допомогою скануючого вікна, в якому застосовуються загальні правила відповідні за особливості обличчя.

Ще один метод було розроблено вченими Котрополосом та Пітасом, він базується на побудові гістограм по вертикалі та горизонталі в області, де перевіряється наявність обличчя [3]. Математичне представлення горизонтальної і вертикальної проєкцій в точці зображення: $HI(x, y) = \sum_{y=1}^n I(x, y)$; $VI(x, y) = \sum_{x=1}^m I(x, y)$, де $I(x, y)$ – інтенсивність зображення в точці (x, y) . Спочатку обчислюється горизонтальна проєкція зображення і знаходяться локальні мінімуми, які визначаються шляхом різких змін в $HI(x, y)$, ці локальні мінімуми відповідають межах обличчя. Подібним чином обчислюється вертикальна проєкція і знаходяться

локальні мінімуми, які визначають розміщення очей, носа і рота. Такі локальні мінімуми зіставляють область на зображенні, де розміщене обличчя.

Емпіричні методи доволі прості в реалізації та не потребують потужних обчислювальних ресурсів, вони дають можливість побудови моделі зображення обличчя та зведення до мінімуму виконання певної кількості перевірок, надають можливість гнучкого налаштування під конкретну задачу шляхом модифікації інтуїтивно зрозумілих параметрів. Проте емпіричним методам притаманна низка недоліків, серед них, високий відсоток не вірних виявлень чи пропусків та обмеженість можливих реалізацій процесів, які відбуваються в мозку під час рішення задачі розпізнавання зображень, так як набір емпіричних знань про людське обличчя не є повним. Емпіричні алгоритми добре виявляють фронтальні лиця, але не є стійкими на зашумлених та складних фонах, їх основною складністю є втілення людських знань в чітко сформовані правила.

1.2 Методи на основі контурних моделей

Виділення контурів використовується як попередній крок в процесі виявлення ознак зображення. Контури зображень це області з високою концентрацією інформації, яка використовується для розпізнавання об'єктів на зображенні. Існує безліч методів виділення границь, які пов'язані з корекцією по гістограмах і бінарізацією зображення. Одним з прикладів реалізації таких методів є сегментація [4]. Сегментація це процес розділення цифрового зображення на певні об'єкти, в залежності від поставленої задачі. Алгоритми сегментації зображень базуються на розривності та однорідності. Розривність – це розбиття зображення на частини відповідно до різких перепадів значень яркості. Однорідність – це розділення на однорідні області заданих критеріїв. Головною ціллю сегментації є виділення об'єктів на зображеннях. Процедура виділення контурів складається з двох етапів. Перший – виявлення ярих перепадів на зображенні, які утворюють контури. Другий – отримані на першому етапі результати порівнюються з пороговим значенням T , який обирається експериментним методом [5]. Якщо $T > |G|$, то піксель з координатами

(x,y) належить контуру і присвоюється значення яркості контуру, якщо $T < |G|$, то піксель належить фону і присвоюється значення яркості фону. Після виявлення контурів зображення, потрібно виявити цільовий об'єкт, а саме обличчя. Його виділення здійснюється за допомогою пошуку пов'язаних областей пікселів і створенням матриці, кожен елемент якої дорівнює номеру об'єкта, якому належить відповідний піксель вхідного зображення. Для виділення перепадів яркості використовують: градієнтний метод, метод Канні, метод з використанням лапласіана.

Градієнтний метод – це метод, який полягає в знаходженні першої похідної функції яркості, його використовують для знаходження наявності контуру на зображенні. Градієнт зображення $f(x,y)$ в точці (x,y) являє собою вектор G , орієнтований в напрямку максимальної зміни яркості.

Математичне представлення градієнта: $|G| = \sqrt{G_x^2 + G_y^2}$, $G_x = \frac{df(x,y)}{dx}$, $G_y = \frac{df(x,y)}{dy}$, де G_x і G_y – компоненти цього вектора. Щоб зменшити обчислювальні затрати, модуль градієнта знаходиться за формулою: $|G| = |G_x| + |G_y|$. Для знаходження частинних похідних градієнта використовують: оператор Робертса, оператор Превітта, оператор Собела, оператор Робінсона.

Метод Кані – є одним з найбільш ефективних контурних методів, в якому застосовуються морфологічні операції.

Спочатку зображення згладжується фільтром, з використанням імпульсної функції, яка описується гауссовим законом:

$$h(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right), \text{ де } \sigma - \text{ параметр, який визначає степiнь}$$

згладжуючої дії імпульсної функції на шум, присутній на зображенні. В кожній точці обчислюється градієнт зміни яркості на зображенні: $|G| = \sqrt{G_x^2 + G_y^2}$. Відбувається подавлення стрибків яркості зображення. В результаті зберігаються значення обчислених градієнтів, які перевищують значення градієнтів в двох сусідніх точках по напрямку градієнта зображення. Потім виконується морфологічна обробка результатів, при якій задаються верхній та нижній пороги.

Всі точки, які перевищили верхній поріг, зберігаються, а точки, які перевищили нижній – зберігаються, якщо вони взаємодіють з точками, які перевищили верхній поріг, інші – виключаються з контурного зображення, яке формується.

Оператор Канні забезпечує краще виділення контурів, ніж інші методи даної групи, навіть при наявності шуму.

Метод виділення контурів з використанням лапласіана – це метод виділення контурів по Лапласу, який відрізняється від градієнтного тим, що межі підкреслюються не залежно від їх напрямку, завдяки цьому особливості контуру на зображенні виділяються сильніше. Метод полягає в обчисленні другої похідної функції яркості. Лапласіан аналогового зображення $f(x,y)$ визначається рівнянням:

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

Лапласіан в чистому вигляді для знаходження контурів не використовують тому, що він занадто чутливий до шуму та використання модуля лапласіана призводить до подвоєння контурів, що значно ускладнює сегментацію. Тому застосовується метод Лапласа і Гаусса – він представляє собою поєднання операторів Лапласа і Гаусса. Математичною моделлю даного методу є похідна від функції:

$$h(r) = -e^{-\frac{r^2}{2\sigma^2}},$$

де $r^2 = x^2 + y^2$, σ - стандартне відхилення. Згортка зображення з такою функцією призводить до його розфокусування, яке визначається значенням стандартного відхилення.

Даний алгоритм працює коректно, якщо об'єкт знаходиться на однорідному фоні. Якщо зображення засвічене чи затемнене, тіні можуть сприйматись алгоритмом, як контури, зв'язку з цим він не буде коректно працювати. Якщо на зображенні багато шуму, то будуть з'являтися хибні знаходження, при низькому розширенні – помилка локалізації.

Можна зробити висновок, що використання методів сегментації не є стійким, адже на складному фоні, при наявності шумів чи низькому розширенні будуть виникати помилки локалізації обличчя.

1.3 Методи на основі порівняння з шаблоном

Методи на основі порівняння з шаблоном є найпростішими, для вирішення задачі розпізнавання людського обличчя на зображенні. Вони базуються на скануванні всього вхідного зображення локальною маскою і співставленні образу в полі цієї маски з певним шаблоном. Шаблоном є зображення певного розміру в якому людське обличчя чи його частина розміщується по центру, а фон відповідає фону вхідного зображення. Використовуються шаблони таких видів: недеформовані – які потрібно масштабувати та зміщувати; деформовані – шаблони, які змінюють свою форму, залежно від вхідних даних [6].

При зіставленні зображень в якості міри близькості виступає середньоквадратична похибка або взаємна кореляція між образом в області маски та шаблоном. Мінімум середньоквадратичної похибки чи максимум взаємної кореляції, отриманої в результаті буде свідчити про найкращу відповідність зображення в області маски шаблону. Недолік функції кореляції полягає в тому, що ця функція є чутливою до зміни амплітуд вхідного зображення і шаблону. Для вирішення цієї проблеми використовується нормована кроскореляційна функція, оскільки вона є інваріантною до змін в зображенні.

Не зважаючи на простоту даного методу, даний підхід реалізує пошук лиця по шаблону швидко і результативно, але він вимагає відповідності розмірів шаблону і вхідного зображення, що є суттєвою складністю. В силу того, що використовується найпростіший алгоритм порівняння, зображення повинні бути зняти в суворо визначених умовах – не допускається змін ракурсу, освітлення, емоційного виразу.

1.4 Методи на основі навчання

Іншою групою методів є методи на основі навчання. Емпіричні методи є складними в реалізації, з зв'язку з тим, що людські знання дуже важко сформулювати у вигляді певних правил, які б могли взаємодіяти з реальними зображеннями. Методи на основі контурних моделей показують погані результати, коли зображення

зашумлене, або має складний фон чи низьке розширення. Тіні за зображенні сприймаються, як контури, таким чином, алгоритм працює не завжди вірно. Ці методи краще використовувати в попередній обробці зображення. Методи на основі порівняння з шаблоном виділяють обличчя в тому випадку, коли зображення має однорідний фон, що є значним недоліком. Методи на основі навчання не мають подібних недоліків і можуть застосовуватись значно ширше.

Дані методи направлені не на формалізацію людських знань в певний набір правил, а на виявлення закономірностей і властивостей зображень облич, застосовуючи методи математичної статистики та машинного навчання. Задача виявлення обличчя відбувається в два етапи. Перший – пошук фрагментів, які є претендентами для класифікації. Другий – розділення знайдених фрагментів на класи. Зображенню ставиться у відповідність вектор ознак, який використовується для класифікації зображень. Кожен піксель зображення стає компонентом вектора ознак, перетворюючи чорно-біле зображення $n * m$ у вектор простору R^{n*m} .

Перевагами даної групи методів є: високі показники знаходження облич на великих колекціях зображень; можливість гнучкого використання апарату математичних функцій для вибору найбільш оптимального методу побудови класифікатора з урахуванням умов поставленої задачі; виключення участі людини із процедури побудови класифікатора, це зменшує можливість помилки при побудові моделі зображення обличчя.

Недоліками даної групи методів є: залежність від орієнтації і масштабу обличчя, більшість класифікаторів не є інваріантними до зміни ракурсу обличчя і зміни його розмірів, потрібна додаткова обробка зображення; залежність від освітлення, яке мало місце в наборі тренування – потрібна додаткова обробка для компенсації впливу освітлення; висока складність обчислення, це ускладнює використання методів в системах реального часу; формування наборів для тренування і навчання класифікатора.

Алгоритм Віюлі-Джонса

В 2001 році було запропоновано даний алгоритм для розпізнавання облич, він використовує метод сканування вікном. Рамка, розміром меншим за вхідне

зображення рухається з певним кроком по зображенню і за допомогою каскаду слабких класифікаторів визначає наявність лиця [7].

Перевагами цього методу є можливість знаходження більше одного обличчя на зображенні, використання простих класифікаторів, що показує хорошу швидкість та дозволяє використовувати даний метод у відео потоці. Проте, він доволі складний в навчанні, бо для навчання потрібна велика кількість вхідних даних і багато часу навчання [8].

Даний алгоритм реалізований у відкритій бібліотеці OpenCV, використовує апріорні знання про форму та колір людського обличчя. Є адаптивними в тому сенсі, що кожен наступний класифікатор будується по об'єктах, які були не вірно класифіковані попередніми класифікаторами. Алгоритм добре працює і розпізнає особливості обличчя під кутом 30 градусів.

Він побудований на таких ключових ідеях: представлення зображення в інтегральному вигляді – для швидкої ідентифікації ознак; виконання ознак Хаару – для видалення ознак на зображенні; використання спеціального методу навчання – бустинг для вибору ознак; отримання ознак на вхід класифікатора, який дає результат; використання каскаду – для швидкого відкидання частин, на яких відсутнє лице.

Алгоритм складається з двох під алгоритмів. Перший – алгоритм навчання, другий – алгоритм розпізнавання. На практиці швидкість роботи алгоритму навчання не є суттєвою, важливою є швидкість роботи алгоритму розпізнавання

Оскільки в алгоритмі відбувається навчання класифікаторів, то потрібна підготовлена вибірка з позитивних та негативних зображень.

На позитивних зображеннях присутні фронтальні лиця розміром 24 на 24 або 20 на 20 пікселів, ці розміри є зручними для навчання, оскільки лице людини може бути маленького розміру і знаходитись далеко. У вибірку для навчання закладені всі інваріанти освітлення, фону та емоцій людей.

Негативні зображення представляють собою набір різних фоторграфій того ж розміру, на яких відсутнє людське обличчя. Кількість позитивних фото – набагато менша, ніж негативних.

Етап навчання – виділення ознак такого типу, розміру і розміщення у вікні пошуку алгоритму, які найкращим чином розділять зображення. Як результат, отримано декілька тисяч особливостей, що об'єднані в каскади. Етап аналізу зображення – алгоритм Віоли-Джонса перевіряє всі розміщення і розміри пошукового вікна на вхідному зображенні, представленому в градаціях сірого, потім відбувається операція віднімання суми значень всіх пікселів зображення, які містяться в чорних областях особливостей із суми значень всіх пікселів зображень в білих областях. Отримана різниця порівнюється з тією різницею, яка було отримана на етапі навчання. У випадку, якщо хоча б для однієї особливості поріг рівності перевищено, вважається, що об'єкт не міститься в пошуковому вікні і воно відкидається (Рисунок 1.1 – Схема алгоритму Віоли-Джонса).

Побудувавши загальну вибірку для навчання, необхідно виявити признаки, за якими буде навчатись класифікатор. В якості признаків класу для алгоритму розпізнавання його авторами були запропоновані признаки Хаару, що були представлені Альфредом Хааром в 1909 році. В задачі розпізнавання лиць застосовуються маски. Кожна маска характеризується розміром світлої та темної областей, пропорціями, а також мінімальним розміром. Вони представляють собою фільтри, які складаються з прямокутних ділянок (Рисунок 1.2 – Признаки Хаара).

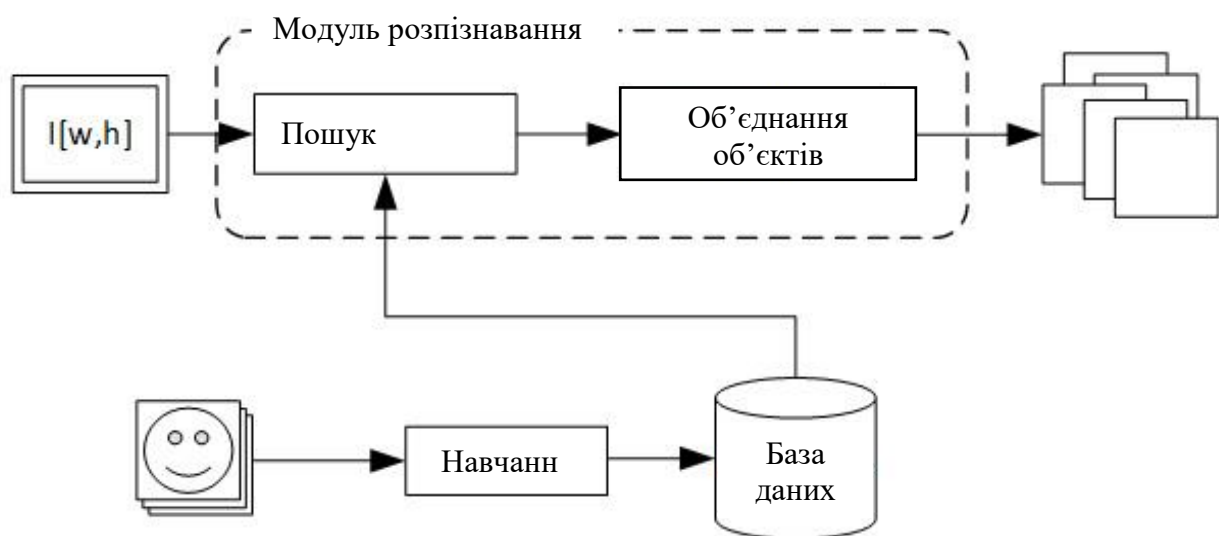


Рисунок 1.1 – Схема алгоритму Віоли-Джонса

Ці прямокутники накладаються на зображення розміром 24 на 24 і відбувається підрахування всіх ярих пікселів, які потрапили в світлу і темну зони [9].

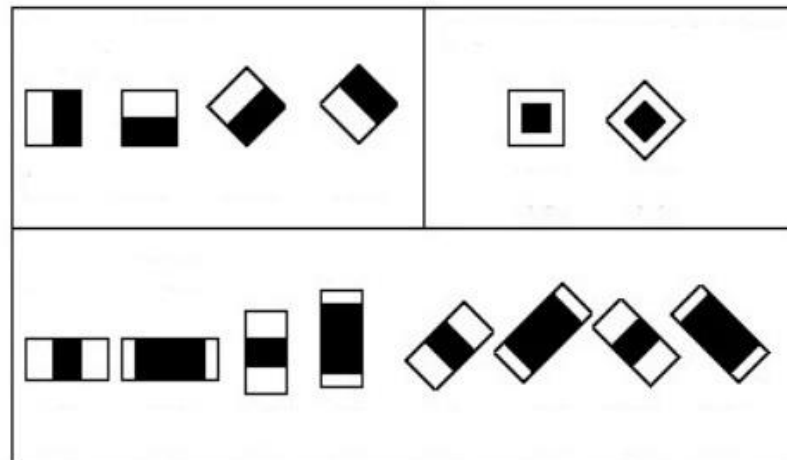


Рисунок 1.2 – Признаки Хаара

Різниця між сумами обчислюється за формулою:

$$f = \sum IS_1 - \sum IS_2 ,$$

Де I – інтенсивність пікселів; IS_1 – всі пікселі в області білого прямокутника, IS_2 – всі пікселі в області чорного прямокутника.

Обчислена різниця і буде значенням певного признаку. Для обчислення ознак з найбільшою швидкістю використовують інтегральне представлення зображень. В інтегральному представленні зображення формується матриця, розміри якої збігаються з розмірами вхідного зображення. В кожному елементі матриці зберігається сума інтенсивностей всіх пікселів, які знаходяться ліворуч і вище даного. Елементи матриці вираховуються за формулою [10]:

$$ii(x', y') = \sum_{x \leq x', y \leq y'} i(x, y),$$

де x, y – координати пікселя на зображення, $ii(x', y')$ – інтенсивність пікселя з координатами x', y' інтегрального зображення, $i(x, y)$ – інтенсивність пікселя з координатами x, y вхідного зображення. Інтегральне зображення обчислюється за один прохід по зображенню, коли воно побудоване, можна обчислити суму пікселів прямокутника, який має будь-яку площу.

Кожен з ознак виявляється дуже швидко, але число ознак є великим, бо для вікна 24 на 24 пікселів це 160000 ознак. Всі ознаки виявити не можливо, тому потрібно обрати деяку підмножину ознак, які дозволять ефективно розділити вибірку навчання на лиця та не лиця. Для вибору конкретної підмножини для навчання використовується метод бустингу. Бустинг – це метод класифікації і розпізнавання образів, який представляє собою процедуру послідовної побудови композицій алгоритмів навчання, коли кожен наступний алгоритм компенсує недоліки попереднього [11]. Класифікатор, який допускає малу кількість помилок називається сильним, а той, який допускає велику кількість помилок – слабким.

В результаті роботи алгоритму бустингу на кожній ітерації формується простий класифікатор. Математичне представлення класифікатора:

$$h_j(z) = \begin{cases} 1, & \text{якщо } p_j f_j(z) < p_j \theta_j \\ 0, & \text{в іншому випадку} \end{cases},$$

де p_j – показує напрямок знаку нерівності, якщо значення признаку негативне і вказує для кожної ознаки окремо, θ_j – значення порога, $f_j(z)$ – обчислене значення ознаки, z – вікно пошуку, j – номер ітерації.

Пізніше було розроблено більш сучасний алгоритм бустингу AdaBoost, він об'єднує прості класифікатори і присвоює кожному з них конкретну вагу. Ця зважена комбінація представляє собою сильний класифікатор.

В методі Віоли – Джонса класифікатори поєднали в послідовність фільтрів. Кожен такий фільтр представляє собою окремий класифікатор AdaBoost, який містить не велику кількість простих класифікаторів.

Алгоритм AdaBoost для пошуку простих класифікаторів складається з трьох кроків:

1. Обчислення всіх ознак Хаара для всіх зображень навчальної вибірки.
2. Виявлення ознаки і порогу, для яких сума ваг помилково класифікованих зображень є мінімальна.
3. Перевизначення ваги всієї вибірки.

Використовуючи метод AdaBoost для пошуку лиць на зображенні потрібно

перевірити велику кількість вікон, оскільки лице на зображенні зустрічається не часто, то потрібно якомога швидше відкидати вікна, на яких немає лиць. Саме з цією метою будується каскад класифікаторів (Рисунок 1.3 – Каскад класифікаторів).

Робота каскаду починається з слабких класифікаторів, які відкидають частину негативних вікон, при цьому приймаючи майже всі позитивні. Якщо перший класифікатор відреагував добре, на фрагмент, де знаходиться потенційне обличчя, то для нього запускається наступний класифікатор, більш складний. Негативний відклик одного із класифікаторів на будь-якому етапі призводить до відбракування вікна. Представлена каскадна структура підвищує швидкість виявлення, акцентуючись на більш інформативних ділянках зображення, там, де присутнє обличчя.

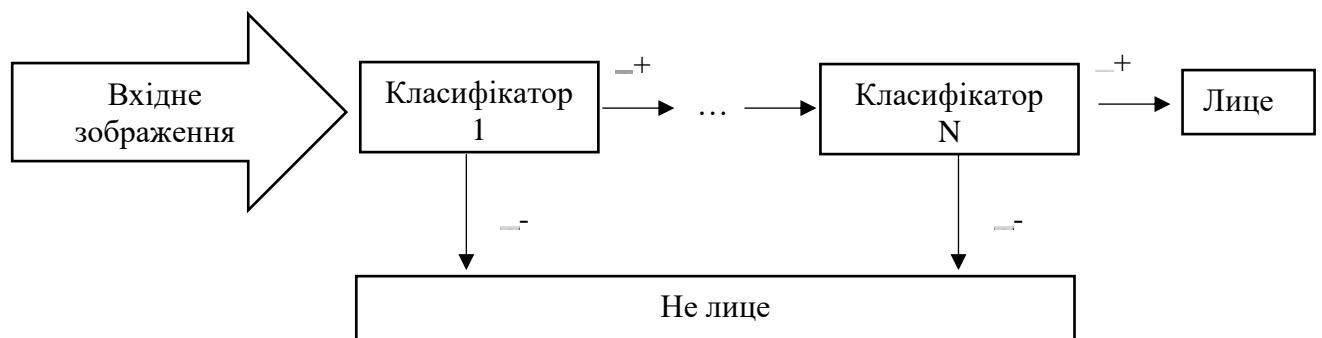


Рисунок 1.3 – Каскад класифікаторів

Метод добре знаходить лиця, на яких присутні предмети, що заважають, частково перекриваючи обличчя та коли присутня міміка.

Для компенсації недоліків традиційного методу AdaBoost було запропоновано гібридний алгоритм, складність якого є лінійною функцією від кількості пікселів зображення, на відміну від квадратичної складності базового алгоритму. Основними змінами є використання кольорової інформації, механізму обрамляючих еліпсів та механізму поворотів.

Гібридний алгоритм складається з двох частин. Перша – є грубою та дозволяє грубо відсікати області, які не містять лиць, використовуючи статистичні характеристики зображення. Друга – використовує безпосередньо алгоритм Віоли – Джонса, модифікований для підвищення швидкості та точності. При цьому

оброблюються лише ті частини зображень, де є велика вірогідності розміщення лиця, а також додаткова інформація, що біла отримана першою частиною алгоритму.

Для додаткового підвищення швидкості роботи алгоритму на етапі виявлення пікселів кольору шкіри і обрамляючих еліпсів використовується попереднє зменшення розмірів зображення в 2 - 4 рази. При цьому відбувається квадратичне зниження об'єму обчислень, пов'язаних з аналізом кольорових характеристик пікселів. Також при цьому не потрібне видалення занадто маленьких обрамляючих еліпсів, так як подібні області кольорового шуму зникнуть в результаті операції пониження розширення. В подальшій обробці виявлені еліпси лиць масштабуються в систему координат вхідного зображення.

Алгоритм складається з п'ятих кроків:

1. Обчислення маски шкіри. Маска шкіри представляє собою однобітне зображення, де одиницями помічені ті пікселі, зображення, які можуть бути шкірою.
2. Отримання і уточнення обрамляючих маску шкіри еліпсів і напрямків їх головних осей.
3. Масштабування положень центрів і розмірів отриманих еліпсів в координати вхідного зображення.
4. Отримання прямокутних ділянок зображення, в яких містяться отримані еліпси.
5. Послідовна обробка отриманих ділянок із запуском модифікованого алгоритму

Віоли – Джонса.

Для підвищення репрезентативних можливостей алгоритму в якості апроксимації особливостей обличчя пропонується використання нового типу трикутних ознак. Виконується виявлення обличчя людини на ділянках, повернутих під усіма кутами з кроком у 20 градусів, починаючи від найбільш пріоритетних напрямків. При цьому виконується до 18 поворотів зображення ($360 / 20 = 18$). У випадку виявлення лиця, нахиленого під певним кутом, обробка поточної ділянки

припиняється і здійснюється перехід до наступної ділянки. Виконується підтвердження виявлення з використанням механізму поворотів.

Представлена модифікація алгоритму має низку переваг: більш висока точність виявлення за рахунок використання трикутних ознак для підтвердження і виявлення з використанням механізму поворотів; більш висока точність виявлення за рахунок відсічення областей зображення, які не є лицями на основі кольорових характеристик зображення; апріорного знання про приблизні розміри обличчя для пошуку алгоритмом Віоли – Джонса на основі аналізу кольорових характеристик і меншої кількості пошукових тонів для обробки; апріорного знання про найбільш вірогідний нахил обличчя до осі зображення і виключення необхідності перевіряти всі кути поворотів.

Висновки до розділу 1

Вибір методу для вирішення задачі виявлення обличчя залежить від конкретної задачі та умов в яких повинен функціонувати алгоритм. Потрібно врахувати такі фактори як: різноманіття облич (фіксована група людей, яка обмежена типом обличчя, відсутність обмежень); орієнтація лиць на зображенні, строго вертикальна чи нахил під певним кутом; кольорове чи чорно-біле зображення; масштаб лиць, розширення, якість зображення – зашумленість рівень зжатості; кількість облич, присутніх на зображенні – відома конкретна кількість, відома приблизна кількість чи взагалі не відома; умови освітлення – фіксовані відомі, приблизно відомі або не відомі; фон – фіксований відомий, контрастний однотонний, слабконтрастний, зашумлений, невідомий; пріоритети важливості – не пропустити жодного лиця чи мінімізувати кількість випадків хибного знаходження.

2. ОГЛЯД ІСНУЮЧИХ МЕТОДІВ РОЗПІЗНАВАННЯ ОБЛИЧЧЯ

2.1 Метод порівняння еластичних графів

В даному методі лице представляється у вигляді графа, вершини якого розміщені на ключових точках – вузлах, таких як контури голови, губ, носа і крайніх точок, кожна грань помічена відстанями між її вершинами.

Кожен вузол характеризується набором ознак, які виявляються в особливих точках обличчя. В якості ознак, як правило, використовуються фільтри Габора для різних частот і орієнтації (Рисунок 2.1 – Фільтри Габора).

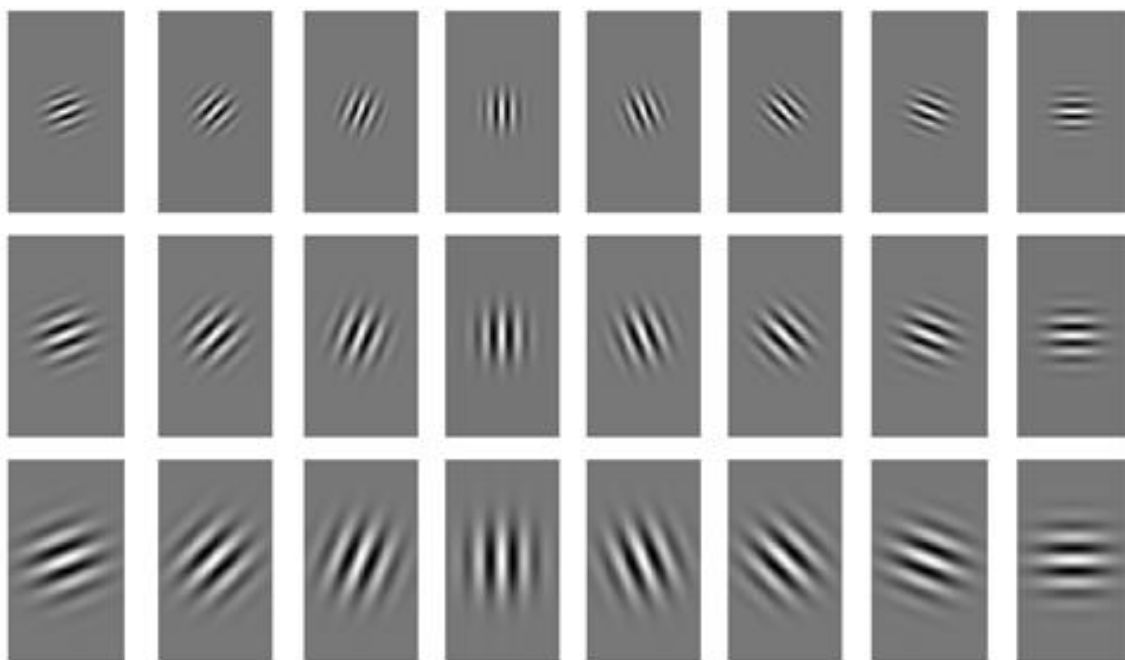


Рисунок 2.1 – Фільтри Габора

Кожен вузол характеризується Евклідовою відстанню до інших вузлів. Система складається з блока виділення обличчя, блока виділення характерних точок обличчя і блока ідентифікації лица.

Блок виділення лиця здійснює знаходження області на зображенні, в якій, скоріше за все розміщене обличчя. Для цього відбувається генерація усередненого графа по всій виборці навчання. Параметрами даного графа є відстань між його вершинами та усереднене значення ознак в кожній вершині. Потім вхідне зображення сканується даним графом в точках зображення, куди накладається граф, включаючи ознаки. В якості найбільш вірогідної позиції обличчя обирається та, у якої найменша сумарна різниця в признака з усередненим графом.

Блок виділення найбільш характерних точок обличчя спочатку визначає найбільш підходящий масштаб графа, який відповідає обличчю на зображення, а потім підставляє вершини графа так, щоб вони відповідали реальним частинам обличчя, за які ці вершини відповідають – глибоке порівняння ознак.

Для виявлення масштабу графа потрібно виявити словник масштабів. Словник масштабів – це міра відмінності усередненого графа від графів – кандидатів, відповідних розпізнаним зображенням з різними масштабами. Для цього обчислюється значення відгуку. Математичне представлення значення відгуку:

$$O(G) = \sum_{m \in G} |w_{m,j} - I_{m,j}| + u_m \sum_{n \in G-m} |V_{mn} - e_{mn}|,$$

де $w_{m,j}$ – признак усередненого графа у вузлі m , $I_{m,j}$ – признак графа-кандидата, V_{mn} – відстань між вузлами m і n , e_{mn} – відстань для графа кандидата. Потім послідовно змінюється масштаб усередненого графа і ці графи накладаються на виділену раніше область вхідного зображення. В результаті отримується граф $Gref \times S_T$. Для кожного зображення із бази і відповідних їм графів G_i обчислюється масштаб, який відповідає найменшому відхиленню цих зображень від $Gref \times S_T$: $r_i = argmin[O(G_i) - O(Gref \times S_T)]$. Далі для r_i будується щільність розподілення ймовірностей і знаходиться її максимум t_{opt} , який приймається за найбільш відповідний масштаб. Після визначення масштабу усередненого графу починається процедура глибокого порівняння ознак. По закінченню процедури перетворення масштабу отримуються грубі позиції характерних точок обличчя, які використовуються як початкова інформація для пошуку точних позицій. Обчислюється початковий відгук за формулою:

$$O_{norm}(G) = \sum_{m \in G} \left\| w_{m,j} - I_{m,j * S_{opt}} \right\| + u_m \sum_{n \in G-m} \left\| V_{mn} - \frac{e_{mn}}{S_{opt}} \right\|$$

Ознаки Габора і Евклідова відстань в рівнянні нормалізується на S_{opt} для масштабу обличчя, потім генерується новий граф пляхом зміни положення вершин і знову обчислюється відгук мережі. Якщо він менший за початковий відгук, то граф-кандидат стає новим графом гіпотез. Процедура гнучкого порівняння ознак повторюється з використанням випадкового дифузного процесу. Значення u_m в рівнянні монотонно знижується в процесі порівняння і визначається за формулою:

$$u_m(t) = u_{ini} \left\| \frac{u_{fin}}{u_{ini}} \right\|^{t_{max}},$$

де u_{ini} – міра положення вершини на початковому етапі процесу порівняння, u_{fin} – міра положення вершини на кінцевому етапі процесу порівняння, t – поточний номер кроку, t_{max} – загальна кількість кроків.

Метод порівняння може ефективно виявляти характерні точки обличчя не зареєстрованого в базі шаблонів тому, що усереднений шаблонний граф використовується як еталонний граф. Також, цей метод чудово працює на зображеннях осіб з аксесуарами, наприклад, окулярами, так як він використовує просторові взаємозв'язки.

Ідентифікація особи відбувається шляхом порівняння графа гіпотез, який було отримано на попередньому кроці з кожним шаблонним графом з бази. Потрім необхідно обчислити відгук O_{norm} . $W_{m,j}$ і $V_{m,j}$ – це ознаки і відстані між вершинами у шаблонних графів. Шаблон з мінімальним відгуком мережі – належить людині, відповідному вхідному обличчю. Якщо відгук мережі менший, ніж певний поріг, то система приймає цю ідентифікацію.

Можна зробити висновок, що даний метод базується на концепції, що зображення реальних лиць мають багато нелінійних характеристик, які не беруться до уваги методами лінійного аналізу. Перетворення Габора створює динамічну архітектуру посилення, які проектують обличчя на еластичну сітку, відбувається опис зображення навколо певної точки. Такий результат має назву фільтру Габора і

використовується на кожному вузлі Габора. Даний процес базується на біології і представляє собою процес, який виконується в корі головного мозку вищих ссавців.

2.2 Метод Eigenface

Алгоритм Eigenface базується на основі методу головних компонент, який вважається доволі ефективним. Ідея методу полягає в тому, що матрицю зображення можна представити у вигляді одновимірного вектора, розташувавши другий стовпець під першим, третій - під другим і по даній аналогії – всі інші. Отримані вектори знаходяться у просторі, який має дуже велику розмірність, а необхідні дані мають меншу розмірності, у зв'язку з цим потрібно знайти оптимальний простір, в якому можна виявити і описати індивідуальні особливості кожної особи. Для вирішення цього завдання використовуються головні компоненти розподілу осіб, які представляють собою власні вектори коваріаційної матриці набору зображень, власні вектори упорядковуються у відповідність з величиною власного значення. Кожен з цих векторів описує особливості особи, їх можна розглядати як набір характерних ознак. Подібним чином, кожна ділянка зображення вносить вклад у кожен власний вектор, саме це представляти вектори, як наближення до фото особи, тому їх називають власними особами (Eigenface). Кожне зображення людини подається у вигляді лінійної

комбінації власних облич (Рисунок 2.2 – Обличчя та власні обличчя).

Розглядається зображення розміром m на n пікселів, це зображення описується вектором розмірністю mn : x_i при $i = 1, 2, 3, \dots, N$, де i – номер зображення. Необхідно знайти вектори, які найкраще описують розподілення зображень облич в середині всього простору зображень. Нехай є набір зображень з векторами x_1, \dots, x_N , які представляють собою вибірку навчання. Середній вектор по всій вибірці навчання обчислюється за формулою: $\bar{x}_0 = \frac{1}{N} \sum_{i=1}^N \bar{x}_i$. Далі необхідно відняти знайдений вектор від кожного зображення: $\bar{\Phi} = \bar{x}_i - \bar{x}_0$. Отримані вектори містять унікальну інформацію про обличчя людини.

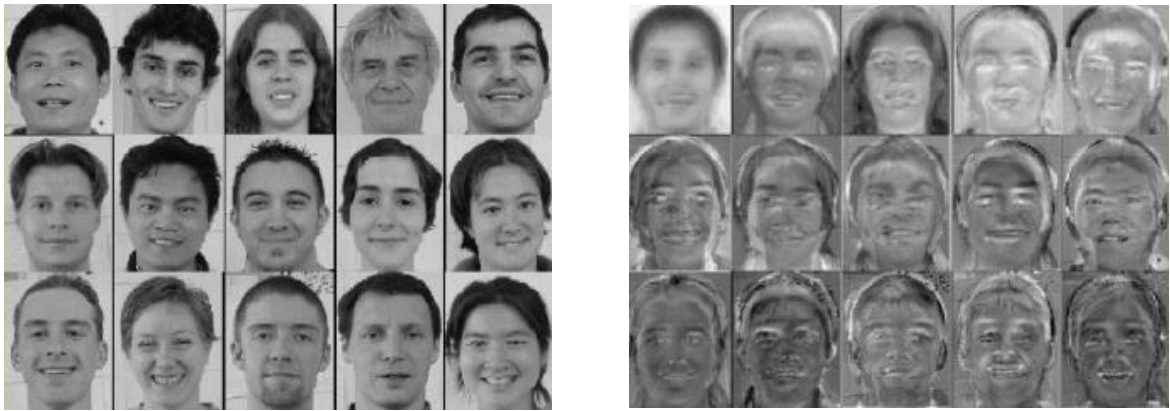


Рисунок 2.2 – Обличчя та власні обличчя

Для найкращого опису розподілення зображень потрібно побудувати набір із N ортогональних векторів \bar{u}_l , де кожен i -тий вектор знаходиться за формулою:

$$\lambda_i = \frac{1}{N} \sum_{k=1}^N (\bar{\Phi}_l \bar{u}_l^T)^2, \quad \bar{u}_l^T \bar{u}_k = \delta_{lk} = \begin{cases} 1, & l = k \\ 0, & \text{інакше} \end{cases}$$

де λ_i – власні значення матриці коваріації, \bar{u}_l – власні вектори матриці коваріації.

Коваріаційна матриця обчислюється за формулою: $K = \frac{1}{N} \sum_{k=1}^N \bar{\Phi}_k \bar{\Phi}_k^T$.

Якщо кількість зображень буде меншою, ніж розмірність простору, то власних векторів \bar{u}_l з ненульовими власними значеннями λ_i буде $N-1$. На цьому етапі процес з навчальною вибіркою завершується.

Тепер на вхід подається тестове зображення з особою \bar{x} , яке розкладається за наявних власних лиць. Далі обчислюються коефіцієнти розкладання, і складається вектор, який описує внесок кожного власного обличчя в представлення тестового зображення. $W_i = \bar{u}_i^T (\bar{x} - \bar{x}_0)$, $\bar{\Omega}^T = \{\omega_1, \dots, \omega_N\}$, де $l=1, \dots, N$.

Зіставимо вектор, який потрібно спроекціувати в новий простір і визначимо до якого зображення навчальної вибірки приклад розміщений найближче, для цього обчислюється відстань між самим зображенням та його проекцією використовуючи формулу: $d^2 = \|\bar{\Phi} - \bar{\Phi}_f\|^2$, $\bar{\Phi}_f = \sum_{i=1}^N W_i \bar{u}_i$, якщо значення d^2 є великим, то вхідне зображення з більшою вірогідністю не належить базі з зображеннями.

Можна зробити висновок, що в даному методі зображення повинні бути одного розміру та бути нормалізованими, вирівняними по лінії очей і рота. Далі використовується метод головних компонент, щоб зменшити розмір даних за допомогою основ стискання даних і виявлення найбільш ефективної низько розмірної структури особливостей обличчя. Таке стискання розмірів видаляє інформацію, яка не є корисною і точно розкладає структуру обличчя в ортогональні компоненти відомі, як власні. Кожне зображення обличчя може бути представлено у вигляді зваженої суми – вектору ознак власних меж, які зберігаються в масиві.

Основою перевагою цього методу є те, що він може зменшувати дані, необхідні для ідентифікації людини о однієї тисячної представлених даних. До недоліків можна віднести великі обчислювальні затрати хвильового перетворення Габора, яке збільшується пропорційно кількості хвиль.

2.3 Метод на основі дерев рішень

Метод на основі дерев рішень представляє собою структуру, яка є послідовною та характеризується певною ієрархією. Вона забезпечує отримання кінцевого рішення про класифікацію об'єкта за допомогою бінарної логіки. При цьому питання, яке задається на наступному ієрархічному рівні залежить від відповіді, що буде отримана на попередньому рівні. Дерево складається з вузлів, які поєднані один з одним ребрами і листових вузлів (Рисунок 2.3 – Приклад простого дерева рішень), важливо те, що ребра не можуть містити цикл, адже в такому випадку, дерево перетвориться в граф. В дереві міститься спеціальний вузол, який називається кореневим. Цей вузол є основою дерева рішень, оскільки від нього можна перейти до будь-якого вузла дерева. Кожен рівень в дереві розглядається як одне із варіантів рішень, кожному вузлу дерева ставитися відповідно питання, яке містить декілька варіантів відповіді, відповідних виходам ребер. Вузли прийняття рішення містять критерій вибору, а ребра перевіряють виконання умов – істина чи хибність.

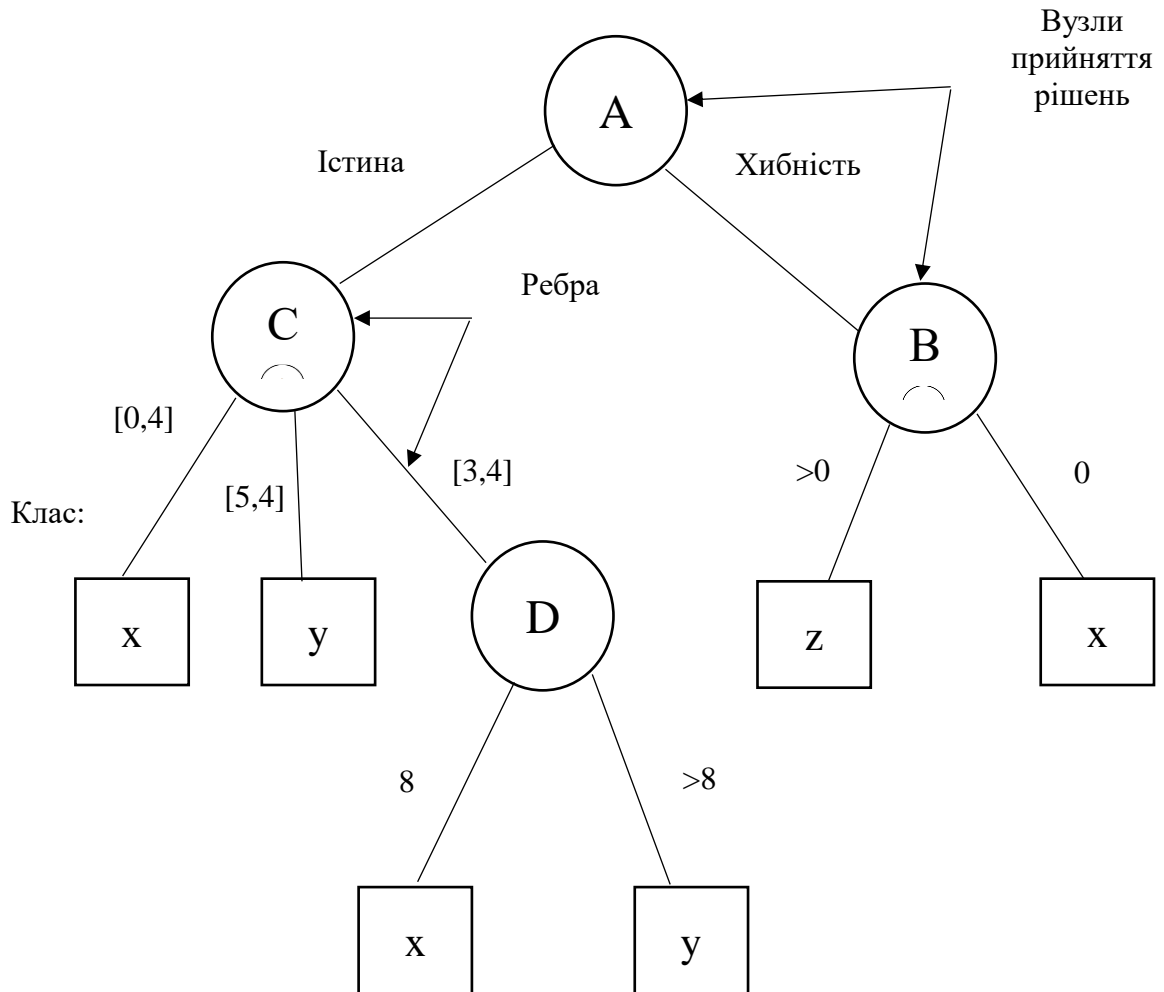


Рисунок 2.3 – Приклад простого дерева рішень

В залежності від обраного варіанту відповіді здійснюється перехід до вузла наступного рівня, поки не буде досягнуто листовий вузол. Листовим вузлом поставлені у відповідність мітки, що вказують на належність об'єкта, що розпізнається до одного з класів. Існує багато алгоритмів, що реалізують побудову дерев рішень, найбільше поширення отримали: CART, C4.5, QUEST.

Алгоритм CART – в даному алгоритмі кожен вузол дерева рішень має двох наслідників, на кожному кроці побудови дерева правило, яке формується у вузлі, ділить задану кількість прикладів, тобто навчальну вибірку, на дві частини. Перша – частина, в якій виконується правило і друга – частина в якій воно не виконується. Для вибору оптимального правила використовується функція оцінки якості розбиття.

Навчання дерева рішень відноситься до класу навчання з вчителем, тобто навчальна і тестова вибірки містять класифікований набір прикладів. Функція оцінки базується на інтуїтивній ідеї зменшення невизначеності у вузлі.

На першій ітерації будуються всі можливі гіперплощини, які ділять простір на дві частини. Для кожного подібного розбиття підраховується кількість спостережень в кожному з підпростору різних класів. В результаті обирається таке розбиття, що максимально відділяє в одному з підпросторів спостереження одного з класів, це розбиття і вважається коренем дерева прийняття рішень, а листами будуть два розбиття. На наступних ітераціях береться найгірший лист і з ним проводиться та ж сама операція розбиття.

Ці кроки повторюються до тих пір, поки не буде досягнуто обмеження по кількості вузлів або від однієї ітерації до іншої перестане покращуватись загальна помилка. Таким чином, дане дерево все одно буде перенавчено, щоб запобігти цьому використовуються тестові вибірки або кросс-валідація, а потім проводиться зворотній аналіз.

Перевагами даного методу є швидкість побудови моделі є швидка побудова моделі, легка інтерпретація. Він непараметричний, тобто для його застосування не потрібно розраховувати різні параметри розподілення вірогідності. Для застосування CART попередній вибір змінних не є необхідним, вони обираються безпосередньо під час аналізу.

Недоліками є нестабільність – результат, отриманий на одній вибірці може відрізнятись на інших даних, а також часта схожимість на локальному рішенні.

Алгоритм C4.5 є удосконаленою версією алгоритму ID3. До нього було додано відсікання гілок, можливість роботи з числовими атрибутами, а також можливість побудови дерева з неповної навчальної вибірки, в якій відсутні певні дані.

Для того, щоб за допомогою C4.5 побудувати дерево рішень і його використання потрібно, щоб інформація про об'єкти мали дискретне або чистове значення та були представлені у вигляді атрибутів – така послідовність має назву приклади. Кількість атрибутів і їх склад є постійним для всіх прикладів.

QUEST – алгоритм, в якому використовуються покращені методи, за допомогою яких вирішуються проблеми вище описаних алгоритмів. Алгоритм в загальному швидший, ніж інші, що використовують побудову дерев рішень, але при роботі з великими базами даних, де вимоги пам'яті значно вищі, він працює повільніше.

2.4 Приховані Марківські моделі

Марківські моделі є потужним засобом для розпізнавання образів. Даний метод дозволяє враховувати просторово-часові характеристики сигналів. Для розпізнавання використовуються одновимірні - 1D-НММ та псевдо-двовимірні – LC 2D-НММ Марківські моделі. У всіх моделях всі зображення облич розбиваються на пов'язані один з одним області, але вони відрізняються між собою. Кожна з таких областей асоціюється з прихованим станом Марківської моделі. В якості стану, що спостерігається прихованої Марківської моделі використовується вікно сканування фіксованого розміру. Це вікно послідовно проходить всі ділянки лица із заданим кроком перекриття. Через чутливість вікна до слабких змін вхідного зображення використовуються різні перетворення, наприклад, двовимірне дискретне косинус перетворення. Як правило, при розпізнаванні облич використовується прихована Марківська модель з непереривною щільністю спостережень, у яких вектор спостережень формується як зважена сума гауссіанів.

Перед навчанням необхідно задати початкові параметри прихованої Марківської моделі, для цього зображення розбивається на частини, які відповідають прихованим станам. В цих частинах розраховуються середні значення векторів спостереження і налаштовуються відповідні значення матриці вірогідності спостережень для кожного стану. При навчанні зображення поступово сканується вікном спостереження у відповідність зі структурою прихованої Марківської моделі і на основі даних, що досліджуються налаштовуються параметри. В результаті навчання, для кожного обличчя формується власна прихована Марківська модель.

При розпізнаванні невідоме лице подається на вхід прихованої Марківської моделі кожного з лиць і обчислюється вірогідність того, що прихована Марківська модель відповідає даному обличчю. В якості відповіді обирається прихована Марківська модель, у якої найбільша вірогідність відповідності даному обличчю, в тому випадку, якщо ця вірогідність більша, ніж поріг невизначеності.

Якщо лице було розпізнане шляхом співвідношення з ним певної прихованої Марківської моделі, може бути проведена сегментація цього обличчя, тобто для кожного із відомих положень вікна спостереження будуть визначені найбільш вірогідні стани.

У випадку одномірних моделей лице послідовно сканується зверху до низу або справа наліво вікном спостереження фіксованого розміру. Стан моделі відповідає розбиттю зображення на частини, які представляють собою набори рядків. В одному з найчастіше використовуваних прикладів лице розбивається горизонтальними лініями на частини, які відповідають п'яти частинам лица.

Нехай зображення має розміри Y по вертикалі і X по горизонталі. Нехай розмір вікна L по вертикалі і H по горизонталі. Це вікно буде сканувати зображення зверху вниз, накладаючи попереднє вікно на M пікселів (Рисунок 10 -). Число блоків визначається за формулою: $T = \frac{Y-L}{L-M} + 1$. Вибір параметрів є важливим, велике число M підвищує якість розпізнавання, при великому L можливий захват зображень із різних станів, при його малих значеннях можливо, що вікно буде містити занадто мало інформації. В загальному при використанні 1D-НММ використовується набір прихованої Марківської моделі $\lambda^{(k)}$:

$$\lambda^{(k)} = (A^{(k)}, B^{(k)}, \pi^{(k)}), 1 \leq k \leq N,$$

де N – загальна кількість різних об'єктів, які потрібно розпізнати; $A^{(k)}$ – вірогідна міра переходу від однієї ділянки до іншої, після навчання $A^{(k)}$ зберігає частоти переходів від однієї ділянки до іншої і товщину відповідних ділянок; $B^{(k)}$ – вірогідна міра досліджуваного вектору ознак при знаходженні в стані k , після навчання $B^{(k)}$ містить векторне розподілення досліджуваного вектору ознак на різних ділянках обличчя; $\pi^{(k)}$ – початкове розподілення вірогідностей, оскільки лице розглядається

зверху до низу, то вірогідність першого стану $-\pi^{(k)} = 1$, а вірогідність інших станів $-\pi^{(k)} = 0, 1 \leq i \leq M, M$ – кількість ділянок обличчя.

Псевдо-двовимірна Марківська модель складається з лінійної моделі суперстанів. При цьому суперстани представляють собою приховані Марківські моделі. Кожен суперстан відповідає розбиттю зображення на рядки, а послідовні переходи по станах всередині суперстану – проходю справа на ліво або зверху в них по заданому рядку або стовбцю. Перехід в новий суперстан можливий тільки в тому випадку, коли система знаходиться в фінальному стані моделі із попереднього суперстану. Вікна спостереження обираються так, щоб не вийти за межі суперстану. Сканування відбувається з певним коефіцієнтом перекриття.

Спрощена двовимірна Марківська модель відповідає розбиттю зображення на прямокутники. При цьому дозволені вертикальні і горизонтальні переходи між станами, але заборонені діагональні переходи, що спрощує модель. Відповідним чином відбувається сканування зображення. Для прихованої Марківської моделі важливе значення має початкова ініціалізація моделі. В якості початкової ініціалізації всіх моделей можуть бути використані всі зображення з набору тренування, коефіцієнт дискретного косинусного перетворення полягає в тому, що воно дозволяє працювати з стиснутими зображеннями.

Недоліком прихованих Марківських моделей є те, що алгоритм навчання тільки мінімізує відгук кожної моделі на свої класи, але не мінімізує на інші класи, тобто не виділяються ознаки, за якими один клас відрізняється від іншого. Таким чином, схожі класи може бути важко відрізнити. При збільшенні бази даних цей метод може виявитись не надійним.

2.5 Морфінгові моделі

Стандартні морфінгові моделі є методом, який базується на аналізі зображення для представлення і порівняння класів об'єктів. Для кожного з об'єктів, що

розпізнається будується морфінгова модель, яка дозволяє порівняти нове зображення з цього класу об'єктів.

Отримані параметри моделі можна використати для багатьох задач аналізу зображень таких, як верифікація об'єктів, розпізнавання облич та аналіз виразів обличчя. Однією з проблем, пов'язаних з морфінговими моделями є кількість прикладів зображень, необхідних для описання класу об'єктів.

Існують ієрархічні морфінгові моделі, в яких модель будується не для всього обличчя, а для певних його частин. Обличчя можуть моделюватись за допомогою форм і текстур та у вигляді трьох вимірної решітки. При заданні морфінгової моделі для певного класу об'єктів, вона навчається на множині облич з навчальної вибірки.

Перевагою ієрархічної морфінгової моделі є інваріантність до фону, освітлення, положення голови у просторі, міміки обличчя, а недоліком – відсутність інваріантності до масштабу вхідного зображення.

2.6 Нейронномережевий метод

При навчанні нейронної мережі використовується вибірка лиць та вказується клас до якого вони відносяться. Зразок представляється у вигляді вектору значень ознак. При цьому сукупність всіх ознак повинна однозначно виділяти клас до якого відноситься екземпляр. Якщо ознак не достатньо, то мережа може співвідносити один і той же екземпляр з декількома класами, що призведе до помилки розпізнавання. Після навчання нейронної мережі, можна передавати в неї тестові дані та отримувати відповідь про належність до певного класу.

Висновки до розділу 2

До переваг методу еластичних графів можна віднести: інваріантність до масштабу, освітлення, часткового зашумленого, фону. До недоліків – низька гнучкість до емоцій обличчя і положення голови в просторі. Також труднощі даного

методу полягають у складності математичних обчислень, на різному обладнанні буде різна точність необхідності точної локалізації орієнтирів.

Перевагами методів на основі дерев рішень є швидкість побудови моделі та легка інтерпретація, для їх його застосування не потрібно розраховувати різні параметри розподілення вірогідності. Недоліками є нестабільність – результат, отриманий на одній вибірці може відрізнятись на інших даних. Недоліком методу прихованих Марківських моделей є те, що схожі класи важко відрізнити, при збільшені бази даних цей метод може виявитись не надійним.

Перевагою морфінгової моделі є інваріантність до фону, освітлення, положення голови у просторі, міміки обличчя, а недоліком – відсутність інваріантності до масштабу вхідного зображення.

Усі представлені методи розпізнавання обличчя мають свої недоліки, найпоширеніші з них це погана якість розпізнавання, якщо голова повернута під певним кутом, умови освітлення відмінні від тих, що були у вибірці тренування та не однорідний фон. За допомогою використання нейронномережевого методу ці недоліки можна виправити.

3. НЕЙРОННІ МЕРЕЖІ

3.1 Загальні відомості про нейронні мережі

Існує багато проблем, які не можуть бути сформульовані, як алгоритм. До таких проблем відносяться ті, які залежать від багатьох факторів, що не є стійкими. Проте, людський мозок, хоча і приблизно, може вирішити такі завдання. Комп'ютер без певного алгоритму не може робити подібне. У зв'язку з цим з'явилась ідея створення штучної моделі, що буде працювати та навчатись подібно до людського мозку.

Розвиток штучних нейронних мереж надихається біологією. Тобто, розглядаючи мережеві конфігурації і алгоритми, дослідники застосовують терміни, запозичені з принципів організації мозкової діяльності. Але на цьому аналогія закінчується. Знання про роботу мозку настільки обмежені, що мало б знайшлося точно доведених закономірностей для тих, хто побажав би керуватися ними. Тому розробникам мереж доводиться виходити за межі сучасних біологічних знань в пошуках структур, здатних виконувати корисні функції.

Якщо порівняти комп'ютер та людський мозок, комп'ютер має певні підрозділи обробки та пам'ять, саме це дозволяє здійснювати складні обчислення за короткий проміжок часу, але головною проблемою є те, що вони не адаптуються, як це робить людський мозок. Теоретично комп'ютер є більш потужним, ніж людський мозок, більша частина якого працює безперервно, тоді як основна частина комп'ютера це пасивне сховище інформації, яке є статичним. Людський мозок, як біологічна нейронна мережа здатний навчатись і виявляти помилки.

Штучні нейронні мережі – це математичні моделі, створення яких було мотивоване їх подібністю до біологічних систем, які складаються з великої кількості нейронних клітин, які працюють паралельно і що є найбільш важливим – можуть навчатись. Як результат процесу навчання, нейронні мережі можуть узагальнювати та знаходити асоціації даних. Після навчання нейронна мережа може знаходити

рішення для подібних проблем, класу, який безпосередньо не був використаний під час навчання. Це формує стійкість нейронних мереж до не повних або не чітких вхідних даних, а особливо погано формалізованих задач.

Штучні нейронні мережі в більшості застосовуються в тих сферах, в яких людський інтелект має низьку ефективність, наприклад, коли обчислення є дуже великими, або якщо вони погано відображають реальні фізичні процеси та об'єкти. Нейронні мережі широко застосовуються для автоматизації процесу розпізнавання, прогнозування, класифікації, прийняття рішень, кодування, декодування, апроксимації залежностей, управління та багато інших.

З використанням нейронних мереж вирішується проблема проектування і оптимізації мереж зв'язку, тобто знаходження оптимального шляху трафіку між вузлами, а також, для отримання ефективних рішень в області їх проектування.

Ще однією важливою сферою застосування нейронних мереж є розпізнавання мови та управління цінами, адже ці процеси є складними, вони залежать від багатьох факторів. Наприклад, ціна може залежати від сезону та курсу валют, залежно від цього потрібно правильно обрати обсяг випуску продукції, нейронна мережа виявляє складні залежності між усіма факторами та виявляє оптимальне рішення поставленої задачі. Також нейронні мережі застосовуються в маркетингу для аналізу ринку, коли класичні методи прогнозування не є точними, в такому випадку, застосовується нейронна мережа з адаптивною архітектурою нейросимулятора. В сфері бізнесу нейронні мережі є важливим інструментом, який дозволяє зберегти бізнес компанії в умовах конкуренції, великі компанії проводять регулярні опитування покупців, аналіз таких опитувань є комплексною задачею, через присутність багатьох корельованих параметрів. Нейронна мережа допомагає виявити фактори, які визначають попит на продукцію та прогнозують поведінку покупців при зміні маркетингової політики компанії.

Однією з найважливіших сфер застосування штучних нейронних мереж є сфера медицини. Це застосування полягає здебільшого в діагностиці хвороб та прогнозуванні їх розвитку. Системи на основі нейронних мереж здатні зробити це точно та швидко, без участі кваліфікованого персоналу.

Біологічна нейронна мережа складається з сукупності нейронів, які з'єднані спеціальними нейронними волокнами, саме вони передають імпульси між нейронами. Біологічний нейрон – основна частина біологічної нейронної мережі, складається з ядра, дендритів (відростки нервових волокон), аксонів та синапсів. По дендритах приймаються імпульси, по аксону нейрон передає імпульс. Аксон контактує з дендритами інших нейронів використовуючи синапси, саме вони впливають на силу імпульсу, що потрібно передати.

3.2 Модель штучного нейрону

Структурою одиницею штучної нейронної мережі є штучний нейрон, він являється аналогом біологічного нейрону. З математичної точки зору, він є суматором. Кожен нейрон характеризується своїм поточним станом, він має групу входів, які з'єднані з виходами інших нейронів. На вхід мережі подаються числа, які символізують величину сигналу, вони пересуваються мережею та змінюються.

В біологічній нейронній мережі саме синапси посилюють або послаблюють сигнал, який проходить по ній, у штучній нейронній мережі цю функцію виконують ваги зв'язків між нейронами. Кожен нейрон містить певну кількість входів (n), через які він приймає сигнал, ваги, на які множаться сигнали, що проходять по зв'язку ($n+1$) та суматор з нелінійним перетворювачем. Сигнал, який пройшов через зв'язок множиться на вагу відповідного зв'язку. Сигнал першого входу x_i множиться на відповідну для даного входу вагу w_i . Сигнали помножені на вагу з кожного входу нейронної мережі передаються в суматор, головною функцією якого є сумування всіх $x_i w_i$ у мережі (Рисунок 3.1 – Структура нейрона).

Математичне представлення роботи суматора:

$x_1 w_1 + x_2 w_2 + \dots + x_n w_n = \sum_{i=1}^n x_i w_i$, де x_i – вхідні значення нейрону ($x_i \in \{0, 1\}$, $x_0=1$), w_i – ваговий коефіцієнт, міра, які визначає вплив вхідного коефіцієнта на стан нейрону, може бути змінений залежно від архітектури та правил навчання [12 – 14].

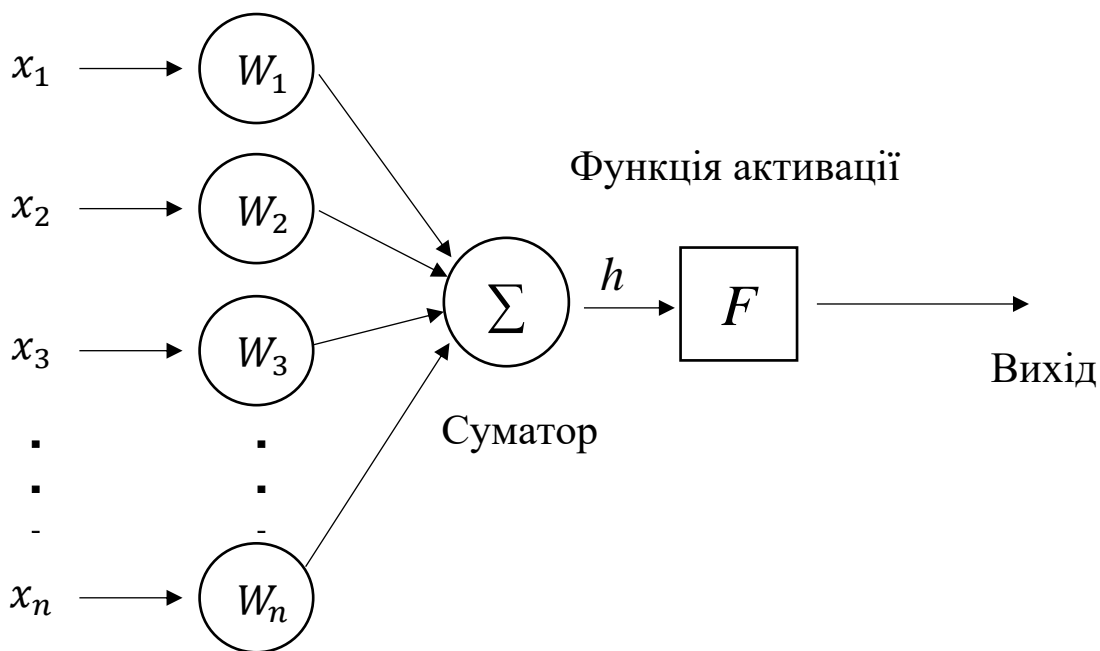


Рисунок 3.1 – Структура нейрона

Результатом роботи суматора є число, яке має назву зважена сума, тобто сума всіх сигналів, помножених на відповідні ваги. Головною функцією суматора є агрегація всіх вхідних сигналів в одне число h , $h = \sum_{i=1}^n x_i w_i$.

3.3 Функція активації

Пряма подача зваженої суми на вихід не є раціональною, адже нейрон повинен в результаті її обробки сформувати вхідний сигнал, саме для цього використовується функція активації. Це функція, яка перетворює зважену суму в певне число, яке є виходом нейрону. Для різних типів нейронів використовуються різні функції активації, вони приймають зважену суму в якості параметру: $act = \varphi(h)$.

Типи функцій активації

Логістична функція. Для активації в штучних нейронних мережах часто використовується логістична функція (Рисунок 3.2 – Графік логістичної функції).

Аналітичний опис функції: $act(h) = \frac{1}{1+\exp(-a*h)}$, де a – це параметр, який характеризує степінь крутизни функції.

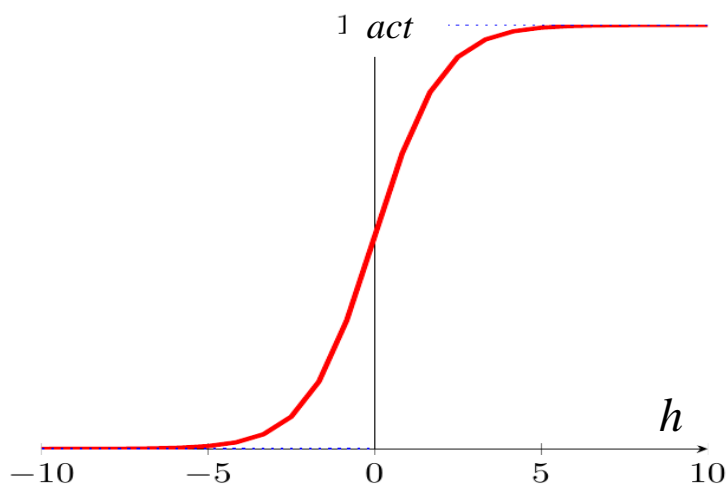


Рисунок 3.2 – Графік логістичної функції

При використанні логістичної функції, не залежно від зваженої суми, вхідний сигнал завжди буде в межах від 0 до 1, вона є гнучкою, адже її значення може бути будь-яким в межах даного діапазону чисел. В всіх точках логістична функція має похідну, ця похідна може бути виражена через ту саму функцію.

Через свої недоліки, дана функція використовується доволі рідко. Насиченість сигмоїди призводить до затухання градієнтів, тобто при насичені функції з тієї чи іншої сторін, градієнт на цих ділянках набуває значення близькому до нуля. В процесі зворотного розповсюдження помилки, локальний градієнт множиться на загальний градієнт, тобто, якщо локальний градієнт має дуже мале значення, від фактично обнуляє загальний градієнт. Таким чином, сигнал майже не буде проходити через нейрон і його даних. Також є складнощі при ініціалізації ваг сигмоїдних нейронів, вони пов'язані з ризиком насичення, якщо більшість нейронів перейде в стан перенасичення, то штучна нейронна мережа буде погано навчатись. Ще одним недоліком сигмоїдної функції є те, що вихід сигмоїди не є центрованим відносно нуля, таким чином, нейрони на наступних шарах будуть отримувати значення, які не є центрованими відносно нуля, що вплине на градієнтний спуск. Якщо значення, які поступають в нейрон, завжди позитивні, то в процесі зворотного поширення похибки

всі градієнти ваг будуть позитивними або негативними. Це може призвести до негативної динаміки оновлення ваг.

Гіперболічний тангенс. Графік функції гіперболічного тангенсу є доволі схожим на графік логістичної функції (Рисунок 3.3 – Графік гіперболічного тангенсу).

Аналітичний опис функції: $act(h) = \tanh \frac{h}{a}$, де a – це параметр, який характеризує степінь крутизни функції. Подібно до сигмоїди, гіперболічний тангенс може насичуватись, але на відміну від неї, вхід даної функції є центрованим відносно нуля.

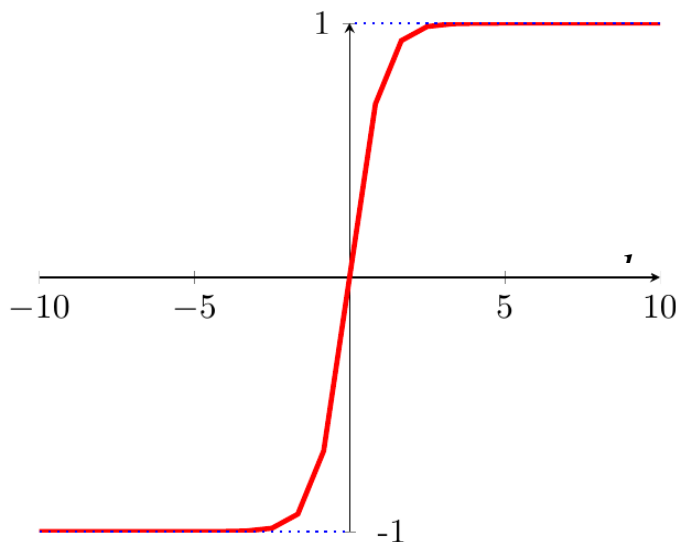


Рисунок 3.3 – Графік гіперболічного тангенсу

Ще одна відмінність даної функції від логарифмічної полягає в тому, що область її значень лежить в інтервалі $[-1;1]$.

Функція одиничного скачка. Це проста кусково-лінійна функція. Її суть полягає в тому, що вхід нейрону може дорівнювати тільки 0 або 1, якщо зважена сума більша, ніж певний поріг k , то вхід нейрону дорівнює 1, а якщо нижче – 0 (Рисунок 3.4 – Графік функції одиничного скачка). На горизонтальній осі розміщуються зважені суми, а на вертикальній – значення вхідного сигналу, його значення можуть бути 1 або 0.

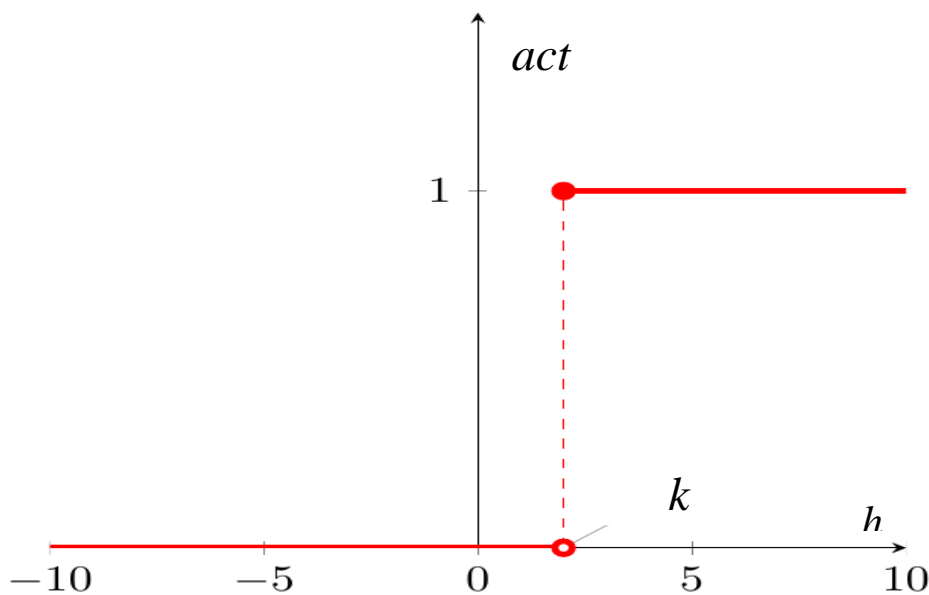


Рисунок 3.4 – Графік функції одиничного скачка

Аналітичний опис функції: $act(h) = \begin{cases} 0, & h < k \\ 1, & h \geq k \end{cases}$, де k – заданий поріг. Таким

чином, вихід нейрона є залежним від зваженої суми. Такий тип функції застосовується у бінарних системах, наприклад, для вирішення задачі бінарної класифікації.

Rectified linear unit є найчастіше використовуваною функцією активації. Її безумовною перевагою є проста умова. Нейрони з даною функцією активації мають назву ReLU (Рисунок 3.5 – Графік ReLU функції). Аналітичний опис функції: $act(h) = \max(0, h)$. Дана функція реалізує простий пороговий перехід в нулі. Проста функція, яка складається з двох лінійних частин, дозволяє моделі враховувати також і нелінійні ітерації.

Обчислення сигмоїди та гіперболічного тангенса потребують ресурсозатратних операцій, в той час, як функція ReLU може бути реалізована за допомогою порогового перетворення матриці активацій в нулі, також вона не є підверженою насиченню. Її використання суттєво підвищує швидкість сходимості градієнтного спуску в порівнянні з іншими функціями активації, що застосовуються в штучних нейронних

мережах, це обумовлено тим, що вона має лінійний характер та відсутністю її насичення.

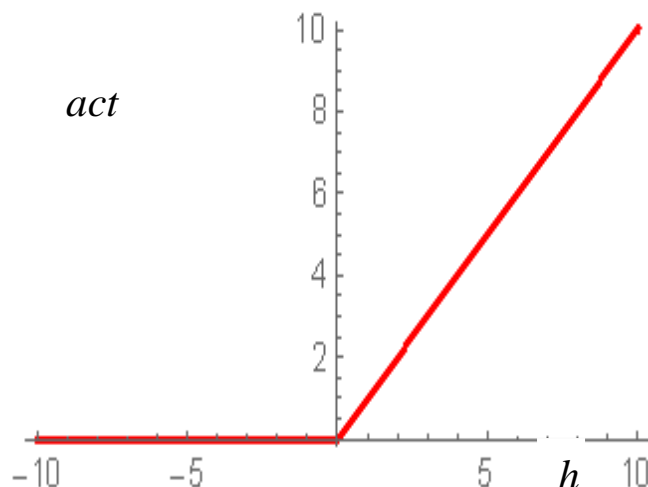


Рисунок 3.5 – Графік ReLU функції

Проте, функція ReLU також має і негативні сторони. Вона не завжди є достатньо надійною, наприклад, великий градієнт, який проходить через ReLU, може призвести до такого обнуління ваг, яке призведе до того, що поточний нейрон більше не активується, в результаті цього, градієнт, який в подальшому буде проходити через цей нейрон буде рівним нулю, тобто такий нейрон буде виведено із строю. Ця проблема є актуальною при дуже високій швидкості навчання, вона вирішується підбором найбільш оптимальної швидкості навчання для вирішення конкретної задачі, з урахуванням особливостей функції активації.

Не зважаючи на свої недоліки, функція ReLU зіграла важливу роль у вирішенні важливої проблеми багатошарового навчання глибоких нейронних мереж – експоненціального затухання зміни ваг на глибоких шарах при навчанні штучної нейронної мережі методом зворотного розповсюдження помилки. Позитивна роль цієї функції активації проявилась в тому, що при $h > 0$, градієнт ReLU всюди дорівнює 1, саме це дозволило перейти від пошарового навчання без вчителя мереж до монолітного навчання мереж, на класифікованих даних. Особливість ReLU, яка полягає в тому, що дана функція активації недиференційована в нулі, на практиці не є суттєвим недоліком.

Існують такі модифікації ReLU (Рисунок 5 – Графіки модифікацій ReLU):

Leaky ReLU – представляє собою спробу вирішити проблему виходу зі строю звичайних ReLU, в наслідок високої швидкості навчання штучної нейронної мережі. Звичайна функція активації ReLU на інтервалі $h < 0$ дає на виході нуль, тоді, як LReLU має на цьому інтервалі невелике негативне значення, тобто функція LReLU має вигляд: $f(h) = ah$, при $x < 0$ і $f(h) = h$, при $h \geq 0$, де a – мала константа. Є відомості про позитивний досвід застосування даної функції, проте, результати її застосування є не завжди стабільними.

Parametric ReLU – для параметричного рівняння ReLU кутовий коефіцієнт на негативному інтервалі попередньо не задається, а визначається на основі даних. Автори даної модифікації ReLU стверджують, що її застосування дозволило інформаційним технологіям перевищити рівень людини в задачі розпізнавання зображень. Процес зворотного поширення похибки і оновлення для PReLU є подібним до ReLU.

Randomized ReLU – кутовий коефіцієнт на негативному інтервалі під час навчання генерується випадковим чином, в межах заданого інтервалу, а під час тестування – залишається постійним. Авторам даного методу вдалось зменшити перенавчання завдяки елементу випадковості, що притаманний для RReLU.

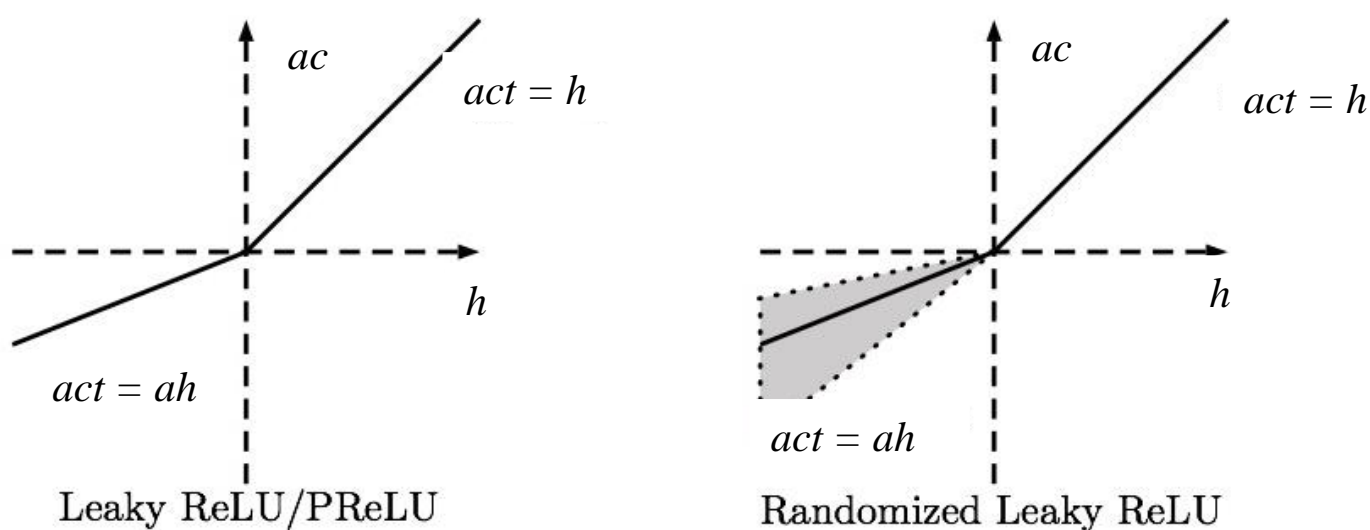


Рисунок 3.6 – Графіки модифікацій ReLU

Можна зробити висновок, що всі модифікації ReLU суттєво перевершили традиційні функції активації, що застосовуються у штучних нейронних мережах. У випадку Leaky ReLU коефіцієнт a – забезпечує більш високу точність, хоча й Parametric ReLU є схильною до перенавчання на малих наборах даних, ця модифікована версія все ж перевершує традиційну ReLU функцію. Randomized ReLU дозволяє уникнути перенавчання.

3.4 Види навчання нейронних мереж

Навчання нейронної мережі – це певне налаштування архітектури і ваг зв'язків між нейронами для ефективного виконання поставлених задач. Методи навчання штучних нейронних мереж поділяються на два типи, стохастичні та детерміновані методи [15]. Детерміновані методи – ті, які базуються на ітеративній корекції параметрів штучної нейронної мережі в межах поточної ітерації, яка бере за основу поточні параметри, найпопулярнішим з цієї групи методів є метод зворотного поширення похибки (backpropagation). Стохастичні – методи, які змінюють параметри штучної нейронної мережі випадковим чином і зберігають лише ті параметри, які спричинили покращення результатів. Вони реалізовані за допомогою порівняння похибок та виявленню локального мінімуму.

Метод навчання з вчителем. Під час навчання штучної нейронної мережі кожному прикладу з навчальної вибірки відповідає певний вектор, який характеризує однозначну правильну відповідь, яка має податись на вихід мережі в обхід всієї її архітектури. Після отримання відповідного результату мережі, алгоритм порівнює результуючий вектор з правильною відповіддю, на основі цього відбувається подальша корекція похибки.

Метод навчання без вчителя. Навчання без вчителя у штучних нейронних мережах реалізується в процесі навчання, коли автоматичне налаштування параметрів мережі призводить до появи однакових результатів її функціонування при достатньо близьких результатах її функціонування при достатньо близьких вхідних значеннях. На практиці це можна порівняти з пониженням розмірності даних в

результаті ітераційного методу головних компонент. Правило Хебба, яке базується на гіпотезі про посилення зв'язків між біологічними нейронами у випадку їх одночасного збудження і методи навчання при конкуренції нейронів шляхом порівняння сили їх реакції є прикладами навчання без вчителя.

Метод навчання зі зворотнім поширенням помилки. Якщо в нейронній мережі лише один шар, а алгоритм її навчання з вчителем є відомим, то зважаючи на те, що вхідні стани нейронів є відомими і налаштування синаптичних зв'язків відбувається в напрямку, який мінімізує помилку на виході мережі. По цьому принципу будується алгоритм навчання одношарового перцептрону [16 – 18]. В багатшарових штучних нейронних мережах оптимальні вхідні значення нейронів усіх шарів, за винятком останнього, як правило, не відомі. Таким чином, багатшаровий перцептрон вже не можна навчити використовуючи лише значенням помилок на виході нейронної мережі. Мережі прямого поширення – це клас мереж, у яких є зв'язки від певного шару до входів того ж шару чи попередніх шарів. У мереж без зворотного поширення немає пам'яті, бо їх вхід цілком визначається поточними входами і значеннями ваг [19 – 21].

Один із варіантів вирішення цієї проблеми – розробка наборів вихідних сигналів, які відповідають вхідним, для кожного шару штучної нейронної мережі або динамічне налаштування вагових коефіцієнтів синапсів, під час якої, найслабші зв'язки змінюються, а зберігаються лише значення, які сприяли зміні похибки на виході нейронної мережі. Ще один варіант – розповсюдження сигналів похибки від виходів нейронної мережі до її входів, в напрямку, зворотному прямому поширенню сигналів. Таки метод роботи нейронної мережі є методом зворотного поширення похибки (Рисунок 3.7 – Схема зворотного поширення похибки).

Згідно методу найменших квадратів, мінімізуючою цільовою функцією є величина $E(w)$, яка визначаються за формулою:

$$E(w) = \frac{1}{2} \sum_{j,p} (y_{j,p}^{(N)} - d_{j,p})^2,$$

де $y_{j,p}^{(N)}$ – реальний вхідний стан нейрона j вхідного шару N нейронної мережі при подачі на її виходи p -того образу; $d_{j,p}$ – бажаний вхідний стан цього нейрона.

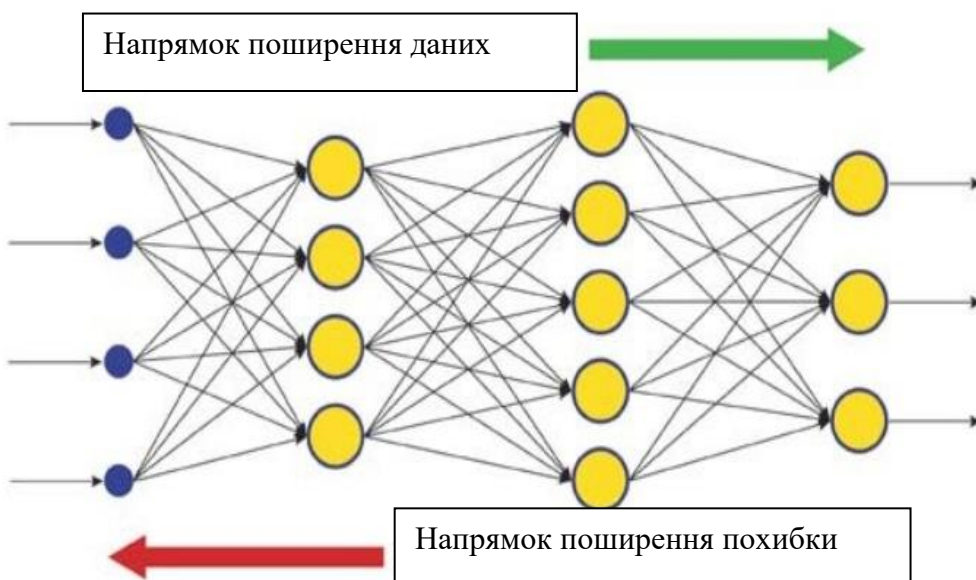


Рисунок 3.7 – Схема зворотного поширення похибки

Сумування відбувається по всіх нейронах вихідного шару і по всіх образах, що обробляються штучною нейронною мережею. Мінімізація відбувається методом градієнтного спуску. Коефіцієнти визначаються за формулою:

$$\Delta w_{ij}^{(n)} = -\eta \frac{\partial E}{\partial w_{ij}},$$

де w_{ij} – ваговий коефіцієнт синаптичного зв'язку, який поєднує i -тий нейрон n -того шару нейронної мережі, η – коефіцієнт швидкості навчання, $0 < \eta < 1$.

$$\frac{\partial E}{\partial w_{ij}} = \frac{\partial E}{\partial y_j} * \frac{\partial y_j}{\partial s_j} * \frac{\partial s_j}{\partial w_{ij}},$$

де y_j – вихід нейрона j , s_j – зважена сума його вхідних сигналів, тобто аргумент функції активації, $\frac{\partial y_j}{\partial s_j}$ – похідна функції активації по її аргументу. З цього слідує, що похідна активаційної функції повинна бути визначена на всій осі абсцис. У зв'язку з цим, активаційні функції з неоднорідностями не можуть застосовуватись до даного типу штучних нейронних мереж. В них застосовуються такі гладкі функції, як гіперболічний тангенс та сигмоїда [22 – 24].

Повний алгоритм навчання штучної нейронної мережі складається з таких п'яти кроків:

1. Подавання на виходи мережі один із можливих образів i в режимі звичайного функціонування штучної нейронної мережі, коли сигнали поширюються від входів до виходів, розрахувати значення виходів використовуючи формулу:

$$S_j^{(n)} = \sum_{i=0}^M y_i^{(n-1)} * w_{ij}^{(n)},$$

де M – кількість нейронів в шарі $n-1$ з врахуванням нейрона з постійним вихідним станом $+1$, який задає зміщення; $y_i^{(n-1)} = x_{ij}^{(n)}$ – i -тий шар нейрону j шару n . $y_i^{(n)} = f(s_j^{(n)})$, де $f()$ – сигмоїд, $y_q^{(0)} = I_q$, де I_q – компонента вектору вхідного образу.

2. Розрахувати $\delta^{(N)}$ для вихідного шару за формулою:

$$\delta_l^{(N)} = (y_l^{(N)} - d_l) \frac{\partial y_l}{\partial s_l}$$

Розрахувати зміну ваг $\Delta w^{(n)}$ шару N за формулою:

$$\Delta w_{ij}^{(n)} = -\eta \delta_j^{(n)} * y_i^{(n-1)}$$

3. Розрахувати $\Delta w^{(n)}$ та $\delta_l^{(n)}$ для всіх шарів штучної нейронної мережі.

4. Скорегувати всі ваги в штучній нейронній мережі використовуючи формулу:

$$w_{ij}^{(n)}(t) = w_{ij}^{(n)}(t-1) + \Delta w_{ij}^{(n)}(t)$$

5. Якщо помилка мережі суттєва, то перейти на крок 1, а якщо ні – цикл завершено.

В процесі навчання може виникнути ситуація, коли великі значення коефіцієнтів ваг змістять робочу точку на сигмоїдах багатьох нейронів в область насичення. Малі величини похідної від логістичної функції призведуть до зупинки навчання. Також застосування методу градієнтного спуску не гарантує, що буде знайдено саме глобальний, а не локальний мінімум цільової функції. Ця проблема зв'язана ще з однією проблемою вибору величини швидкості навчання. Доведення сходимості навчання в процесі зворотного розповсюдження базується на похідних,

тобто збільшення ваг і швидкості навчання можуть призвести до постійної нестабільності процесу навчання. Для уникнення випадкових попадань в локальні мінімуми після стабілізування значень вагових коефіцієнтів, коефіцієнт швидкості навчання короткочасно швидко збільшують, з метою початку градієнтного спуску з нової точки. Якщо повторення цієї процедури приведе алгоритм в один і той самий стан нейронної мережі, то це буде сигналізувати про те, що знайдено саме глобальний мінімум [24 – 28].

Алгоритм навчання нейронної мережі за допомогою процедури зворотного розповсюдження потребує наявності зовнішньої ланки, яка представляє штучній нейронній мережі не тільки вхідні дані, а й цільові образи [29 – 33].

Методи на основі генетичних алгоритмів. Генетичні алгоритми є різновидом еволюційних обчислень, які розвивались, як евристичні алгоритми пошуку. Подібні методи використовуються для рішення задач моделювання і оптимізації шляхом випадкового підбору, реалізуючи у вигляді алгоритму базові принципи теорії еволюції.

Генетичні алгоритми можна використовувати для навчання нейронних мереж, коли важко або неможливо сформулювати задачу у вигляді, який би міг підійти для більш швидких алгоритмів локальної оптимізації та у випадках багатоекстремальної оптимізації чи оптимізації недиференційованої функції.

Гіперпараметри – це всі межі варіацій алгоритму, які алгоритм не оптимізує напряму. Сучасні глибокі мережі реалізують різні методи налаштування гіперпараметрів, в основному це різні методи регуляції.

Dropout – простий і ефективний метод регуляції, який полягає в тому, що в процесі навчання мережі з її загальної топології багато разів випадковим чином обирається підмережа, і наступна зміна ваг відбувається тільки в межах обраної підмережі. Кожен нейрон виключається з мережі з певною вірогідністю, яка має назву dropout коефіцієнту.

Існують і інші методи регуляції, які базуються на обмеженні ваг, такі як обмеження норми вектора ваг, L1 і L2 регуляції.

3.5 Типи нейронних мереж

В своїй більшості, архітектури штучних нейронних мереж мають фіксовані функції активації нейронів. Ваги їх синапсів є параметрами мережі. Також в мережі є зовнішні входи та виходи. Деякі входи можуть бути виходами для інших шарів і навпаки.

Обчислення в штучній мережі полягають в перетворенні вхідного вектору у вихідний, це здійснюється за допомогою ваг, біасів та певної топології мережі. Для класифікації штучних нейронних мереж використовують різні ознаки [34, 35].

Штучні нейронні мережі можуть бути: повнозв'язними, тобто кожен нейрон поєднаний з нейронами наступного шару нейронної мережі; шаровими – коли нейрони згруповані в шари та є поєднаними один з одним; слабкозв'язними – тип штучної нейронної мережі, для якого характерне розміщення нейронів у вузлах прямокутної чи гексальної решітки.

Штучні нейронні мережі різних топологій застосовуються для вирішення різних задач, саме наявність багатьох типів мереж забезпечує їх широке використання у різних сферах для вирішення задач розпізнавання, прогнозування, класифікації, ідентифікації та багатьох інших [36, 37].

Одношарові нейронні мережі. Навіть один нейрон здатен виконати не важкі процедури розпізнавання, але головна особливість, що забезпечує нейронним мережам високу точність розпізнавань – поєднання нейронів в мережі. Проста нейронна мережа складається з групи нейронів, що згруповані в певний шар [38 – 43].

Вхідний шар (Input) служить лише для розподілення сигналів. Кожен елемент з множини входів X поєднаний за допомогою ваги з кожним штучним нейроном, а кожен штучний нейрон передає зважену суму входів в мережу, також можливі зв'язки між виходами і входами елементів шарів.

Багатошарові нейронні мережі. Відмінності в обчислювальних процесах нейронних мереж обумовлені характером взаємодії нейронів між шарами мережі. За сукупністю критеріїв багатошарові штучні нейронні мережі можна поділити на динамічні та статичні. Відповідно дані класи мають певні відмінності між собою.

Кожен з представлених класів архітектур багатошарових штучних нейронних мереж може включати в себе множину підкласів, таким чином реалізуючи різні підходи.

До статистичних архітектур можна віднести мережі прямого поширення, в яких реалізовано однонаправлений зв'язок між шарами, відсутні динамічні елементи та зворотній зв'язок, а вихід штучної нейронної мережі визначається однозначно входом та є не залежним від попередніх станів мережі. Прикладами статистичних багатошарових штучних нейронних мереж є: перцептрон, нейронна мережа Кохена, когнітрон, неокогнітрон, згорткова нейронна мережа [43 – 48].

Динамічні архітектури багатошарових штучних нейронних мереж є протилежністю статистичних, вони реалізують рекурентну структуру з використанням зворотних зав'язків, завдяки чому стан мережі в кожен момент часу є залежним від попереднього стану. Як правило, рекурентні штучні нейронні мережі базуються на багатошаровому перцептроні. Прикладами таких мереж є: нейронна мережа Хопфілда, нейронна мережа Коско, нейронна мережа Джордана, нейронна мережа Елмана.

Багатошарові нейронні мережі, як правило, мають більші обчислювальні можливості, ніж одношарові. Пошарова структура, в певній мірі, копіює шарові структури деяких відділів людського мозку.

Елементарний перцептрон організовується на основі сенсорних даних на вході, асоціативних елементів та реагуючих елементів на виході. Набір сенсорних елементів пов'язаних з асоціативним елементом організовує асоціацію. Асоціативний елемент активізується після досягнення певного числа сигналів від сенсорних елементів. Він передає зважений сигнал, сумуючий реагуючі елементи і в залежності від того чи перевищує зважена сума певний поріг, реагуючий елемент видає результат роботи перцептрона [48 - 53].

Багатошаровий перцептрон організовується за допомогою прихованих шарів асоціативних елементів, які розміщені між сенсорними та реагуючими елементами. Складність задач, які вирішуються за допомогою багатошарового перцептрону є найвищою для класу перцептронів. В одношаровому перцептроні вхідні елементи прямо поєднані з вихідними за допомогою ваг, зв'язки асоціативних та сенсорних

елементів організовані за принципом однозначної відповідності. Одношаровий перцептрон є випадком класичного елементарного перцептрону, найпростішою мережею прямого поширення – лінійним класифікатором і має свої обмеження.

Багатошарові нейронні мережі можуть утворюватися каскадами шарів. Вхід одного шару є виходом для наступного (Рисунок 3.8 – Схема багатошарової нейронної мережі). Багатошарові мережі не можуть підвищити обчислювальні можливості лише в тому випадку, коли функція між шарами є нелінійною. Обчислення виходу шару полягає у множенні вхідного вектору на першу матрицю ваг та подальшим множенням, якщо відсутня нелінійна функція активації.

Двошарова лінійна мережа еквівалентна одному шару з ваговою матрицею, рівною двом перемноженим ваговим матрицям. Отже будь-яка багатошарова нейронна мережа може бути замінена еквівалентною одношаровою мережею, проте, одношарові мережі є доволі лімітованими в своїх обчислювальних можливостях. Тобто для того, щоб мати ширші можливості мереж потрібна функція активації.

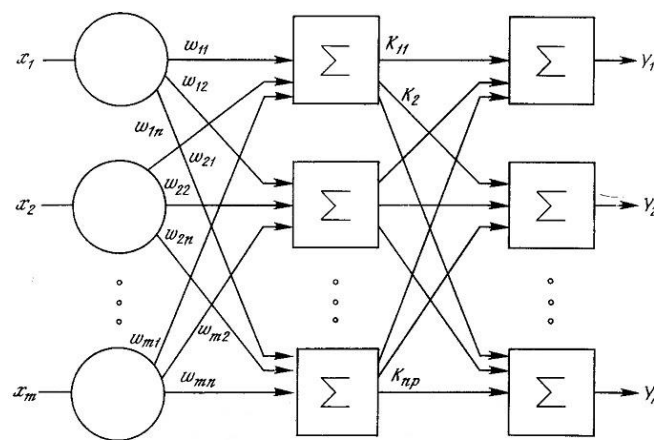


Рисунок 3.8 – Схема багатошарової нейронної мережі

Нейронна мережа Кохена. Нейронна мережа Кохена представляє собою мережу, яка базується на спеціальних шарах Кохена. Вони, як і звичайні шари нейронних мереж складаються з звичайних нейронів, які в свою чергу, виконують функцію суматорів.

Нейронна мережа Кохена – це типовий приклад нейронномережевої архітектури, яка навчається без вчителя. Зазвичай нейронні мережі навчаються на

вибірках даних, що відносяться до певного класу та складаються з відповідних один одному пар вхідних та вихідних векторів, вхідні значення приймають участь в налаштуванні вагових коефіцієнтів. В нейронних мережах Кохена вихідні вектори навчальної вибірки можуть бути відсутні, вони не приймають безпосередньої участі в навчанні, виходи не використовуються при корекції синапсів [54].

В нейронній мережі Кохена сигнал розповсюджується від входів до виходів в прямому напрямку. Структура нейронної мережі містить лише один шар нейронів, який не містить коефіцієнти зміщення (Рисунок 3.9 – Схема нейронної мережі Кохена). Кількість нейронів є рівною кількості кластерів, серед яких відбувається початкове розподілення і подальше перерозподілення навчальних прикладів. Кількість вхідних змінних дорівнює кількості ознак, якими характеризується об'єкт [55 – 60].

Важливо розрізнити самонавчання та самоорганізацію нейронної мережі Кохена. При самонавчанні мережа має фіксовану кількість нейронів, які не змінюються, при самоорганізації – мережа не має постійної структури. В залежності від знайденої відстані до нейрона-переможця цей нейрон використовується для кластеризації прикладу або для вхідного прикладу створюється новий кластер з відповідними йому ваговими коефіцієнтам. В процесі самоорганізації мережі Кохена деякі нейрони можуть бути виключені з неї.

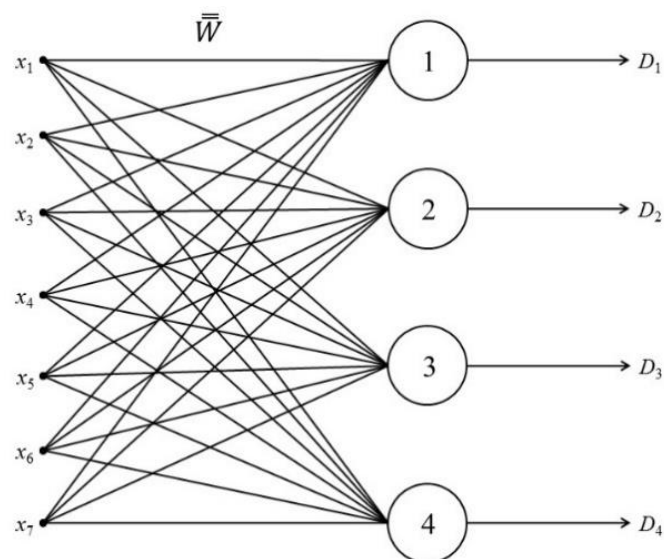


Рисунок 3.9 – Схема нейронної мережі Кохена

Нормалізація вхідних змінних виконується в межах $[-1;1]$ або $[0;1]$. Для циклу нейронних мереж Кохена притаманні такі життєві етапи, як: навчання, кластерний аналіз, практичне використання. Алгоритм навчання даної нейронної мережі залежить від типу структури – мережа самонавчання чи самоорганізації.

Для самонавчання послідовно виконуються такі п'ять кроків:

1. Задання кількості нейронних шарів Кохена.
2. Випадкова ініціалізація вагових коефіцієнтів.
3. Подання на входи мережі випадкового прикладу навчання поточної епохи навчання і розрахунок евклідових відстаней від входу до центрів кластерів.
4. За найменшим значенням евклідової відстані від вхідного вектора до центрів кластерів обирається нейрон-переможець, який найближчий за значенням до вхідного вектора. Далі виконується корекція вагових коефіцієнтів.
5. Виконується цикл починаючи з третього кроку, поки не буде досягнута хоча б

одна з наступних умов завершення циклу: досягнута максимальна кількість епох навчання; не відбулось вагомих змін вагових коефіцієнтів в межах заданої точності на протязі всієї епохи навчання; було досягнуто граничний час навчання.

У випадку самонавчання алгоритм складається з наступних п'яти кроків:

1. Задається критична відстань $R_{кр}$, яка відповідає максимально допустимій евклідовій відстані між входами прикладу і вагами нейрона-переможця. Початкова структура не містить нейронів, при подаванні на входи мережі першого прикладу навчальної збірки створюється перший нейрон с ваговими коефіцієнтами, які дорівнюють вхідним значенням.

2. На входи мережі подається новий випадково обраний приклад поточної епохи навчання, розраховуються евклідові відстані від прикладу до центру кожного кластера і визначається нейрон-переможець з найменшим R_{min} .

3. Якщо виконується умова $R_{кр} \leq R_{min}$, то робиться корекція вагових коефіцієнтів

нейрона-переможця, якщо умова не виконується – додається новий нейрон, вагові коефіцієнти якого є рівними вхідним значенням поданого прикладу.

4. Повторюється цикл з пункту 2, якщо на протязі епохи якісь кластери не були задіяні, то відповідні нейрони виключаються з мережі.

5. Цикл завершується, якщо: досягнута максимальна кількість епох навчання;
не

відбулось вагомих змін вагових коефіцієнтів в межах заданої точності на протязі всієї епохи навчання; було досягнуто граничний час навчання [60, 61].

Практичне використання штучної нейронної мережі Кохена полегшується за рахунок візуалізації результатів кластеризації. В результаті самонавчання мережі отримується набір кластерів, кожен з яких характеризується центром і кількістю прикладів навчання, які його сформували. Обчислюється евклідова відстань між центрами всіх можливих пар кластерів і зображається графічно на карті Кохена – двовимірній графічній структурі, яка дозволяє аналізувати розміщення кластерів один відносно одного.

Нейронна мережа Хебба. Недоліки звичайних перцептронів можна уникнути за допомогою використання багатошарових нейронних мереж та введенням в мережу зв'язків, які допускають контурну циркуляцію сигналів. Такий підхід лежить в основі нейронної мережі Хеба.

Згідно поглядів Хебба, нервові клітини мозку поєднані між собою за допомогою великої кількості прямих та зворотних збуджуючих зв'язків. Кожен нейрон здійснює просторово-часове сумування сигналів, що приходять до нього. Ефективність зв'язків може бути змінена в процесі функціонування мережі, підвищуючись між одночасно збудженими нейронами, це призводить до поєднання нейронів у групи клітин, які разом збуджуються.

Саме ці ідеї Хебба полягли в основу створення нейронних мереж Хебба. Нейронну мережу Хебба можна розглядати, як структуру, в якій реалізована функція розподіленої асоціативної пам'яті. Формування асамблей в мережі такого типу відповідають запам'ятанням образів, які є закодованими патерном активності

нейронів. Сформовані ансамблі є внутрішнім представленням (Рисунок 10 – Схема нейронної мережі Хебба).

Модель пам'яті, яка базується на ансамблевій нейронній мережі володіє асоціативністю, розподіленістю, паралельністю та стійкістю до шумів та збоїв і надійність. Сигнальний метод Хебба полягає у обчисленні ваг використовуючи формулу:

$$w_{ij}(t) = w_{ij}(t - 1) + \alpha * y_i^{(n-1)} * y_j^{(n)},$$

де $y_i^{(n-1)}$ – вхідне значення нейрону i шару $(n - 1)$; $y_j^{(n)}$ – вихідне значення нейрона j шару; $w_{ij}(t)$ і $w_{ij}(t - 1)$ – ваговий коефіцієнт синапсу, який поєднує ці нейрони; α – коефіцієнт швидкості навчання нейронної мережі; n – довільний шар мережі.

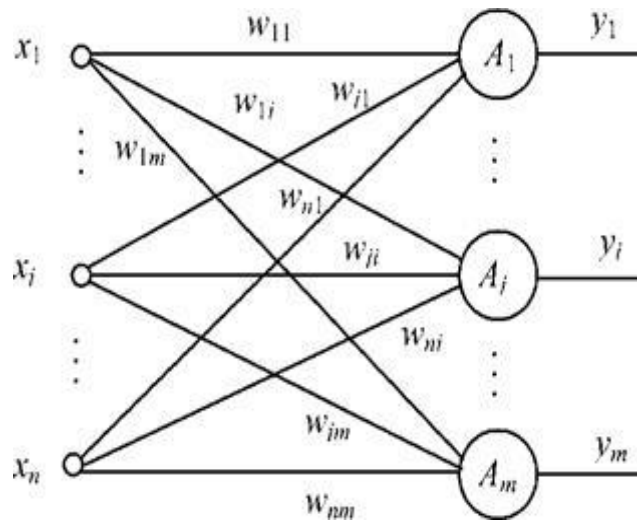


Рисунок 3.10 – Схема нейронної мережі Хебба

Існує диференційний метод навчання Хебба. Математична модель цього методу:

$$w_{ij}(t) = w_{ij}(t - 1) + \alpha * [y_i^{(n-1)}(t) - y_i^{(n-1)}(t - 1)] * [y_j^{(n)}(t) - y_j^{(n)}(t - 1)],$$

де $y_i^{(n-1)}(t)$ і $y_i^{(n-1)}(t - 1)$ – вхідні значення нейрона i шару $n-1$ на ітераціях t та $(t - 1)$ відповідно; $y_j^{(n)}(t)$ і $y_j^{(n)}(t - 1)$ – вхідні значення нейрона j шару $n-1$ на ітераціях t та $(t - 1)$ відповідно.

Алгоритм навчання нейронної мережі Хебба складається з наступних п'яти кроків:

1. На стадії ініціалізації всім ваговим коефіцієнтам присвоюються малі випадкові

значення.

2. На входи мережі подається вхідний образ і сигнали розподіляються по шарах,

для кожного нейрона розраховується зважена сума його входів, а після цього до неї застосовується активаційна функція.

3. На основі отриманих вхідних значень нейронів відбувається зміна ваг коефіцієнтів.

4. Відбувається перехід на другий крок, поки вхідні значення мережі не будуть стабілізовані з заданою точністю.

Застосування способу завершення навчання дозволяє уникнути обмеженості налаштування значень синапсів. На другому кроці алгоритму поступово представляються всі образи вхідного набору. Вид відгуків на кожен клас вхідних образів не відомий і представляє собою довільне поєднання станів нейронів вхідного шару, обумовлене випадковим розподіленням ваг на стадії ініціалізації. Разом з тим, мережа здатна розпізнавати подібні образи, які відносяться до одного класу. Тестування навченої мережі дозволяє визначити топологію класів вхідного шару. Щоб привести відгуки навченої мережі до зручного представлення можна доповнити мережу ще одним шаром.

Нейронна мережа Хопфілда. Нейронна мережа Хопфілда представляє собою штучну нейронну мережу, яка містить один рекурентний шар з кроковою функцією активації та не містить входів. Даний тип нейронної мережі не можна віднести ні до типу навчання з вчителем, ні до типу навчання без вчителя. Вагові коефіцієнти розраховуються тільки один раз перед початком функціонування мережі на основі даних, що обробляються і все навчання мережі зводиться до цього розрахунку. Мережа фактично запам'ятовує зразки до потрапляння реальних даних і не може змінювати свою поведінку.

Структура мережі Хопфілда складається з одного шару нейронів, кількість яких одночасно є кількістю входів і виходів (Рисунок 3.11 – Схема нейронної мережі Хопфілда). Кожен нейрон поєднаний з іншими нейронами і має один вхідний синапс, через який здійснюється вхід сигналу. Вхідні сигнали створюються на аксонах.

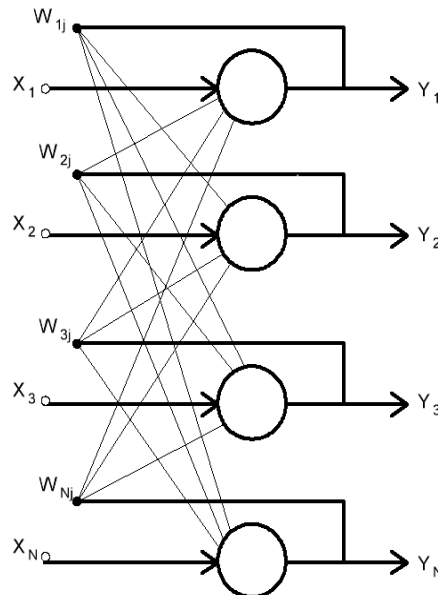


Рисунок 3.11 – Схема нейронної мережі Хопфілда

Нейронна мережа складається з певної кількості нейронів, кожен з яких може набувати один із двох станів. Нейрони можуть взаємодіяти один з одним згідно виразу:

$$E = \frac{1}{2} \sum_{i,j=1}^N w_{ij} x_i x_j,$$

де w_{ij} – елемент матриці взаємодій, яка складається з вагових коефіцієнтів зв'язків між нейронами, в неї в процесі навчання записуються образи, які представлені бінарними векторами.

В архітектурі мережі Хопфілда матриця зав'язків є симетричною, а діагональні елементи матриці дорівнюють нулю, це допомагає виключити ефект впливу нейрона на самого себе і є необхідним для даної архітектури. Ця умова не є достатньою для стійкості системи в процесі навчання мережі.

Навчання здійснюється за модифікаційним правилом Хебба. Мережа з сигналу, що поступає на її вхід виділяє відповідний зразок або повідомляє про не відповідність вхідних зразків.

Алгоритм функціонування мережі складається з наступних трьох кроків:

1. На входи мережі подається сигнал, його вхід здійснюється за допомогою задавання значень аксонів: $y_i(0) = x_i, i = 0 \dots n - 1$.

2. Розраховується новий стан нейронів наступною формулою:

$$s_j(p + 1) = \sum_{i=0}^{n-1} w_{ij}y_i(p), j = 0 \dots n - 1.$$

Та значення аксонів за формулою:

$$y_j(p + 1) = f[s_j(p + 1)],$$

де f – активаційна функція.

3. Відбувається перевірка на зміну вхідних значень аксонів за останню ітерацію, якщо вони змінилися відбувається перехід до другого пункту, а якщо ні – завершення циклу. Вхідний вектор представляє собою зразок, який найкращим чином поєднується з вхідними даними.

Інколи мережа не може виконати розпізнавання і на вихід подає неіснуючі образи, це пов'язано з обмеженими можливостями даної архітектури штучної нейронної мережі. Для мережі Хопфілда кількість образів для запам'ятовування не повинна перевищувати величину, що приблизно рівна кількості нейронів в мережі помноженій на 0.15. Також, якщо два образи є схожими, то у мережі можуть виникнути асоціації, що пересікаються і результат, який мережа подасть на вихід буде не вірним.

Когнітрон та неокогнітрон. Когнітрон було розроблено базуючись на поточних знання про анатомію і фізіологію людського мозку. Він конструюється у вигляді шарів нейронів, які є поєднаними. Предсинапсичний нейрон в одному шарі пов'язаний з предсинапсичним нейроном наступного шару. Є нейрони, які намагаються викликати збудження постсинапсичного вузла та сповільнюючі вузли, які сповільнюють його збудження. Збудження нейрону визначається зваженою сумою

його збуджуючих і сповільнюючих входів. Кожен нейрон пов'язаний тільки з нейронами сусідньої ділянки, що має назву ділянка зв'язку.

Нейрони впорядковані у вигляді у вигляді шарів, які пов'язані. Таким чином, когнітрон реалізовано ц вигляді багат шарової структури. Отримуючи навчальну вибірку входних образів, нейронна мережа самоорганізовується через зміну синапатичних зав'язків, при цьому відсутня попередньо визначена вибірка вихідних образів.

В заданій області шару навчається тільки найбільше збуджений нейрон, ті нейрони, що вже навчені достатньо, що виражається силою їх збудження, отримують прирощені сили своїх синапсів. Області зв'язку сусідніх вузлів перекриваються значною мірою, це пояснюється взаємною конкуренцією між найближчими вузлами. Якщо вузли в початковий момент мають ідентичний вихід, все одно вони будуть відрізнитись не значною мірою. Сильне збудження вузла буде стримувати збудження сусідніх вузлів і тільки його синапси будуть посилюватись, а синапси сусідніх вузлів залишаться незмінними.

Вхід нейрона в когнітроні визначається співвідношенням збуджуючих входів до сповільнюючих входів. Алгоритм навчання когнітрону дозволяє вагам синапсів зростати без обмежень. Завдяки відсутності механізму зменшення ваг, вони зростають в процесі навчання, що в звичайних лінійних порогових елементах призвело б до великого виходу елемента.

Синаптичні ваги сповільнюючих вузлів не змінюються в процесі навчання, їх ваги є попередньо визначеними таким чином, що сума ваг в будь-якому сповільнюючому вузлі дорівнює одиниці. Таким чином, вихід сповільнюючого вузла є зваженою сумою його входів, які в даному випадку представляють середнє арифметичне значення входів збуджуючих нейронів з якими він поєднаний.

Ваги збуджуючих нейронів змінюються лише тоді, коли нейрон збуджений сильніше, ніж будь-який з вузлів, що знаходиться в області конкуренції. Вузли з більшою реакцією змушують збуджуючі синапси, якими вони керують, збільшуватись швидше, ніж сповільнюючі синапси. Вузли, які мають малу реакцію,

викликають мале збільшення збуджуючих синапсів, але більше зростання сповільнюючих синапсів.

Якщо область зв'язку нейронів має постійний розмір у всіх шарах, потрібна велика кількість шарів, щоб перекрити все вхідне поле вхідними нейронами. Кількість шарів може бути зменшена шляхом розширення області зв'язку в попередніх шарах, результатом цього може бути велике перекриття областей зв'язку, таким чином, що нейрони вхідного шару будуть мати однакову реакцію. Для вирішення цієї проблеми може бути використано розширення області конкуренції, так як в певній області конкуренції може бути збуджено лише один вузол, вплив малої різниці в реакціях нейронів вхідного шару посилюється.

В альтернативному варіанті зв'язки з попереднім шаром можуть бути розподілені з більшістю синаптичних зв'язків в обмеженій області з більш довгими зв'язками, які рідко зустрічаються. Цей процес відображає розподілення нейронів в головному мозку. В когнітроні це дає змогу кожному нейрону вхідного шару реагувати на повне вхідне поле при наявності обмеженої кількості шарів.

Неокогнітрон – це спроба покращити когнітрон. Неокогнітрон та когнітрон мають певні однакові особливості, але між ними існують фундаментальні відмінності. Дві архітектури є багаторівневими ієрархічними нейронними мережами, які організовані подібним чином до зорової кори, проте, неокогнітрон більш їй відповідає. Він є більш потужною парадигмою з точки зору здатності розпізнавання зображень не залежно від перетворень та змін масштабу, ніж когнітрон. Неокогнітрон використовує самоорганізацію в процесі навчання та є орієнтованим на моделювання зорової системи людини.

На вході неокогнітрон отримує двовимірні образи, аналогічні зображенням на сітківці ока і обробляє їх в подальших шарах аналогічно тому, як це відбувається в зоровій корі людини. Він складається з послідовних шарів обробки, які організовані в ієрархічну структуру (Рисунок 3.12 – Шари когнітрона). Кожен шар неокогнітрону складається з двох масивів шарів. Масив шарів, які містять прості вузли отримує входи попереднього шару, а потім виділяє образи і після цього передає їх в масив

площин, які містять комплексні вузли, де вони оброблюються таким чином, щоб зробити виділені образи менш позиційно залежними.

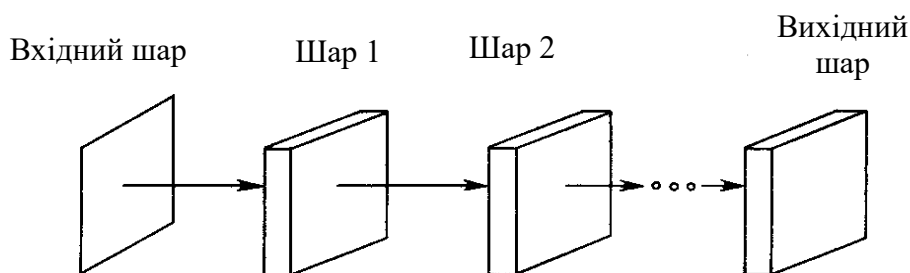


Рисунок 3.12 – Шари когнітрона

Всередині шару площини простих комплексних вузлів представлені парами, бо для площини простих вузлів існує одна площина комплексних вузлів, яка обробляє її виходи. Кожна площина може бути візуально представлена як двовимірний масив вузлів. Всі вузли площини вузлів реагують на один образ, площина простих вузлів представляє собою масив вузлів, кожен з яких налаштовується на специфічний вхідний образ. Кожен простий вузол чутливий до обмеженої області вхідного образу, яка має назву рецептивна область. Рецептивні області вузлів кожної площини простих вузлів перекриваються з ціллю покрити весь вхідний образ цього шару. Кожен вузол отримує входи від відповідних областей всіх площин комплексних вузлів в попередньому шарі. Простий вузол реагує на появу образу а будь-якій складній площині попереднього шару, якщо він появиться всередині його рецептивної області.

Задачею комплексних вузлів є зменшення залежності реакції системи від позиції образів у вхідному полі, для досягнення цього кожен комплексний вузол отримує в якості вхідного образу виходи набору простих вузлів із відповідної площини того ж шару. Ці прості вузли покривають неперервну область простої площини, яка називається рецептивною областю комплексного вузла. Збудження будь-якого простого шару в цій області є достатнім для збудження поточного комплексного вузла. Таким чином, комплексний вузол реагує на той же образ, що і прості вузли у відповідній йому площині, але він менш чутливий до позиції образу, чим будь-який з них. Кожен шар комплексних вузлів реагує на більш широку область

вхідного образу, чим на попередніх вузлах. Ця прогресія зростає лінійно від шару до шару, що призводить до зменшення чутливості системи в цілому.

Кожен шар, як результат реакції нейрону на певний образ, має менш специфічну реакцію вихідний шар реагує на повні образи, що показує незалежність до їх положення на вхідному полі. Комплексний вузол забезпечує виділення певного образу у вхідному полі з більшою вірогідністю.

В неокогнітоні лише прості вузли мають властивості налаштування ваг, це ваги зв'язків, поєднуючих вузол з комплексним вузлом попереднього шару.

Згорткова нейронна мережа. Для розуміння принципів роботи та необхідності використання згорткових нейронних мереж (Convolutional Neural Network), потрібно зрозуміти відмінності у базових принципах машинного навчання [62].

Неглибоке навчання (shallow learning) – навчання в штучній системі, яка може мати певні шари, вхідний шар, певна кількість шарів особливостей входів та класифікатор, але в ній немає ніяких вузлів регулювання. Класифікатор отримає певні числа, які можуть бути змінені за допомогою алгоритму оптимізації. Він приймає вихідні дані та сигнал помилки, щоб її мінімізувати. Навіть до появи глибокого навчання, на кожному шарі розпізнавання особливостей, виявляються більш детальні особливості та після цього вони передаються в класифікатор. Відсутність параметрів, які можна регулювати на шарах, створює проблему розпізнавання образів через нестійкість до змін масштабу зображень, поворотів, масштабування. Головна відмінність неглибокого навчання від глибокого полягає в тому, що ієрархія виявлення особливостей є фіксованою, тобто не може змінюватись, лише параметри в класифікаторі змінюються [63, 64].

Глибоке навчання (deep learning) – це навчання в штучній системі, яка має певні шари з багатьма вузлами регулювання на кожному з них. Регулюючі вузли мають параметри, які можуть бути відрегульовані за допомогою використання алгоритму тренування.

Тобто це глибока нейронна мережа, кількість прихованих шарів якої визначається комплексністю проблеми, яку потрібно вирішити. Кожен шар містить нейрони, які поєднані з іншими нейронами за допомогою зв'язків, кількість яких зі

збільшенням використовуваних прихованих шарів також зростає. Проблема полягає в тому, що сигнал похибки проходить через всі зв'язки, він розповсюджений через велику кількість ваг, що робить його занадто слабким для ефективної зміни ваг та біасів, тобто для ефективного тренування нейронної мережі. Таким чином, оптимізація стає складною, тому потрібно контролювати саму архітектуру нейронної мережі. Один із способів контролювання архітектуру нейронної мережі втілює в собі ідею розподілення ваг.

Наприклад, якщо нейрон шару 2 має зв'язки з трьома нейронами попереднього шару і вони є рівними зв'язкам наступного нейрону поточного шару з нейронами попереднього шару, то в такому випадку, алгоритмічно чи архітектурно задавши ці обмеження, можна регулювати лише одне значення, що репрезентує певну групу зв'язків (Рисунок 3.13 – Розподілення ваг).

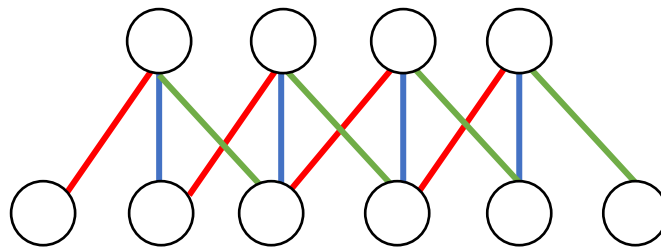


Рисунок 3.13 – Розподілення ваг

Такі нейрони розподіляють ваги. Структура ваг нейронів є фактично ідентичною відносно інших нейронів шару, їхні значення є однаковими. Тобто можна зробити висновок, що ці нейрони не є окремими, а один нейрон, застосований на різних локаціях і кількість входів, що він отримує не є суцільною з попереднього рівня, а лише її частина. Така концепція називається згорткою.

Згорткові нейронні мережі (Convolutional Neural Networks) – тип штучних нейронних мереж, що є найпопулярнішим для аналізу зображень. Даний тип нейронних мереж може бути використано для аналізу інших даних та класифікації. Згорткова нейронна мережа здатна виявляти певні патерни на зображенні, саме ця властивість робить її також ефективною у вирішенні завдань аналізу зображень. В архітектуру мережі закладено апріорні знання предметної області комп'ютерного

бачення – піксель зображення пов'язаний сильніше з сусіднім, що має назву локальна кореляція і об'єкт на зображенні може бути розміщений у будь-якій його частині.

Згорткові нейронні мережі відрізняються від звичайного багат шарового перцептрону та інших типів нейронних мереж тим, що вона містить в собі згорткові шари. CNN також містить і інші, не згорткові шари.

Згортковий шар (Convolution layer)

ЗШ, як і інші шари отримують входи та трансформують їх, потім виходи трансформують входи на наступний шар. На кожному ЗШ необхідно вказати номер фільтрів, що він буде мати. Це головна частина ЗШМ. Припускається, що пікселі, які знаходяться поруч, тісніше взаємодіють між собою при формуванні ознак, ніж ті, що знаходяться далеко. Вводиться поняття оператора згортки – це матриця розміром $h \times w$. Вона графічно кодує певну ознаку. Обчислюється згорнуте зображення, шляхом обчислення зваженої суми всіх значень пікселів (Рисунок 3.14 – Обчислення в згортковому шарі).

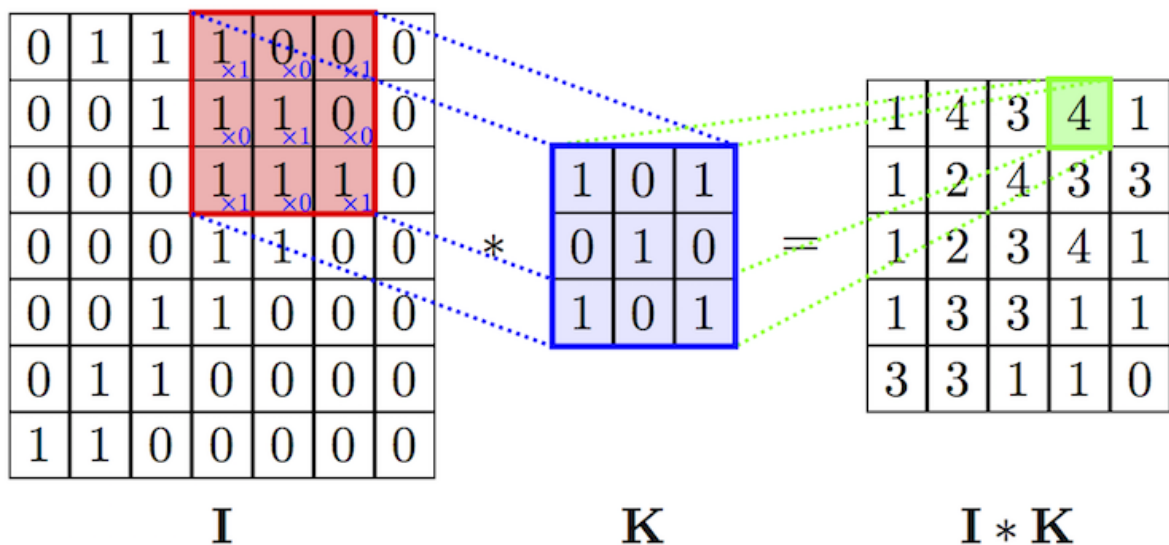


Рисунок 3.14 – Обчислення в згортковому шарі

На даному рисунку I – вхідне зображення; K – фільтр згортки (ядро), вагові коефіцієнти якого є невідомими та встановлюються в процесі навчання; $I * K$ – результуюча матриця застосування фільтру. Тобто задається зображення,

виділяються його певні особливості, значення в обох матрицях сумуються. Під згорткою розуміється процедура:

$$(I * K)[m, n] = \sum_{k,l} I[m - k, n - l] * K[k, l]$$

Головною особливістю цього шару є невелика кількість параметрів, які встановлюються при навчання. Одне зображення стає стеком особливостей цього зображення. Згортковий шар виконує одразу дві структурні системи представлення.

Пулінг або шар субкредитації (Pooling layer)

На шарі пулінгу обирається розмір вікна та крок. Після цього відбувається прохід заданим вікном по фільтрованих зображеннях (виходах з шару згортки), з кожного вікна обирається максимальне значення. Шар пулінгу побудований на принципі просторової субкредитації, тобто зменшення розмірності нової карти ознак y_j^k за рахунок усереднення значень сусідніх пікселів. Для роботи методу карта ознак ділиться на однакові області, які не пересікаються. Формула для розрахунку карти ознак y_j^{k-1} на нову y_j^k розраховується наступним чином:

$$y_j^k(x, y) = \varphi_j \left(w_{ij}^k \sum_{(u_x, u_y) \in U} y_j^{k-1}(sx + u_x, sy + u_y) \right),$$

де $U = (1 \dots s) * (1 \dots s)$, s – розмір ядра згортки; y_j^{k-1} – карта ознак; w_{ij}^k – загальні ваги нейронів j -тої карти і k -го шару різних входів x та y ; s – коефіцієнт зміни розмірності. Зазвичай s знаходиться за формулою:

$$s = 2, w_{ij}^k = \frac{1}{s^2} \equiv const$$

Для підвищення ефективності роботи згорткової нейронної мережі вводиться нелінійний коефіцієнт деформації. Найбільш ефективним методом є тип max-rolling, за рахунок якого відбувається навчання великих нейронних мереж (Рисунок 3.15 – Субкредитуєчий шар). Для цього методу формула відображення карт ознак має вигляд:

$$y_j^k(x, y) = \max y_j^{k-1}(sx + u_x, sy + u_y),$$

де $U = (1 \dots s) * (1 \dots s)$, s – розмір ядра згортки; y_j^{k-1} – карта ознак; s – коефіцієнт зміни розмірності [65].

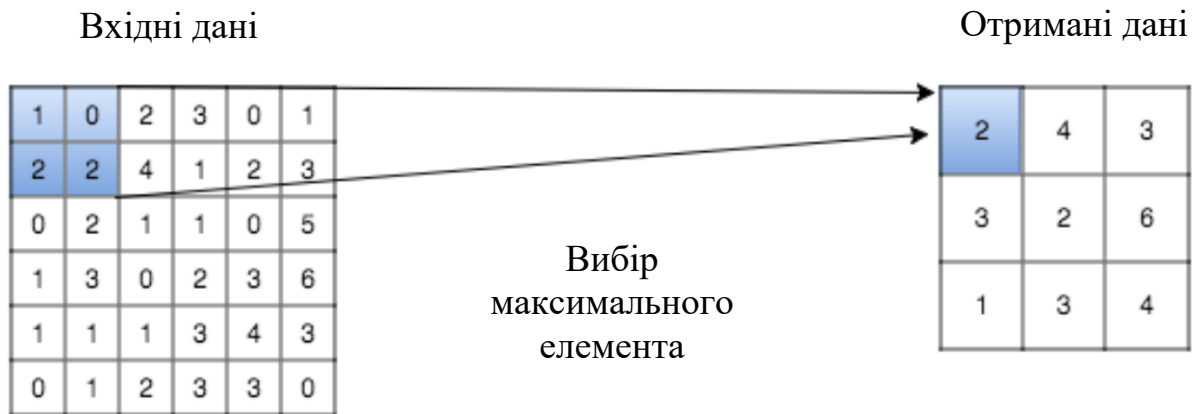


Рисунок 3.15 – Субкредитуєчий шар

Повнозв'язний шар

Є локальні поля v_j^k і їх активації y_j^k нейронів для шару N_k дані знаходяться методом прямого поширення сигналу зі значенням функції активації попереднього шару N_{k-1} і обчислюються за формулою:

$$v_j^k = \sum_i w_{ij}^k y_i^{k-1}$$

$$y_j^k = \varphi_j(v_j^k)$$

Порогове зміщення згорткової нейронної мережі функції задається для активації з фіксованим значенням $y_0^l = 1$. Це порогове зміщення є в кожного нейронного шару.

Softmax

При проектуванні задачі інколи необхідно представити виходи згорткової нейронної мережі, як вірогідності класів розпізнавання [66 – 69].

Для цього використовується функція активації Softmax, яка представляється у вигляді:

$$\varphi_j(y^k) = P(\lambda = j | y^k) = \frac{e^{(y^k)^T W_j^k}}{\sum_l (y^k)^T W_l^k},$$

де λ – коефіцієнт, який містить індекс класу, W_l^k, W_j^k – ваги функцій. Кількість класів залежить від поставленої задачі.

Висновки до розділу 3

Повнозв'язні нейронні мережі не підходять для розпізнавання зображень, головна проблема їх використання для вирішення такої задачі полягає в тому, що сигнал похибки проходить через всі зв'язки, він розповсюджений через велику кількість ваг, що робить його занадто слабким для ефективної зміни ваг та біасів, тобто для ефективного тренування нейронної мережі. Таким чином, оптимізація стає складною, тому потрібно контролювати саму архітектуру нейронної мережі. Один із способів контролювання архітектури нейронної мережі втілює в собі ідею розподілення ваг.

Згорткова нейронна мережа вирішує проблему розподілу ваг та найкраще підходить для виявлення і класифікації зображень; можливе навчання методом зворотного розповсюдження похибки; можливість використання в реальному часі; стійкість до поворотів і здвигов елементів на зображенні; можливість навчання на графічному процесорі; менша кількість параметрів для налаштування.

4. ОПИС МЕТОДІВ РЕАЛІЗАЦІЇ ТА ЗАСОБІВ РОЗРОБКИ ПРОГРАМНОЇ СИСТЕМИ

4.1 Особливості реалізації модулів системи

Щоб ефективно реалізувати систему, необхідно провести аналіз поставленого завдання та знайти оптимальний шлях його вирішення. Після проведення аналізу поставленої задачі було виділено ті критерії, яким має задовольняти система. Серед них: необхідність розбиття програми на модулі, кожен з яких буде відповідати за певний функціонал, насамперед, це обумовлено тим, що для формування сценаріїв розпізнавання вкидань, необхідно використовувати велику кількість бібліотек, їх використання краще розподілити по різних модулях для кращої читабельності коду; архітектурна можливість розширення наявних можливостей системи без зміни інтерфейсу її роботи з користувачем. В результаті аналізу вимог, яким має задовольняти система, було прийнято рішення про побудову архітектури системи з використанням модульного принципу та об'єктно-орієнтованого програмування. Цей підхід надає можливість розширення та внесення змін до системи в майбутньому.

Модульний принцип програмування – це такий принцип, при якому програма розбивається на певну групу компонентів, які мають назву модулі. Модульний принцип організації системи сприяє швидкому пошуку помилок, на відміну від монолітної архітектури, в якій не можлива вільна зміна окремих елементів, що робить такі системи низько масштабованими. Кожен модуль надає певний набір методів та можливостей, що можуть використовуватись в інших модулях, функціональні компоненти модульної програми можуть бути написані та відлагоджені окремо, інтерфейс модуля з точки зору проектування є незмінним, проте, спосіб його реалізації може бути змінено, саме таким чином забезпечується легкість тестування модуля. Час компіляції також є важливим фактором, особливо для масштабних систем, при незміні інтерфейсу модуля є можливість приєднувати до програми різні реалізації модуля, які були скомпільовані окремо, таким чином, загальний час

компіляції проекту зменшується. Модульний принцип програмування також спрощує процедуру завантаження в оперативну пам'ять великої програми, яка потребує сегментації.

Правильний розподіл функціоналу, який реалізується в модулях та оптимальність взаємодії між ними є дуже важливими факторами для реалізації модульної системи. Ця задача вирішується на етапах аналізу поставленого завдання та проектування системи. Спочатку реалізується рівень абстракції, який буде достатнім для аналізу внутрішніх елементів кожного модуля, а після цього, визначається його внутрішній склад.

Формування сценаріїв розпізнавання інформаційних вкидань на основі нейронних мереж включає в себе обробку зображень, їх аналіз та ідентифікацію лиць з метою виявлення вкидань.

Для вирішення задач розпізнавання інформаційних вкидань, що включають в себе ідентифікацію лиць та базуються на нейронних мережах пропонується об'єднане використання сучасних перспективних ідей в галузі розпізнавання зображень з використанням згорткової нейронної мережі та алгоритмів попередньої обробки зображень.

Створення такої системи можливе з об'єднаним використанням бібліотек, розроблених для вирішення подібних задач [70 - 72].

4.2 Структурно – функціональна схема роботи системи

Програмним продуктом передбачена якісна структура, яка забезпечує зручний інтерфейс для використання та містить у собі такі основні модулі:

- інтерфейс користувача – модуль, призначений для швидкої та зручної взаємодії користувача із системою та мінімізує помилки вводу;
- модуль вводу та попередньої обробки зображень – модуль, призначений для проведення аналізу вхідних даних, та їх внесення до бази даних, як зразків лиць людей, виконується виділення обличчя на зображенні, його нормалізація та приведення до розміру, необхідного для подальшої роботи системи;

- модуль навчання системи – модуль, призначений для проведення навчання нейронної мережі на основі класифікованих та попередньо оброблених попереднім модулем даних;
- модуль розпізнавання – модуль призначений для перевірки тестових даних на наявність вкидань, відбувається розпізнавання лиця на зображеннях, що були представлені користувачем для перевірки на наявність облич, що не містяться в базі даних. Функціональна схема роботи системи програмної системи представлена на рисунку 4.1.

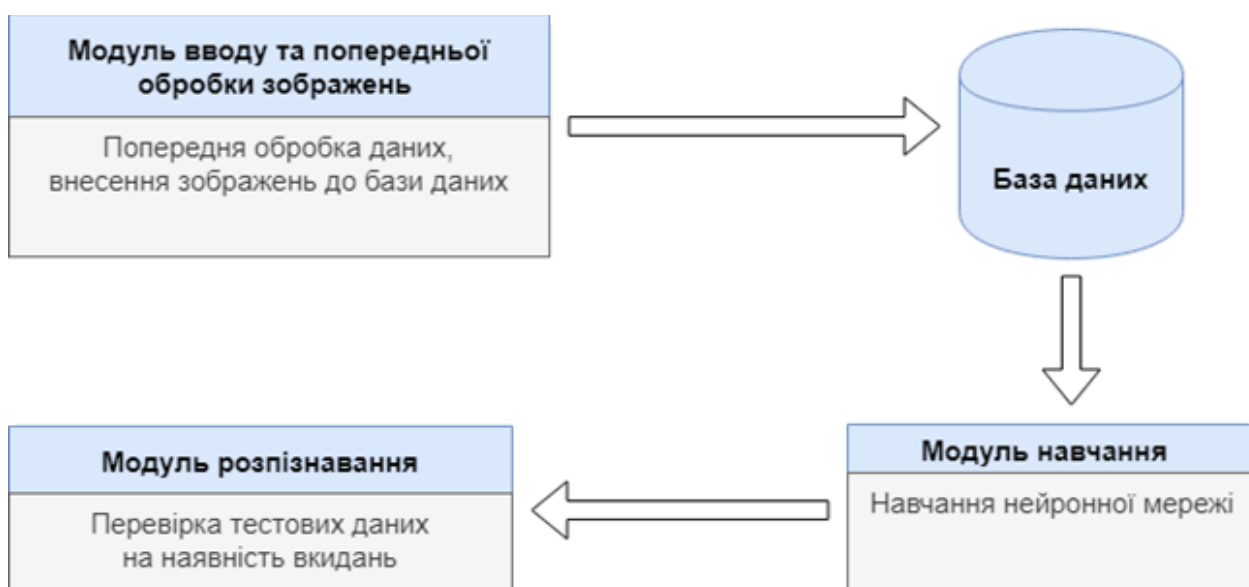


Рисунок 4.1 –Функціональна схема роботи програмної системи

При запуску програми користувач вносить класифіковані зображення людей, які потрібно внести до бази даних. Програма складається з модулів, призначення яких – аналіз даних та ідентифікація вкидань. Всі модулі пов’язані між собою, в кожному наступному модулі використовуються розрахунки попереднього. Користувач може бачити результати виконання програми, а саме ідентифікацію облич, відсутність обличчя або виявлення фейків у вигляді вікон або зберегти результати в папку, шлях до якої потрібно вказати.

На рисунку 4.2 зображено діаграму прецедентів системи де представлено виконання таких функцій системи, як завантаження даних, очищення даних,

попередня обробка даних, очистка оброблених даних, збереження даних, навчання нейронної мережі, виконання розпізнавання, збереження результатів розпізнавання, завершення роботи.



Рисунок 4.2 – Діаграма прецедентів системи

4.3 Операційна система Windows

При розробці програмного продукту потрібно враховувати особливості, можливості і поширення операційної системи, яка буде використана для роботи застосунку. Програма написана для операційної системи Microsoft Windows 10. Вибір

операційної системи насамперед обумовлений її поширеністю та функціональністю. Windows 10 відповідає вимогам сучасних апаратур, сумісна з іншими операційними системами, має високий рівень захисту, зручності та надійності. Також вона є найновішою операційною системою від Microsoft. Microsoft Windows 10 — найактуальніша операційна система від Microsoft із пріоритетною багатозадачністю.

Система може мати звичайний інтерфейс операційних систем сімейства Windows, з деякими додаваннями і розширеннями, підтримку файлових систем NTFS5, NTFS4, FAT16 і FAT32. Більшість програм, написаних під MSDOS, W9x, NT4, а також деякі програми під OS/2 і POSIX запускаються і функціонують без проблем.

Архітектура операційної системи захищає програми від ушкодження один одним і самою операційною системою. При цьому використовується відмовостійка структурована обробка особливих ситуацій на всіх архітектурних рівнях, що включає поновлювану файлову систему NTFS і забезпечує захист за допомогою вбудованої системи безпеки й удосконалених методів керування пам'яттю.

Система надає можливості для роботи в багатьох країнах світу на національних мовах, що досягається за допомогою застосування стандарту ISO Unicode. Система працює на різних процесорах сімейства x86 виробництва Intel і AMD.

4.4 Середовище розробки PyCharm 2017

PyCharm – це інтелектуальна Python IDE з повним набором засобів для ефективної розробки на мові Python. Це середовище володіє функціями автодоповнення, базуючись на інформації, яка була отримана під час виконання коду, навігації по коду та забезпечує рефакторинг.

PyCharm має такі ключові можливості:

- потужний та функціональний код редактора з підсвічуванням синтаксису, автоформатуванням та авто-відступами для підтримуваних мов;
- проста і потужна навігація по коду;
- допомога при написанні коду, що включає в себе автодоповнення, авто-

імпорт, код шаблонів, перевірку на сумісність версії інтерпретатора мови та багато іншого;

- швидкий перегляд документації для будь-якого елемента прямо в редакторі вікна, перегляд недоступних документів через браузер, підтримка docstring – генерація, підсвітка, автодоповнення;

- велика кількість інспекцій коду;

- рефакторинг коду, який надає широкі можливості по виконанню

швидких глобальних змін в проекті;

- повна підтримка новітніх версій Django фреймворка;

- підтримка Google App Engine;

- підтримка IronPython, Jython, Cython, PyPy wxPython, PyQt, PyGTK;

- підтримка Flask фреймворка та мов Mako і Jinja2;

- редактор Javascript, Coffescript, HTML / CSS, SASS, LESS, HAML;

- інтеграція з системами контролю версій (VCS);

- UML діаграми класів, діаграми моделей Django і Google App Engine;

- інтегроване тестування;

- інтерактивні консолі для Python, Django, SSH, debugger і бази даних;

- повнофункціональний графічний відладчик;

- підтримка схем найбільш популярних IDE / редакторів. таких як

Netbeans, Eclipse, Emacs, емуляція VIM редактора;

- підтримувані мови: Python (версії: 2.x, 3.x), Jython, Cython, IronPython,

PyPy, Javascript, CoffeScript, HTML / CSS, шаблони Django / Jinja2, Gql, LESS / SASS / SCSS / HAML, Mako, Puppet , RegExp, Rest, SQL, XML, YAML;

- PyCharm має кілька кольорових схем, а також підсвітку синтаксису

коду;

- інтеграція з JIRA, Youtrack, Маяк, Pivotal Tracker, GitHub, Redmine, Trac;

- велика колекція плагінів;

- кросс-платформність (Windows, Mac OS X, Linux).

Висновки до розділу 4

При розробці програмного продукту важливим чинником є правильний вибір засобів програмної реалізації, що впливає на час розробки, якість, надійність та швидкість кінцевого продукту. Було представлено функціональну схему роботи програмної системи та діаграму прецедентів системи.

Було розроблено сценарій розпізнавання інформаційних вкидань на основі нейронної мережі. Створена програма дозволяє здійснювати розпізнавання облич та їх ідентифікацію.

При розробці і реалізації програмного продукту було використане інтегроване середовище розробки PyCharm 2017, операційна система Windows, відкрита бібліотека для машинного навчання TensorFlow, бібліотека комп'ютерного бачення OpenCV.

5. МЕТОДИКА РОБОТИ КОРИСТУВАЧА З ПРОГРАМНОЮ СИСТЕМОЮ

5.1 Інсталяція та системні вимоги

Розроблена система працює в середовищі операційних систем сім'ї Windows, зокрема в найпоширеніших операційних системах, Windows 7, Windows 8 та Windows 10.

Рекомендовані вимоги до конфігурації ПК:

- оперативна пам'ять 2056 Мб;
- тактова частота процесору 1500 МГц;
- роздільна здатність екрану 1280x800 пікселів;
- периферійні пристрої: маніпулятор миша;
- кількість вільного місця на жорсткому диску 100 Мб.

5.2 Алгоритми роботи програми

Для формування сценарію розпізнавання інформаційних вкидань на основі нейронної мережі було обрано згорткову нейронну мережу та алгоритм її навчання triplet loss.

Triplet loss – метод при якому виконується порівняння пар зображень. Обирається зразок, позитивний та негативний екземпляри (anchor, positive, negative), відстань між зразком та позитивним екземпляром – мала, між зразком та негативним екземпляром – велика (порівняно з першою парою). Лінійний вектор $f(x)$ із зображення x в функціональний простір R^d , $f(x) \in R^d$ [73].

$$\|f(a) - f(p)\|^2 \leq \|f(a) - f(n)\|^2$$

$$\|f(a) - f(p)\|^2 = d(a, p), \|f(a) - f(n)\|^2 = d(a, n),$$

$d(a, p)$ – функція відстані між зразком та позитивним екземпляром, $d(a, n)$ – функція відстані між зразком та негативним екземпляром.

$$\|f(a) - f(p)\|^2 - \|f(a) - f(n)\|^2 \leq 0$$

Найпростіший спосіб задовільнити дане рівняння – дослідити всі варіанти, які рівні 0, бо якщо $\|f(a) - f(p)\|^2 = 0$, $\|f(a) - f(n)\|^2 = 0$, то $0 - 0 \leq 0$. Щоб запобігти описаному вище сценарію, вводиться спеціальний параметр α :

$$\|f(a) - f(p)\|^2 - \|f(a) - f(n)\|^2 + \alpha \leq 0$$

За допомогою параметра α відстань між зразком та позитивним екземпляром зменшується, а відстань між зразком та негативним екземпляром збільшується.

Втрата для одного триплету:

$$f(a, p, n) = \max(\|f(a) - f(p)\|^2 - \|f(a) - f(n)\|^2 + \alpha, 0).$$

Якщо $\|f(a) - f(p)\|^2 - \|f(a) - f(n)\|^2 = 0$, то $f(a, p, n) = 0$, якщо $\|f(a) - f(p)\|^2 - \|f(a) - f(n)\|^2 + \alpha > 0$, то $f(a, p, n) = \|f(a) - f(p)\|^2 - \|f(a) - f(n)\|^2 + \alpha$ [74].

Функція витрат (cost function) для всієї мережі [74]:

$$I = \sum_{i=1}^n f(a^i, p^i, n^i)$$

Дане програмне забезпечення являє собою інструментарій з використанням алгоритмів виявлення, ідентифікації лиць, попередньої обробки зображень. На рисунку 5.1 зображена діаграма активності системи. Робота програми починається з того, що користувач вводить зображення людей з їх ідентифікацією до системи (Рисунок 5.2 – Додавання нових даних), далі виконується попередня обробка зображень з подальшим представленням інформації для навчання нейронної мережі, після цього виконується перевірка вхідних даних з їх подальшим збереженням (Рисунок 5.3 – Розпізнавання даних). Додана можливість вибору папки з зображеннями, що потрібно внести до системи за допомогою діалогового вікна та можливість збереження знайдених системою вкидань.

Результати виконання перевірки зображені на рисунку 5.4 – результати перевірки.

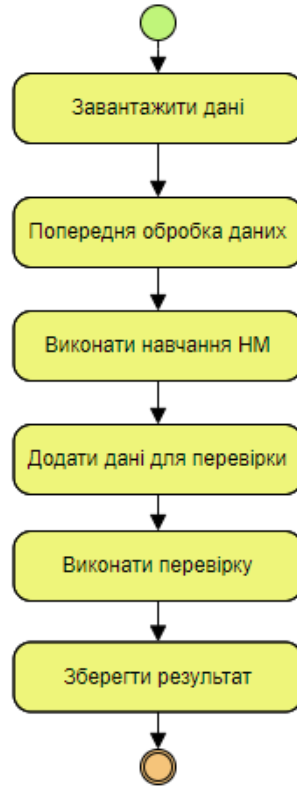


Рисунок 5.1 – Діаграма активності системи

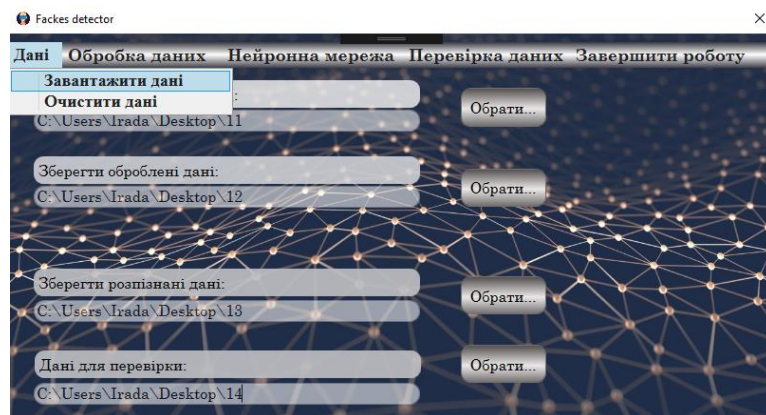


Рисунок 5.2 – Додавання нових даних

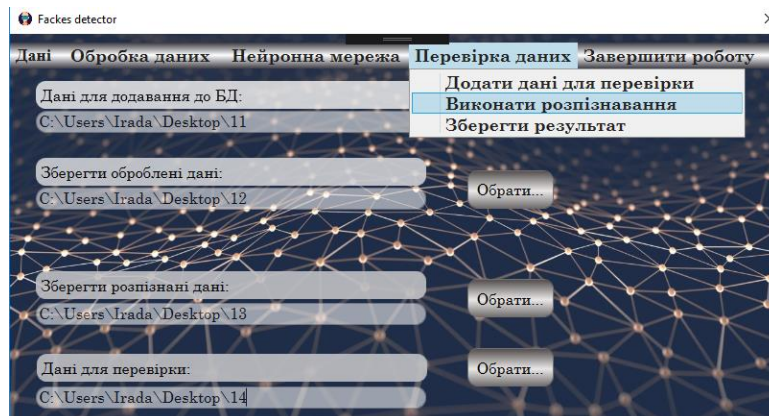


Рисунок 5.3 – Розпізнавання даних

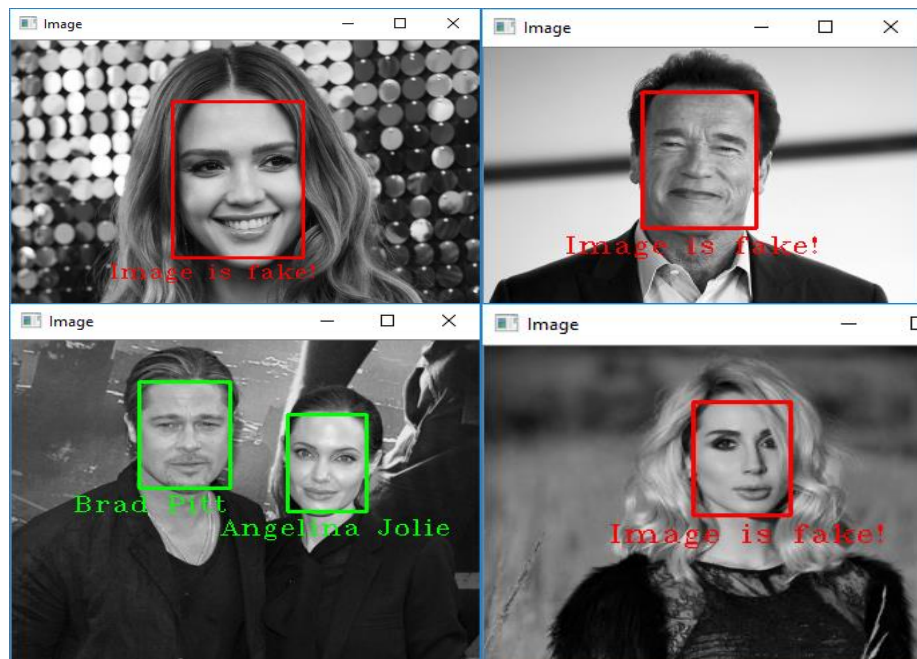


Рисунок 5.4 – Результати перевірки

Висновки до розділу 5

У розділі було описано процес інсталяції програмної системи та алгоритм її роботи.

Описано основні вимоги до ПК, для забезпечення коректної роботи програми. Було представлено діаграму активності, методику роботи користувача з системою та продемонстровано результати перевірки інформації на наявність вкидань.

6. БІЗНЕС-ПЛАН ІННОВАЦІЙНОГО ПРОЕКТУ

Ідея полягає в формуванні сценаріїв розпізнавання інформаційних вкидань на основі нейронних мереж, тобто системи, яка надає можливість ідентифікації вкидань, а саме зображень з лицами людей, які не були внесені в базу даних. Така система стане в пригоді в галузях, де ідентифікація особистості є надзвичайно важливою, наприклад, в аеропортах, на вокзалах, в банківських установах. Хоча й проблема розпізнавання та ідентифікації обличчя не є новою, існуючі методи мають свої переваги та недоліки, багато з них потребують великих обчислювальних ресурсів, часу виконання та не є достатньо точними.

6.1 Опис ідеї стартап-проекту

Проаналізуємо зміст ідеї, її можливі напрямки застосування, чим запропонована ідея відрізняється від існуючих аналогів, а також основні вигоди, які може отримати користувач товару. Результати аналізу представлені у таблиці 6.1.

Таблиця 6.1. Опис ідеї стартап-проекту

Зміст ідеї	Напрямки застосування	Вигоди для користувача
Формування сценаріїв розпізнавання інформаційних вкидань на основі нейронних мереж	1. Розпізнавання обличчя на зображенні	Розпізнавання обличчя людей на зашумленому фоні, на зображеннях низької якості та при великій кількості людей
	2. Попередня обробка зображень для тренування нейронної мережі та інших алгоритмів розпізнавання	Можливість зберегти зображення певного розміру, типу та якості, що підходить для тренування нейронної мережі та для використання отриманої бази картинок для інших методів розпізнавання
	3. Ідентифікація обличчя	Можливість ідентифікувати людину, що знаходиться на фото

На ринку існують аналоги подібних систем, але більшість з них потребують великих обчислювальних ресурсів, часу навчання та є дорогими.

Тому доцільно проводити аналіз потенційних техніко-економічних переваг ідеї порівняно з пропозиціями конкурентів. Результат аналізу у таблиці 6.2.

Таблиця 6.2. Визначення характеристик ідеї проекту

Техніко-економічні характеристики ідеї	Продукція конкурентів			Слабкі (W), нейтральні (N) та сильні (S) сторони		
	FindFace	Verifeyed	JPEGSnoop			
Назва продукту	FindFace	Verifeyed	JPEGSnoop			
Операційна система та версії	Платформи Windows	Платформи Windows	Платформи Windows		✓	
Системні вимоги	Мінімальні	Мінімальні	Від 1 ГБ ОЗУ		✓	
Розміри	200 Мб	380 Мб	400 Мб			✓
Мови програмування	C++	C++	Java			✓
Необхідність встановлення додаткового ПЗ	Тільки онлайн робота з системою	наявність АПК	наявність АПК			✓

6.2 Технологічний аудит ідеї проекту

Для проведення технічного аудиту ідеї проекту, потрібно провести аудит технології, за допомогою якої можна реалізувати ідею проекту. І для початку

потрібно визначити можливість технологічної здійсненності проекту. Результат представлений у таблиці 6.3.

Таблиця 6.3. Технологічна здійсненність ідеї проекту

Ідея проекту	Технології її реалізації	Наявність технологій	Доступність технологій
Формування сценаріїв розпізнавання інформаційних вкидань на основі нейронних мереж	Середовище розробки файлів-застосунків PyCharm 2017	✓	Доступно (є безкоштовна версія)
	Бібліотека OpenCV	✓	Доступно (безплатна)
	Бібліотека FaceNet	✓	Доступно (безплатна)

Обрана технологія доступна, не потребує доробки, а також безкоштовна та надає усі необхідні можливості для реалізації поставленої задачі.

PyCharm – це інтелектуальна Python IDE з повним набором засобів для ефективної розробки на мові Python, представлене середовище володіє функціями автодоповнення, базуючись на інформації, яка була отримана під час виконання.

6.3 Аналіз ринкових можливостей запуску стартап-проекту

Визначення ринкових можливостей, які можна використати під час ринкового впровадження проекту, та ринкових загроз, які можуть перешкодити реалізації проекту, дозволяє спланувати напрями розвитку проекту із урахуванням стану ринкового середовища, потреб потенційних клієнтів та пропозицій проектів-конкурентів. Для цього спочатку проводиться аналіз попиту (таблиця 6.4).

Таблиця 6.4. Попередня характеристика потенційного ринку стартап-проекту

Показники стану ринку	Характеристика
Загальна потреба в продукції	Необхідна, але офіційних запитів на розробку немає(брак фінансування)
Можливі річні обсяги випуску в натуральних показниках	До 1000 копій
Ціна одиниці продукції	35\$
Річні обсяги випуску в вартісних показниках	1000 – 3000\$
Динаміка ринку (якісна оцінка)	Зростає
Наявність обмежень для входу	Отримання доступу до правдивої статистичної інформації. Отримання доступу до звітів енергопідприємств
Показники стану ринку	Характеристика
Специфічні вимоги до стандартизації та сертифікації	Для ПЗ відсутні. Для коректної роботи - використання стандартів ISO 9126 та ISO 25010
Середня норма рентабельності в галузі (або по ринку)	78%

За попереднім оцінювання ринок не здається достатньо привабливим для входження. Але при проведенні аналізу поточного стану ринку виявлено підвищений інтерес до сфери енергетики та до питань забезпечення енергетичної безпеки.

Визначення ринкових можливостей, які можна використати під час ринкового впровадження проекту, та ринкових загроз, які можуть перешкодити реалізації проекту, дозволяє спланувати напрями розвитку проекту із урахуванням стану ринкового середовища, потреб потенційних клієнтів та пропозицій проектів-конкурентів.

Надалі визначаються потенційні групи клієнтів, їх характеристики, та формується орієнтовний перелік вимог до товару для кожної групи (таблиця 6.5).

Таблиця 6.5. Характеристика потенційних клієнтів стартап-проекту

Потреба, що формує ринок	Цільова аудиторія	Особливості поведінки споживачів	Вимоги споживачів до товару
Оцінка якості розрахунку показників	Персонал банків та юридичних установ	Розробники займаються написанням програм, які не завжди: відповідають стандартам (за стильовою характеристикою, наприклад), не завжди достатньо оптимізовані, що впливає на подальше життя створених програмних продуктів – виникають проблеми, недоліки та конфлікти. Тривале вирішення проблем несумісності або критичних помилок ПЗ.	– доступна ціна; – зручність і простота використання; – мобільність
	Персонал аеропортів та вокзалів	Ведені основною метою створити програмний продукт та випустити його на ринок, власники програмних розробок націлені на основні завдання такі, як: швидше створити ПП і якомога	– зручність і простота використання

		<p>вигідніше продати (більше копій, вища ціна).</p> <p>Тому важливо мати й можливість контролю.</p>	
--	--	---	--

Таблиця 6.6. Фактори загроз

Фактор	Зміст загрози	Можлива реакція компанії
Поява конкурентів	<p>Можлива поява конкурентів, які спроможуться створити більш якісний продукт.</p> <p>Можлива поява більш дешевих продуктів</p>	<p>Зменшення ціни з підвищенням якості при цьому, розробка удосконалень, розширення асортименту (додавання нових можливостей, нового функціоналу) збільшення точності розпізнавання, підвищення швидкості розпізнавання, зменшення тренувальної вибірки зображень</p>
Зміни тенденцій ринку	<p>Можлива ситуація, в якій з'явиться більш досконала програмна система від конкурентів, які значно довше на ринку.</p>	<p>Можлива така ситуація.</p> <p>Але можливості вирішення найпростіші - розробка нових сучасних необхідних удосконалень, тобто додання або заміна старого функціоналу на можливості розрахунку нових параметрів</p>
Зниження репутації компанії	<p>Можлива ситуація, коли конкуренти спроможуться на більший попит</p>	<p>Зміна партнерів, заключення нових контрактів, проведення рекламних та промо-акцій</p>

Економічний спад	Відсутність попиту на товар компанії через економічну складову	Збільшення обсягів продажів, зменшення ціни; зміна цільової аудиторії
------------------	--	---

Таблиця 6.7. Фактори можливостей

Фактор	Зміст загрози	Можлива реакція компанії
Невелика кількість конкурентів	На ринку на сьогоднішній день дуже не значна кількість конкурентів, їх програмні продукти в переважній більшості вузько спеціалізовані	Розповсюджувати створений продукт, розвивати його можливості
Відповідні тенденцій ринку	ІТ-ринок на сьогоднішній день потребує, а відповідно і надає всі можливості для впровадження систем, які надаватимуть користувачам можливість розпізнавання інформаційних вкидань	Розповсюджувати створений продукт, розвивати його можливості
Можливість побудови власної репутації	Новий «гравець» на ринку має всі можливості для побудови власної репутації з «чистого листка»	Пошук замовників, можливих покупців створеного продукту, розширення бази замовників. Зарекомендувати себе, як надійну компанію. Можливо на вигідних умовах співпраці

Надалі проводиться аналіз пропозиції – визначаються загальні риси конкуренції на ринку (таблиця 6.8): визначаються тип можливої майбутньої конкуренції та її

інтенсивність, рівень конкурентоспроможності за рівнем конкурентної боротьби, видами товарів і галузевою ознакою.

Таблиця 6.8. Ступеневий аналіз конкуренції на ринку

Особливості конкурентного середовища	У чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії)
Тип конкуренції	Чиста Залежить від кількості конкурентів та якості надання ними послуг у порівнянні з послугами компанії	Покращення власного продукту через зниження ціни та підвищення якості
За рівнем конкурентної боротьби	Локальна Конкуренція на вітчизняному ринку	На вітчизняному ринку конкурентів не виявлено, а тому компанія має можливість встановлення власної бажаної ціни, та набляти клієнтську базу. Перспектива – вихід на міжнародний рівень
За галузевою ознакою	Внутрішньогалузева Продукт націлений лише на конкретну сферу діяльності	Немає можливостей та сенсу розширювати функціонал за межі ІТ-сфери, але існує багато варіантів розвиватись всередині неї
Конкуренція за видами товарів	Марки-конкуренти	Зниження ціни, розширення функціональних, безплатне

	Створений товар може мати конкурентів, які пропонують аналогічний товар	встановлення в державних закладах охорони здоров'я (зادля популяризації методу)
За характером конкурентних переваг	Цінова Важливо за скільки продається товар, та скільки з нього прибутку	Можливе підвищення ціни на нові розробки, зниження на старі версії для заохочення покупців у порівнянні з цінами конкурентів
За інтенсивністю	Марочна Можуть з'являтися конкуренти	На ринку цільової аудиторії поки що конкурентів не виявлено. Але при виході на міжнародний ринок потрібно рекламувати кращий функціонал створеного продукту, встановлювати конкурентоспроможні ціни, та доводити свою надійність

Після аналізу конкуренції проводиться більш детальний аналіз умов конкуренції в галузі (таблиця 6.9) - за моделлю п'яти сил М. Портера, яка вирізняє п'ять основних факторів, що впливають на привабливість вибору ринку з огляду на характер конкуренції:

- конкурент, що вже є у галузі;
- потенційні конкуренти;
- наявність товарів-замінників;
- постачальники, що конкурують за ринкову владу;
- споживачі, які конкурують за ринкову владу.

На основі аналізу конкуренції, проведеного у таблиці 6.9, а також із урахуванням характеристик ідеї проекту (таблиця 6.2), вимог споживачів до товару

(таблиця 6.5) та факторів маркетингового середовища (таблиці 6.6 і 6.7) визначається та обґрунтовується перелік факторів конкурентоспроможності. Аналіз оформлюється за таблицею 6.9, обґрунтування факторів за таблицею 6.10.

Таблиця 6.9. Аналіз конкуренції в галузі за М.Портером

Складові галузі	Прямі конкуренти в галузі	Потенційні конкуренти	Клієнти	Товари-замінники
	Розробники аналогічних систем	Кращі продукти, менші ціни	Мають найбільше значення. Більш важлива їх кількість, аніж постійна співпраця	Відсутні. Є лише конкуренти аналогічних розробок
Висновки	Інтенсивність конкурентної боротьби з боку прямих конкурентів незначна	Наявні усі можливості входу на ринок. Потенційні конкуренти не виявлені. Строки виходу на ринок – один день	Необхідність клієнтської-бази, тому важливо знаходити можливості приваблення споживачів до власного продукту	Немає обмежень

Таблиця 6.10. Обґрунтування факторів конкурентоспроможності

Фактор конкурентоспроможності	Обґрунтування
Невелика кількість конкурентів на ринку	На вітчизняному ринку, на який для старту націлена розроблена система, конкурентів немає
Доступність створеного продукту (програмно)	Немає жорстких системних вимог, програма буде працювати навіть на застарілих ПК

Легкість і простота використання	Зручний зрозумілий інтерфейс, створені довідка та інструкція для користувача
Відсутня потреба у постійному супроводі	Не потребує супроводу спеціалістів і постійних доробок з боку розробника
Підключення до мережі Інтернет	Немає потреби у підключенні до мережі Інтернет після придбання продукту, на відміну від більшості аналогів
Додаткові компоненти	Немає необхідності встановлення додаткових компонентів, на відміну від деяких аналогів, які не працюють без АПК

За визначеними факторами конкурентоспроможності проводиться аналіз сильних та слабких сторін стартап-проекту, проведений у таблиці 6.11.

Таблиця 6.11. Порівняльний аналіз сильних та слабких сторін «назва проекту»

Фактор конкурентоспроможності	Бали 1-20	Рейтинг товарів-конкурентів						
		-3	-2	-1	0	+1	+2	+3
Мала кількість / відсутність конкурентів	10				✓			
Системні вимоги	18				✓			
Простота використання	18	✓						
Не потрібен супровід	10					✓		

Фінальним етапом ринкового аналізу можливостей впровадження проекту є складання SWOT-аналізу (матриці аналізу сильних (Strength) та слабких (Weak) сторін, загроз (Troubles) та можливостей (Opportunities) (таблиця 6.12) на основі виділених ринкових загроз та можливостей, та сильних і слабких сторін.

Таблиця 6.12. SWOT-аналіз проекту

<p>Сильні сторони (S):</p> <ul style="list-style-type: none"> – невелика кількість працівників; – молодий і перспективний колектив; – гнучка політика керівництва; – інноваційні технології 	<p>Слабкі сторони (W):</p> <ul style="list-style-type: none"> – брак власного устаткування; – брак робочої сили; – недостатньо оборотних коштів; – відсутність репутації компанії;
<p>Можливості (O):</p> <ul style="list-style-type: none"> – розширення виробничої лінії; – додаткові послуги; – вихід на нові ринки; – розширення клієнтської бази; – співробітництво з іншими компаніями 	<p>Загрози (T):</p> <ul style="list-style-type: none"> – поява нових конкурентів; – зміни тенденцій попиту; – зниження репутації компанії; – економічний спад

На основі SWOT-аналізу розробляються альтернативи ринкової поведінки (перелік заходів) для виведення стартап-проекту на ринок та орієнтовний оптимальний час їх ринкової реалізації з огляду на потенційні проекти конкурентів, що можуть бути виведені на ринок.

Визначені альтернативи аналізуються з точки зору строків та ймовірності отримання ресурсів (таблиця 6.13).

Таблиця 6.13. Альтернативи ринкового впровадження стартап-проекту

Альтернатива ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації
Вихід на нові ринки	Пошук інвесторів	1-6 місяців
Розширення виробничої лінії	Пошук інвесторів	Після виходу на ринок основного продукту, до 6 місяців

Отже, спочатку потрібно вивести на основний ринок розроблену систему, а вже потім шукати можливості розширення програмного функціоналу для користувачів.

6.4 Розроблення ринкової стратегії

Розроблення ринкової стратегії першим кроком передбачає визначення стратегії охоплення ринку: опис цільових груп потенційних споживачів, які визначені у таблиці 6.14.

Таблиця 6.14. Вибір цільових груп потенційних споживачів

Опис цільової групи потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтовний попит в сегменті	Інтенсивність конкуренції в сегменті	Простота входу у сегмент
Працівники банків	Потребують	Попит є	Незначна	Просто
Працівники аеропортів	Потребують	Попит є, проте нижчий, ніж у працівників банків	Незначна	Помірно

Які цільові групи обрано: оскільки різниця між цільовими групами зовсім незначна, а також враховуючи той факт, що компанія має бажання почати продажі (а відповідно і отримання прибутку) якомога швидше, то доцільно враховувати обидві цільові групи, тобто використовувати масовий маркетинг, пропонуючи стандартизовану програму.

За результатами аналізу потенційних груп споживачів (сегментів) автори ідеї обирають цільові групи, для яких вони пропонуватимуть свій товар, та визначають стратегію охоплення ринку.

Для роботи в обраних сегментах ринку необхідно сформуванати базову стратегію розвитку, яка визначається у таблиці 6.15.

Таблиця 6.15. Визначення базової стратегії розвитку

Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентоспроможні позиції відповідно до обраної альтернативи	Базова стратегія розвитку
Вихід на нові ринки	Стратегія диференціації	Надання товару відмінних якостей, які роблять систему особливою на фоні аналогічних розробок	Стратегія диференціації
Розширення виробничої лінії	Стратегія диференціації (допускається стратегія спеціалізації)	Надання товару кращих властивостей	Стратегія диференціації (допускається стратегія спеціалізації)

Вибір стратегії конкурентної поведінки визначається у таблиці 6.16.

Таблиця 6.16. Визначення базової конкурентної поведінки

Чи є проект «першопроходцем» на ринку	Так
Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?	Обидва варіанти
Чи буде компанія копіювати основні характеристики товару конкурента, і які?	Ні
Стратегія конкурентної поведінки	Стратегія виклику лідера

На основі вимог споживачів з обраних сегментів до постачальника (стартап-компанії) та до продукту, а також в залежності від обраної базової стратегії розвитку та стратегії конкурентної поведінки розробляється стратегія позиціонування (таблиця 6.17), що полягає у формуванні ринкової позиції (комплексу асоціацій), за яким споживачі мають ідентифікувати торгівельну марку або проект.

Таблиця 6.17. Визначення стратегії позиціонування

Вимоги до товару цільової аудиторії	Базова стратегія розвитку	Ключові конкурентоспроможні позиції власного стартап-проекту	Вибір асоціацій, які мають сформувати комплексну позицію власного проекту
Доступна ціна, простота і зручність використання, універсальність	Стратегія диференціації	Вирішення важливих поставлених задач швидко, легко та зрозуміло навіть без інструкцій. Легкість і простота у використанні. Доступність через ціну та технічні характеристики	– стандарти якості – метрики ПЗ – ASQAS - automated system of quality assessment software – Холстед, LOC, Джилб, МакКейб

Результатом є узгоджена система рішень щодо ринкової поведінки стартап-компанії, яка визначатиме напрями роботи стартап-компанії на ринку.

Отже, робота стартап-компанії на ринку повинна бути спланована орієнтовано таким чином: за стратегією диференціації виконаний і буде поширюватись товар відмінний за властивостями від своїх аналогів, дотримуючись у конкурентній поведінці стратегії «виклику лідера», тобто випускається один товар для усіх можливих споживачів.

6.5 Розроблення маркетингової програми стартап-проекту

Першим кроком є формування маркетингової концепції товару, який отримає споживач. Для цього у таблиці 7.18 підсумовані результати попереднього аналізу конкурентоспроможності товару.

Таблиця 6.18. Визначення ключових переваг концепції потенційного товару

Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами
Оцінка якості ПП	Оцінка за 4 метриками. Удосконалення оцінки будь-якої з обраних характеристик.	Розрахункові показники, точність та достовірність яких можна оцінювати; кількість вхідних параметрів; самостійність програмної системи.

Надалі розроблена трирівнева маркетингова модель товару: уточнюються ідея продукту, його фізичні складові, особливості процесу його надання (таблиця 6.19).

Таблиця 6.19. Опис трьох рівнів моделі товару

Рівні товару	Сутність та складові
Товар за задумом	Формування сценаріїв розпізнавання інформаційних вкидань на основі нейронних мереж. Можливість виявлення облич, ідентифікації особистості та додавання нових лиць до системи.
Товар у реальному виконанні	Властивості/характеристики Реалізовано систему розпізнавання інформаційних вкидань на основі нейронних мереж. Реалізована система розпізнавання облич. Реалізована система ідентифікації облич. Реалізовано графічне представлення результатів. Якість: тестування пройшло задовільно
Товар із підкріпленням	До продажу: стандартна розроблена система (модуль «Попередня обробка» та модуль «Розпізнавання») Після продажу: додані додаткові можливості, збільшення споживчої бази

За рахунок чого потенційний товар буде захищено від копіювання: від копіювання потенційний товар захистити не складає проблеми. Розроблена модель, на якій базується програмна система, без математичної моделі лише набір рядків коду.

Визначення цінових меж, якими необхідно керуватись при встановленні ціни на потенційний товар, яке передбачає аналіз ціни на товари-аналоги, а також аналіз рівня доходів цільової групи споживачів описано в таблиці 6.20.

Таблиця 6.20. Визначення меж встановлення ціни

Рівень цін на товари-аналоги	Рівень доходів цільової групи споживачів	Верхня та нижня межі встановлення ціни
50 – 52000 \$	500 – 5000 \$	30 – 50 \$

Наступним кроком є визначення оптимальної системи збуту, в межах якого приймається рішення (таблиця 6.21): проводити збут власними силами або залучати сторонніх посередників, вибір та обґрунтування оптимальної глибини каналу збуту, вибір та обґрунтування виду посередників.

Таблиця 6.21. Формування системи збуту

Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту
Бажання отримати більше за менші гроші	Пошук клієнтської бази та продаж	Нульовий рівень: тільки виробник	Вертикальна маркетингова система

Останньою складовою маркетингової програми є розроблення концепції маркетингових комунікацій, що спирається на попередньо обрану основу для позиціонування, визначену специфіку поведінки клієнтів (таблиця 6.22).

Таблиця 6.22. Концепція маркетингових комунікацій

Поведінка цільових клієнтів	Канали комунікацій цільових клієнтів	Ключові позиції, обрані для позиціонування	Завдання рекламного повідомлення
Бажання отримати більше за менші гроші	Будь-які	Низька ціна Широкий вибір функціоналу Легкий і простий у використанні продукт	Донести до користувача суть продукту, його якість, та залучити якомога більше зацікавлених клієнтів

Висновки до розділу 6

Маркетингова програма орієнтовано має бути побудована таким чином:

- створення продукту;
- перемови та домовленості з потенційним клієнтами;
- базова стратегія розвитку – стратегія диференціації, тобто

конкурентоспроможність формується шляхом надання споживачеві бажаного товару.

На основі ретельного вивчення споживчого середовища розробляється одна або декілька відмітних характеристик власного товару;

- стратегія конкурентної поведінки – стратегія виклику лідера, тобто на споживчому

ринку орієнтуватись на всіх можливих споживачів, у тому числі клієнтів фірм-конкурентів. Така стратегія передбачає принцип слідування за лідером ринку. За подальші цілі ставиться можливість обігнати лідерів цільового сегменту.

Стан та динаміка ринкового середовища на сьогоднішній день і ще багато років є і будуть залишатись сприятливими для впровадження розробленої системи, а також для її необхідності. Перспективи впровадження з огляду на потенційні групи клієнтів, бар'єри входження, стан конкуренції та конкурентоспроможності проекту – прямі, і тільки доводять можливість впровадження, та не марну розробку створеного продукту.

ВИСНОВКИ

Мета магістерської дисертації полягала у формуванні сценарію розпізнавання інформаційних вкидань на основі нейронної мережі. Було проаналізовано алгоритми виявлення лиця на зображенні, алгоритми попередньої обробки зображень, алгоритми ідентифікації лиця, топології нейронних мереж, методи їх навчання. В результаті дослідження були виявлені сильні та слабкі сторони цих алгоритмів, проаналізовано проблему розпізнавання обличчя та її актуальність. Врахувавши отриману інформацію, було сформовано сценарій розпізнавання інформаційних вкидань на основі нейронних мереж, було обрано та реалізовано алгоритми для кожного кроку сценарію розпізнавання. Враховуючи особливості поставленої задачі було обрано згорткову нейронну мережу, алгоритм навчання нейронної мережі triplet loss.

На основі запропонованого сценарію розпізнавання інформаційних вкидань була розроблена система для розпізнавання та ідентифікації облич. Було розроблено зручний користувацький інтерфейс, що забезпечує легку та швидку взаємодію користувача з програмним продуктом.

Розроблена система надає змогу додавати нові обличчя та виявляти ті, що не були додані до системи. При розробці і реалізації програмного продукту було використане інтегроване середовище розробки PyCharm, мова програмування Python, відкрита бібліотека для машинного навчання TensorFlow, бібліотека комп'ютерного бачення OpenCv.

Програмний продукт забезпечує такі можливості:

- завантаження даних до системи за допомогою діалогового вікна;
- завантаження даних до системи за допомогою задання шляху до папки в текстовому полі;
- очищення початкових даних;
- попередня обробка зображень;
- видалення попередньо оброблених даних з системи;
- збереження попередньо оброблених даних в окрему папку за допомогою

діалогового вікна;

- збереження попередньо оброблених даних в окрему папку за допомогою задання шляху до папки в текстовому полі;
- навчання нейронної мережі;
- завантаження даних для перевірки за допомогою діалогового вікна;
- завантаження даних для перевірки за допомогою задання шляху до папки в текстовому полі;
- перевірка даних;
- збереження результатів перевірки за допомогою діалогового вікна;
- збереження результатів перевірки за допомогою задання шляху до папки в текстовому полі;
- вихід з програми.

Розроблена система вирішує найрозповсюдженіші проблеми розпізнавання лиць, такі як чутливість до зашумленого фону, чутливість до освітлення відмінного від того, що було присутнє на навчальній вибірці, нестійкість до присутності декількох людей на зображенні.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. «Формування сценаріїв розпізнавання інформаційних вкидань на основі нейронних мереж // Сучасні проблеми наукового забезпечення енергетики: Матеріали XVI Міжнародної науково-практичної конференції аспірантів, магістрантів і студентів, 2018. – С. 240.
2. Yang G., Huang T. S. Human face detection in complex background. *Pattern Recognition*.
3. Kotropoulos C., Pitas I. Ruled based face detection in frontal views // *IEEE Intern. Conf. on Acoustics, Speech, and Signal Processing, Washington DC, USA Piscataway: IEEE Computer Society, 1997. P. 2537 – 2540.*
4. Гонсалес Р., Вудс Р. Цифрова обробка зображень. М.: Техносфера, 2005. 1072 с.
5. Красильников Н. Н. Цифровая обробка 2D- і 3D зображень. СПб: БХВ-Петербург, 2011. 608 с.
6. Кухарев, Г. А. Біометричні системи. СПб: Політехніка, 2001. 240 с.
7. Viola P., Jones M.J. Robust real-time face detection. // *Intern. J. of Computer Vision*. 2004. Vol. 57, iss. 2. P. 137 – 154.
8. Татаренков Д. А. Анализ методов обнаружения лиц на изображении // *Молодой ученый*. 2015. № 4. С. 270 – 276.
9. Viola P., Jones M. J. Rapid Object Detection using a Boosted Cascade of Simple Features. // *IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, Kaia, Hawaii, USA Piscataway: IEEE Computer Society, 2001. P. 511 – 518.*
10. Виявлення лиць на кольоровому растровому зображенні із застосуванням модифікованого методу Віоли-Джонса / М. Н. Вязовых, К. И. Зайцев, М.В. Мухортов, А. Н. Перов.
11. Freund Y., Schapire R. E. A Short Introduction to Boosting // *J. of Japanese Society for Artificial Intelligence*. 1999. Vol. 14, iss. 5. P. 771 – 780.

12. Форсайт Д.А., Понс Ж. Компьютерное зрение. Современный подход. - М.: Вильямс, 2004. – 928 с.
13. Уоссермен Ф. Нейрокомпьютерная техника: Теория и практика. - М.: Мир, 1992. – 184 с.
14. Хайкин С. Нейронные сети: полный курс, 2-е издание. – М.: Вильямс, 2008. – 1103 с.
15. Schmidhuber J. Deep Learning in Neural Networks: an Overview. *Neural Networks*. 2015. vol. 1. pp. 85–117, DOI: 10.1016/j.neunet.2014.09.003.
16. Linnainmaa S. The Representation of the Cumulative Rounding Error of an Algorithm as a Taylor Expansion of the Local Rounding Errors. University of Helsinki. 1970.
17. Linnainmaa S. Taylor Expansion of the Accumulated Rounding Error. *BIT Numerical Mathematics*. 1976. vol. 16, no. 2. pp. 146–160. DOI: 10.1007.
18. Werbos P.J. Applications of Advances in Nonlinear Sensitivity Analysis. *Lecture Notes in Control and Information Sciences*. 1981. vol. 38, pp. 762–770. DOI: 10.1007
19. Parker D.B. Learning Logic. Technical Report TR-47, Center for Computational Research in Economics and Management Science, Massachusetts Institute of Technology, Cambridge, MA. 1985.
20. LeCun Y., Bottou L., Orr G.B. Efficient BackProp. *Neural Networks: Tricks of the Trade*. 1998. pp. 9–50.
21. Rumelhart D.E., Hinton G.E., Williams R.J. Learning Internal Representations by Error Propagation. *Parallel Distributed Processing*. 1986. vol. 1. pp. 318–362.
22. Qian N. On the Momentum Term in Gradient Descent Learning Algorithms. *Neural Networks: The Official Journal of the International Neural Network Society*. 1999. vol. 12, no. 1. pp. 145–151. Stone M.N. The Generalized Weierstrass Approximation Theorem. *Mathematics Magazine*. 1948. vol. 21, no. 4. pp. 167–184.

23. Duchi J., Hazan E., Singer Y. Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. *Journal of Machine Learning Research*. 2011. vol. 12. pp. 2121–2159.
24. Kingma D.P., Ba J.L. Adam: a Method for Stochastic Optimization. *International Conference on Learning Representations (San Diego, USA, May 7-9, 2015)*, 2015. pp. 1–13.
25. Fukushima K. Increasing Robustness Against Background Noise: Visual Pattern Recognition by a Neocognitron. *Neural Networks*. 2011. vol. 24, no. 7. pp. 767–778. DOI: 10.1016/j.neunet.2011.03.017.
26. Prince S. J. D. *Computer vision: models, learning, and inference*. – Cambridge University Press. – 2012.
27. Ежов А.А., Шумский С.А. *Нейрокомпьютинг и его применения в экономике и бизнесе*/ А.А. Ежов, С.А. Шумский.— М.: МИФИ, 1998. 602 с.
28. Круг П.Г. *Нейронные сети и нейрокомпьютеры*/ П.Г. Круг.—М.: МЕИ, 2002. —176 с.
29. Оссовський С. *Нейронные сети для обработки информации* / С. Оссовский. — М.: Фінанси і статистика, 2002. — 344 с.
30. Уоссермен Ф. *Нейрокомпьютерна техніка*/ Ф. Уоссерман.— М.: Світ, 1992. — 508 с.
31. Фролов А. *Характеристики нейронных сетей*/А. Фролов.— М.: Наука, 2005. — 160 с.
32. Короткий С. *Нейронные сети: Алгоритм обратного распространения*/ С. Короткий.— М.: Вильямс, 2002. — 328 с.
33. Круглов В. В., Борисов В. В. *Искусственные нейронные сети. Теория и практика*/ В. В. Круглов, В. В. Борисов.— М.: Телеком, 2001. 382 с.
34. Хайкин С. *Нейронные сети*/ С. Хайкин.— М.: Вильямс, 2006. 1103 с.
35. Фон Нейман Дж. *Теория самовоспроизводящихся автоматов*/ Дж. Фон Нейман.— М.: Мир, 2001. — 382 с.

- 36.Галушкин, А.И. Нейрокомпьютеры: Учеб. пособие (Нейрокомпьютеры и их применение. Кн.3) / А.И. Галушкин. М.: ИПРЖР, 2000.- 528 с.
- 37.Нейронные сети: история развития теории: Учеб. пособие для вузов / ред.Галушкин А.И., ред. Цыпкин А.З. - М.: Радиотехника, 2001.- 840 с. (Нейрокомпьютеры и их применение; кн.5).
- 38.Галушкин, А.И. Теория нейронных сетей: Учеб. пособие / А.И. Галушкин. (Нейрокомпьютеры и их применение. Кн.1). М.: ИПРЖР, 2000.- 416 с.
- 39.Галушкин, А.И. Нейрокомпьютеры и их применение на рубеже тысячелетий в Китае. Т.1 и 2 / А.И. Галушкин. М., 2004.- 367 + 464 с.
- 40.Нейропрограммы: Учебное пособие: В 2 ч. Ч. 1/ Л.В.Гилева, С.Е.Гилев, А.Н.Горбань и др.; Красноярск: КГТУ, 1994.- 123 с.
- 41.Нейропрограммы: Учебное пособие: В 2 ч. Ч. 2/ Л.В.Гилева, С.Е.Гилев, А.Н.Горбань и др.; Красноярск: КГТУ, 1994.- 123 с.
- 42.Горбань, А.Н. Нейронные сети на персональном компьютере / А.Н. Горбань, Д.А. Россиев. Новосибирск: Наука, 1996.- 276 с.
- 43.Нейроинформатика / А.Н. Горбань, В.Л. Дунин-Барковский, А.Н. Кирдин и др. Новосибирск: Наука, 1998.- 296 с.
- 44.Горбань, А.Н. Обучение нейронных сетей / А.Н. Горбань. М.: СП "Paragraph", 1990.- 160 с.
- 45.Миркес, Е.М. Нейрокомпьютер. Проект стандарта / Е.М. Миркес. Новосибирск: Наука, 1999.- 337 с.
- 46.Осовский, С. Нейронные сети для обработки информации / Осовский С. - М.: Финансы и статистика, 2002. – 343 с.
- 47.Рутковская, Д. Нейронные сети, генетические алгоритмы и нечеткие системы / Рутковская Д. и др. М., 2004.- 452 с.
- 48.Хайкин, С. Нейронные сети: Полный курс. 2-е издание / Хайкин С.. М.: "Вильямс", 2006.- 1104 с.
- 49.Дорогов А.Ю. Быстрые нейронные сети / Дорогов А.Ю. М.: Изд-во СанктПетербургского университета, 2002.- 77 с.

50. Легалов А.И. Нейроинформатика: Уч. пособие / Легалов А.И., Миркес Е.М., Сиротинина Н.Ю. - Красноярск, 2006. – 172 с.
51. Методы нейроинформатики: Сб. научн. трудов / Под ред. А.Н. Горбаня. - Красноярск: КГТУ, 1998. - 204 с.
52. Терехов, С.А. Вводные лекции по теории и приложениям искусственных нейронных сетей (рукопись) / Терехов С.А. Красноярск, 2000. 69 с.
53. Минский М. Перцептроны / Минский М., Пейперт С. М.: Мир, 1971.
54. Kohonen T. Self-Organizing Maps / T. Kohonen. – Berlin: Springer-Verlag, 1995. – 362 p.
55. Bezdek J.C. Pattern Recognition with Fuzzy Objective Function Algorithms / J.C. Bezdek. – N.Y.: Plenum Press, 1981. – 272 p.
56. Vuorimaa P. Fuzzy self-organizing maps / P. Vuorimaa // Fuzzy Sets and Systems. – 1994. – 66. – P. 223-231.
57. Vuorimaa P. Use of fuzzy self-organizing maps in pattern recognition / P. Vuorimaa // In: Proc. 3-rd IEEE Int. Conf. Fuzzy Systems “FUZZ-IEEE’94”. – Orlando, USA, 1994. – P. 798-801.
58. Tsao E. C.-K. Fuzzy Kohonen clustering networks / E. C.-K. Tsao, J.C. Bezdek, N.R. Pal // Pattern Recognition. – 1994. – 27. – P. 757-764.
59. Pascual-Marqui R.D. Smoothly Distributed Fuzzy cMeans: a New Self-Organizing map / R.D. Pascual-Marqui, A.D. Pascual-Montano, K. Kochi, J.M. Caraso // Pattern Recognition. – 2001. – 34. – P. 2395-2402.
60. Haykin S. Neural Networks. A Comprehensive Foundation / S. Haykin. – Upper Saddle Press, N.Y.: Prentice Hall, Inc., 1999. – 842 p.
61. Vazan M.T. Stochastic Approximation / M.T. Vazan. – Cambridge: AT The University Press, 1969.
62. Scherer D., Müller A., Behnke S. Evaluation of Pooling Operations in Convolutional Architectures for Object Recognition // Lecture Notes in Computer Science. 2010. Vol. 6354, P. 92–101. DOI: 10.1007.
63. Viren J., Sebastian S. Natural Image Denoising with Convolutional Networks // Advances in Neural Information Processing Systems (NIPS) 21. 2009. P. 769–776.

64. Ruo Chen W., Zhe X. A Pedestrian and Vehicle Rapid Identification Model Based on Convolutional Neural Network // Proceedings of the 7th International Conference on Internet Multimedia Computing and Service (ICIMCS '15) (Zhangjiajie, China, August 19–21, 2015), 2015. P. 32:1–32:4. DOI: 10.1145.
65. Boominathan L., Kruthiventi S.S., Babu R.V. CrowdNet: A Deep Convolutional Network for Dense Crowd Counting // Proceedings of the 2016 ACM on Multimedia Conference (Amsterdam, The Netherlands, October 15–19, 2016), 2016. P. 640–644. DOI: 10.1145.
66. Kinnikar A., Husain M., Meena S.M. Face Recognition Using Gabor Filter And Convolutional Neural Network // Proceedings of the International Conference on Informatics and Analytics (Pondicherry, India, August 25–26, 2016), 2016. P. 113:1–113:4. DOI: 10.1145.
67. Szegedy C., Liu W, Jia Y. et al. Going Deeper with Convolutions // IEEE Conference on Computer Vision and Pattern Recognition (Boston, MA, USA, June 7–12, 2015), 2015. P. 1–9. DOI: 10.1109.
68. He K., Sun J. Convolutional Neural Networks at Constrained Time Cost // 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (Boston, MA, USA, June 07–12, 2015), 2015. P. 5353–5360. DOI: 10.1109.
69. Jia Y., Shelhamer E., Donahue J., et al. Caffe: Convolutional Architecture for Fast Feature Embedding // Proceedings of the 22nd ACM International Conference on Multimedia (Orlando, FL, USA, November 03–07, 2014), 2014. P. 675–678. DOI: 10.1145.
70. Бойко В. В. C# Language Specification/ В. В. Бойко. — М.: ECMA International, 2006. — 286 с.
71. Троелсен Е. Язык программирования C# 5.0 и платформа .NET 4.5/ Е. Троелсен — М.: Вильямс, 2013. — 1312 с.
72. Шилдт Г. C# 4.0: полное руководство C#/ Г. Шилдт.— М.: Вильямс, 2010. — 1056 с.
73. Florian Schroff, Dmitry Kalenichenko, James Philbin – FaceNet: A Unified Embedding for Face Recognition and Clustering.

ДОДАТОК А

Апробації

Формування сценаріїв розпізнавання інформаційних вкидань на основі
нейронних мереж

УКР.НТУУ"КПІ"_ТЕФ_АПЕПС_ ТР22224_18М

Аркушів 4

2018

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ
СІКОРСЬКОГО»

СУЧАСНІ ПРОБЛЕМИ НАУКОВОГО ЗАБЕЗПЕЧЕННЯ ЕНЕРГЕТИКИ

Матеріали XVI Міжнародної
науково-практичної конференції
аспірантів, магістрантів і студентів
м. Київ, 24-27 квітня 2018 року,

ТОМ 2



Київ- 2018

<i>ГВОЗДЕНКО О.В., магістрант гр. ПІ-71мп</i> <i>Керівник - доц., к.т.н. Стативка Ю.І.</i>	
Методи та засоби аналізу надвеликих масивів даних у реальному часі.	239
<i>ЛИСТОПАДОВ Д.О., магістрант гр. ТМ-71мп</i> <i>Керівник - доц., к.т.н. Коваль О.В.</i>	
Формування сценаріїв розпізнавання інформаційних вкидань на основі нейронних мереж.	240
<i>МАГЕРАМОВА І.І., магістрант гр. ТР-71мп</i> <i>Керівник - доц., к.ф.-м.н. Коваль О.В.</i>	
Інструментальні засоби фінансового аналізу бюджету.	241
<i>МАРЧИШИНА О.В., магістрант гр. ТР-71мп</i> <i>Керівник - доц., к.т.н. Коваль О.В.</i>	
Відновлення траєкторії руху морських об'єктів векторно-фазовим методом.	242
<i>МОРОЗОВ М.С., магістрант гр. ПІ-71мп</i> <i>Керівник - доц., к.т.н. Варава І.А.</i>	
Синтаксичний аналіз простих речень.	243
<i>МУЗИКА В.В., магістрант гр. ПІ-71мп</i> <i>Керівник - доц., к.т.н. Стативка Ю.І.</i>	
Методичні основи розробки системи моделювання ризиків досягнення сталого розвитку України .	244
<i>ФЕДОРЕНКО В.Є., магістрант гр. ТМ-71мп</i> <i>Керівник - доц., к.е.н. Караєва Н.В.</i>	
Характеристика методів аналізу ризиків в енергетиці.	245
<i>ФЕДЧИШИН М.О. магістрант гр. ТМ-71мп</i> <i>Керівник - доц., к.е.н. Караєва Н.В.</i>	
Реінжиніринг бізнес-процесів діяльності кафедри.	246
<i>ФЕДЬКІН С.С., магістрант гр. ТР-71мп</i> <i>Керівник - доц., к.т.н. Кузьмініч В.О.</i>	
Обробка гідроакустичних сигналів за допомогою вейвлетів.	247
<i>ШЕВЧУК О.О., магістрант гр. ПІ-71мп</i> <i>Керівник - доц., к.т.н. Варава І.А.</i>	
Автоматична класифікація текстів за жанровими ознаками.	248
<i>ІЛЬЧИШИН Д.В., магістрант гр. ТВ-71мп</i> <i>Керівник - доц., к.т.н. Стативка Ю.І.</i>	
Система аналізу ризиків радіаційної безпеки.	249
<i>ВЛАСЕНКО Н.А., студент гр. ТМ-41</i> <i>Керівник - ст.викл. Бандурка О.І.</i>	
Система для аналізу і оптимізації поширення госпітальних інфекцій в лікарняному відділенні.	250
<i>ВІННІЧУК М.В., студент гр. ТВ-41</i> <i>Керівник - ст.викл. Бандурка О.І.</i>	
Програма для графічного порівняння витрат і корисності діагностичних методів .	251
<i>ВІННІЧУК М.В., студент гр. ТВ-41</i> <i>Керівник - ст.викл. Бандурка О.І.</i>	
Середовище для написання SQL запитів.	252
<i>ГОЛЬДИЧ Я.Є., студент гр. ТВ-42</i> <i>Керівник - доц., к.т.н. Стативка Ю.І.</i>	
Енергоаудит системи опалення багатоквартирного будинку - методика, ефективність результатів.	253
<i>ГУРСЄВ М.В., студент гр. ОТ-41</i>	

УДК 004.4

Магістрант 5 курсу, гр. TP-71мп Магерарова І.І.
Доц., к.ф.-м.н. Коваль О.В.

ФОРМУВАННЯ СЦЕНАРІЇВ РОЗПІЗНАВАННЯ ІНФОРМАЦІЙНИХ ВКИДАНЬ НА ОСНОВІ НЕЙРОННИХ МЕРЕЖ

З розвитком інформаційних технологій та мережі Інтернет проблема розпізнавання інформаційних вкидань постає доволі гостро. За допомогою інформації відбувається нанесення шкоди різним процесам та системам, які на ній базуються. Інформаційні операції – це засіб, який використовується для впливу на інформацію та інформаційні системи, вони зустрічаються в таких сферах: воєнна, економічна, соціальна.

Для аналізу та виявлення інформаційних вкидань використовуються спеціалізовані системи моніторингу інформаційного простору, вони забезпечують оперативність та необхідні аналітичні засоби.

Одним із найвпливовіших інструментів проведення інформаційних операцій виступає “розвідка по відкритих джерелах” (Open source intelligence, OSTIN), вона включає в себе пошук, відбір і збір інформації, яка була отримана із загальнодоступних джерел, а також аналіз цієї інформації. OSTIN включає в себе комерційну, інтелектуальну розвідку і бізнес-аналітику. Веб-ресурси є основними інструментами, які використовуються при зборі загальнодоступної інформації з відкритих ресурсів.[1,2]

Засоби контент-моніторингу оброблюють запити, які складаються з ключових слів, вони мають різний рівень важливості у відповідності з компонентами, яких стосуються.

Проблема існуючих сценаріїв розпізнавання інформаційних вкидань полягає в тому, що вони не є достатньо ефективними, не здатні впоратись з великим об’ємом інформації та є слабкими відносно нових засобів розповсюдження вкидань, оскільки не здатні під них підлаштовуватись.

Нами проаналізовано використання алгоритму нейронномережових технологій для вирішення даної проблеми. Цей алгоритм підходить для обробки великих масивів даних та може працювати з неповними даними.

Перелік посилань:

1. Додонов А.Г. Распознавание информационных операций/ А.Г. Додонов, Д.В. Ландэ, В.В. Цыганок, О.В. Андрейчук, С.В. Каденко, А.Н. Грайворонская. – К.: ООО «Инжиниринг», 2017. – 282 с.
2. Aguiar-Contraria, L. Using Wavelets to Decompose the Time-Frequency Effects of Monetary Policy / Luis Aguiar-Contrariaa, Nuno Azevedob, Maria Joana Soares // *Physica A: Statistical Mechanics and its Applications*. – 2008. – Vol. 387. – Issue 12. –P. 2863–2878.

ДОДАТОК Б

Акт впровадження

Формування сценаріїв розпізнавання інформаційних вкидань на основі
нейронних мереж

УКР.НТУУ"КПІ"_ТЕФ_АПЕПС_ ТР22224_18М

Аркушів 2

2018

