

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Інститут інформатики та радіоелектроніки
Факультет комп'ютерних наук та технологій
(повне найменування інституту, факультету)

Кафедра комп'ютерних систем та мереж
(повне найменування кафедри)

Пояснювальна записка

до дипломного проекту (роботи)

бакалавра

(ступінь вищої освіти (освітній ступінь))

на тему РОЗРОБКА СИСТЕМИ РОЗПІЗНАВАННЯ ЕМОЦІЙ
НА ОБЛИЧЧІ ЛЮДИНИ

Виконав: студент 4 курсу, групи КНТ-527
спеціальності _____

123 «Комп'ютерна інженерія»

(код і найменування спеціальності)

Освітня програма (спеціалізація) _____

«Комп'ютерна інженерія»

Тронькіна Олеся Едуардівна

(прізвище та ініціали)

Керівник Тягунова М. Ю.

(прізвище та ініціали)

Рецензент Морщавка С. В.

(прізвище та ініціали)

м. Запоріжжя
2021 рік

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»
 (повне найменування вищого навчального закладу)

Інститут, факультет інформатики та радіоелектроніки, комп'ютерних наук і технологій
 Кафедра «Комп'ютерні системи та мережі»
 Ступінь вищої освіти (освітній ступінь) бакалаврський
 Спеціальність 123 Комп'ютерна інженерія
 (код і найменування)
 Освітня програма (спеціалізація) Комп'ютерна інженерія
 (назва освітньої програми (спеціалізації))

ЗАТВЕРДЖУЮ

Завідувач кафедри Кудерметов Р.К.

“ ” 2021 року

З А В Д А Н Н Я
НА ДИПЛОМНИЙ ПРОЕКТ (РОБОТУ) СТУДЕНТА

Тронькіної Олесі Едуардівни

(прізвище, ім'я, по батькові)

1. Тема проекту (роботи) Розробка системи розпізнавання емоцій на обличчі людини
 керівник проекту (роботи) Тягунова Марія Юріївна, к. т. н., доцент

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

затверджені наказом вищого навчального закладу від “17” березня 2021 року № 81

2. Строк подання студентом проекту (роботи) 06 травня 2021 року

3. Вихідні дані до проекту (роботи) наукові статті по тематиці розпізнавання об'єктів на зображенні, відкриті бази сегментованих зображень осіб з різним емоційним станом, мова програмування Python

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити

1) Аналіз предметної області;

2) Розробка системи розпізнавання емоцій;

3) Навчання та тестування нейронної мережі.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)

Плакат 1 – Точність розпізнавання на моделі та функція втрат;

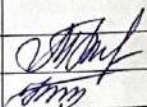
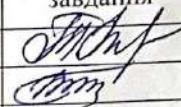
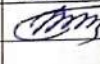
Плакат 2 – Результат роботи системи;

Креслення 1 – Блок-схема алгоритма системи розпізнавання емоцій;

Плакат 3 – Тестування емоцій на моделі;

Плакат 4 – Архітектура нейронної мережі.

6. Консультанти розділів проекту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	прийняв виконане завдання
1-4	Тягунова М. Ю., к.т.н., доцент		
нормоконтроль	Щербак Н.В., ст. викл.		

7. Дата видачі завдання 01.03.2021 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів проекту (роботи)	Примітка
1	Аналіз підходів розпізнавання емоцій	10.02.2021 р.	
2	Аналіз інструментальних засобів, реалізуючих розпізнавання емоцій (PE)	15.02.2021 р.	
3	Розробка структури системи PE	20.02.2021 р.	
4	Розробка алгоритму та блок-схеми алгоритму функціонування системи	25.02.2021 р.	
5	Визначення навчальної, тестувальної та контрольної вибірки даних	01.03.2021 р.	
6	Розробка архітектури нейронної мережі	05.03.2021 р.	
7	Реалізація нейронної мережі	08.03.2021 р.	
8	Налаштування параметрів нейронної мережі	25.03.2021 р.	
9	Навчання нейронної мережі	30.03.2021 р.	
10	Тестування нейронної мережі	01.04.2021 р.	
11	Тестування роботи системи	10.05.2021 р.	
12	Оформлення пояснювальної записки	20.05.2021 р.	
13	Захист роботи	26.05.2021 р.	

Студент



О. Е. Тронькіна

(ініціали та прізвище)

Керівник проекту (роботи)



М. Ю. Тягунова

(ініціали та прізвище)

РЕФЕРАТ

ПЗ: 69 с., 7 формул, 36 рис., 60 джерел.

КОМП'ЮТЕРНИЙ ЗІР, НЕЙРОННІ МЕРЕЖІ, РОЗПІЗНАВАННЯ ОБЛИЧЧЯ, СИСТЕМА РОЗПІЗНАВАННЯ ЕМОЦІЙ, ШТУЧНИЙ ІНТЕЛЕКТ.

Об'єкт дослідження – система розпізнавання емоцій.

Предмет дослідження – методи, алгоритми і моделі класифікації об'єктів на зображеннях.

Мета роботи – створити систему, що дозволяє ідентифікувати емоції людини з відеопотоку з нахиленим обличчям та поганим освітленням.

Проект складається з трьох розділів.

У першому розділі розглянуто предметну область, визначено поняття штучного інтелекту, глибинного навчання, описано переваги та недоліки технологій. Також було розглянуто сучасні системи визначення емоцій від провідних компаній світу.

У другому розділі розглянуто інструменти для побудови моделей штучного інтелекту. Розібрано класичний алгоритм розпізнавання емоцій по ключовим точкам, зроблено висновки щодо його ефективності. Описано процес побудови моделі машинного навчання, процес збору навчальних даних, їх нормалізація. Представлено архітектуру згорткової нейронної мережі, обрано метрики якості, скомпільовано.

У третьому розділі представлено процес навчання нейронної мережі, її тестування, оцінено якість розпізнавання на тестовій вибірці та інтегровано з системою розпізнавання області обличчя. Також описано повний алгоритм роботи системи.

ЗМІСТ

Перелік скорочень та умовних позначок	5
Вступ.....	7
1 Аналіз предметної області.....	10
1.1 Основні поняття Artificial Intelligence.....	10
1.2 Deep Learning та нейронні мережі	21
1.3 Аналіз систем розпізнавання обличчя	30
2 Розробка системи розпізнавання емоцій	33
2.1 Інструментальні засоби розробки нейронних мереж	35
2.2 Алгоритм розпізнавання області обличчя	38
2.3 Алгоритм розпізнавання емоцій	41
2.3.1 Згорткові нейронні мережі	43
2.3.2 Збір навчальних даних та їх нормалізація	46
2.3.3 Архітектура згорткової нейронної мережі	50
2.3.4 Компіляція моделі	54
3 Навчання та тестування системи	55
3.1 Процес навчання нейронної мережі	55
3.2 Тестування працездатності нейронної мережі	57
3.3 Інтеграція моделі в систему	61
Висновки	63
Перелік джерел посилання	64

ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАК

API – Application Programming Interface;

AI – Artificial Intelligence;

IDE – Integrated Development Environment,

CV – Computer Vision;

CNN – Convolutional Neural Network;

OpenCV – бібліотека комп'ютерного зору;

ER – Emotion Recognition;

FR – Face Recognition;

DL – Deep Learning;

ВСТУП

Розпізнавання емоцій є важливою областю вивчення психології людини та пересікається з такими областями, як інформатика чи когнітивна наука. Емоція – основний показник людського стану і наміри до дій, реакція на зовнішній подразник. Емоція – важлива властивість людини, яка необхідна для ефективної взаємодії з суспільством. Людське спілкування може бути вербальним або невербальним. Доведено, що більшість людей використовують невербальний метод спілкування. У невербальному спілкуванні емоції грають ефективну роль і передають думку людей про предмет, а в психологічних дослідженнях доведено, що міміка більш ефективна, ніж усне мовлення в розмові.

Одним з перших досліджень емоцій належить Дюшену ще в 1862 році. Він досліджував як м'язи людини виробляють міміку. Але найважливіша робота в цій області належить Полу Екману. Саме він описав шість основних універсальних людських емоцій: гнів, страх, відраза, щастя, печаль, здивування. Розпізнавання емоцій може широко застосовуватися в взаємодії людини і робота, відеоіграх, рекламі, соціальних мережах, медицині і освітній системі. Зчитувати емоції можна через зображення, відеопотік, мову, текст, мову тіла, інтонацію та електроенцефалографію.

В останні роки взаємодія людини і комп'ютера розвивається все більш стрімкіше і відіграє важливу роль в розвитку інформатики шляхом створення широкої різноманітності практичних навиків, щоб досліджувати поведінку людей з комп'ютерних пристроїв. У дослідженнях робототехніки, особливо теорії розуму, корисно застосувати розпізнавання емоцій на машині, щоб забезпечити природний спосіб спілкування між людиною і машиною, як людські істоти.

Інформація, що вказує на вираз обличчя поширюється по різних ділянках обличчя, і кожна з них має різну інформацію, тому рот і очі містять більше інформації ніж щока і лоб. Серед наявних психологічних досліджень було зроблено

висновки, що культура і етнічна культура можуть вплинути на характер емоцій і спосіб вираження почуттів до людських істот. Багато з психологічних досліджень показали, що гендер та вік не впливають на характер вираження емоцій, поки є нечіткі докази важливості навколишнього середовища для тенденції вираження емоцій.

Автоматичне розпізнавання виразів емоцій включає в себе три кроки: отримання зображення особи, витяг рис обличчя і визначення області особи розпізнавання виразу емоцій. В оптимальних витягнутих ознаках варіації вираження всередині класу повинні бути мінімальними, при цьому міжкласові варіації повинні бути максимальними. Якщо витягнуті ознаки не підходять для даного завдання і не містять достатньо інформації, навіть найкращий класифікатор може бути безуспішно для вирішення завдання.

Виділення ознак для розпізнавання емоцій можна розділити на два підходи: геометричні та методи, засновані на зовнішньому вигляді. У перших методах грає роль розташування та форма частин обличчя, таких як очі, рот, брови і ніс, тоді як при другому способі враховуються окремі ділянки або ціле обличчя.

Простір ознак виразів, що не схожі та розрізняються між собою є важкою проблемою, тому розпізнавання виразу обличчя як і раніше є складною задачею для комп'ютерів. Деякі проблеми можуть бути пов'язані з тим, що витягнуті ознаки двох осіб з однаковим виразом обличчя можуть відрізнятися, також риси, що витягнуті з однієї особи можуть мати різні схожі класи емоцій, наприклад «страх» і «смуток» можуть бути подібними. Тому, обрана тема дипломної роботи є актуальною.

Мета роботи – створити систему, що дозволяє ідентифікувати емоції людини з відеопотоку з нахиленим обличчям та поганим освітленням.

Для досягнення поставленої мети необхідно виконати наступні завдання:

- проаналізувати особливості предметної області та методи розпізнавання облич і емоцій на них;
- проаналізувати та обрати необхідні програмні засоби для розробки;

- розробити систему розпізнавання емоцій на обличчі людини на базі згорткової нейронної мережі;
- навчити та протестувати нейронну мережу;
- проаналізувати отримані результати.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Основні поняття Artificial Intelligence

Сфера AI – це автоматизація завдань, раніше розв'язуваних лише людським інтелектом. Цей величезний спектр включає в себе дуже різноманітні методи машинного навчання, глибинного навчання, генетичні алгоритми та безліч підходів, не пов'язаних напряду із навчанням.

Аренд Хінтце, доцент кафедри інтегративної біології, інформатики та інженерії в Університеті штату Мічиган, описав в статті «Understanding the Four Types of Artificial Intelligence» [1], що AI можна розділити на чотири категорії, починаючи зі спеціалізованих інтелектуальних систем, широко використовуваних сьогодні, і закінчуючи розумними системами, яких ще немає. Категорії наступні:

- Тип 1 [2]: реактивні машини. Ці системи штучного інтелекту не мають пам'яті і залежать від конкретного завдання. Прикладом може служити Deep Blue, шахова програма IBM, яка перемогла Гаррі Каспарова в 1990-х роках. Deep Blue може визначати фігури на шахівниці і робити прогнози, але, оскільки у нього немає пам'яті, він не може використовувати минулий досвід для інформування майбутніх.

- Тип 2: обмежена пам'ять. У деяких систем AI є пам'ять, тому вони можуть використовувати минулий досвід для прийняття майбутніх рішень. Саме за таким принципом влаштовані функції в безпілотних технологіях [2].

- Тип 3: Теорія розуму. Цей психологічний термін пояснює, що наша когнітивна система обробляє інформацію так само, як це робить комп'ютер. Стосовно до AI це означає, що система буде мати соціальний інтелект, щоб розуміти емоції. Тобто AI зможе робити висновки про людські наміри і передбачати поведінку [3].

- Тип 4: Самосвідомість. У цій категорії системи мають почуття себе, яке надає їм усвідомленість. Це в якомусь розумінні розширене поняття теорії розуму.

Машини із самосвідомістю розуміють свій поточний стан. Такого типу AI поки не існує [4].

Когнітивні обчислення – це використання комп'ютеризованих моделей для моделювання розумового процесу людини в складних ситуаціях. Терміни AI і когнітивні обчислення іноді використовуються як синоніми, але, термін AI використовується відносно машин, які замінюють людський інтелект, моделюючи те, як ми сприймаємо, навчаємося, обробляємо і реагуємо на інформацію в навколишньому середовищі [5]. Ярлик когнітивні обчислення використовується відносно продуктів і послуг, які імітують і покращують процеси мислення людини.

Штучний інтелект інтегровано в різні технології. Він присутній в функції розпізнавання осіб мобільних телефонів, в віртуальних голосових помічниках, таких як Siri від Apple, Alexa від Amazon або Cortana від Microsoft, і інтегрований в наші повсякденні пристрої через роботів і мобільних додатків. Перше – це автоматизація. У поєднанні з технологіями штучного інтелекту роботизована автоматизація процесів (RPA) може розширити обсяг і типи виконуваних завдань. Роботу, що постійно повторюється можна значно спростити та обробляти дані на основі правил, що виконуються людьми. У поєднанні з машинним навчанням і сучасними інструментами штучного інтелекту RPA може автоматизувати велику частину корпоративних завдань, особливо на підприємствах з великим ризиком дозволяючи ботам RPA передавати інформацію і реагувати на зміни процесів [6].

Інша технологія – це Machine Learning, яка впливає з назви, система навчається, але не програмується в явному вигляді. На вхід система приймає численні і часто різномірні приклади, які стосуються розв'язуваної задачі, а вона шукає у вхідних прикладах статистичну структуру. У свою чергу, дана структура дозволяє алгоритму визначити правила для автоматичного вирішення поставленого завдання [7]. Є три типи алгоритмів машинного навчання:

- Supervised learning [8]. Набори даних позначаються, щоб можна було виявляти закономірності і використовувати їх для маркування нових наборів даних.

В рамках навчання з учителем існують 2 типи популярних задач – регресія та класифікація. Завдання регресії [9] – це передбачення деякого дійсного числа на основі вхідних даних. Звідси випливає, що на виході моделі не буде якогось заздалегідь відомого обмеженого числа значень. Модель машинного навчання може видати будь-яке число в діапазоні від мінус нескінченності до нескінченності. Проблема регресії – це проблема, в якій ми намагаємося зробити прогноз в безперервному масштабі [10]. Прикладами можуть бути прогнозування вартості акцій компанії або прогнозування температури на основі історичних даних, трейдинг. Модель регресії наведена на рисунку 1.1.

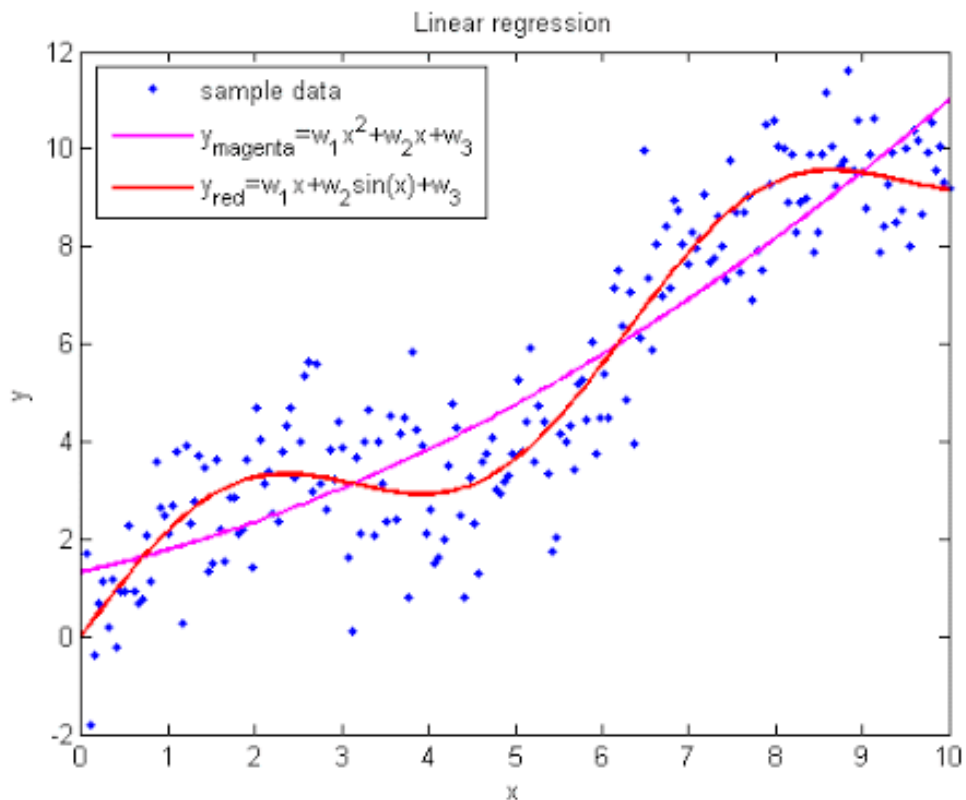


Рисунок 1.1 – Приклад лінійної регресії

Приклади використання алгоритмів для задач контрольованої регресії: лінійна регресія, нелінійна регресія, Байєсова лінійна регресія.

Класифікація – категоріальна відповідь на основі набору ознак об'єктів. Прикладом проблеми класифікації може бути аналіз зображення, щоб визначити,

чи є на ньому автомобіль або людина, або аналіз медичних даних, щоб визначити, чи належить певна людина до групи високого ризику по певного захворювання чи ні. Іншими словами, ми намагаємося використовувати дані для прогнозування дискретного набору значень або категорій [11].

Приклади алгоритмів, використовуваних для задач контрольованої класифікації: наївний байесовський класифікатор, метод опорних векторів, логістична регресія (наведено на рисунку 1.2), нейронні мережі.

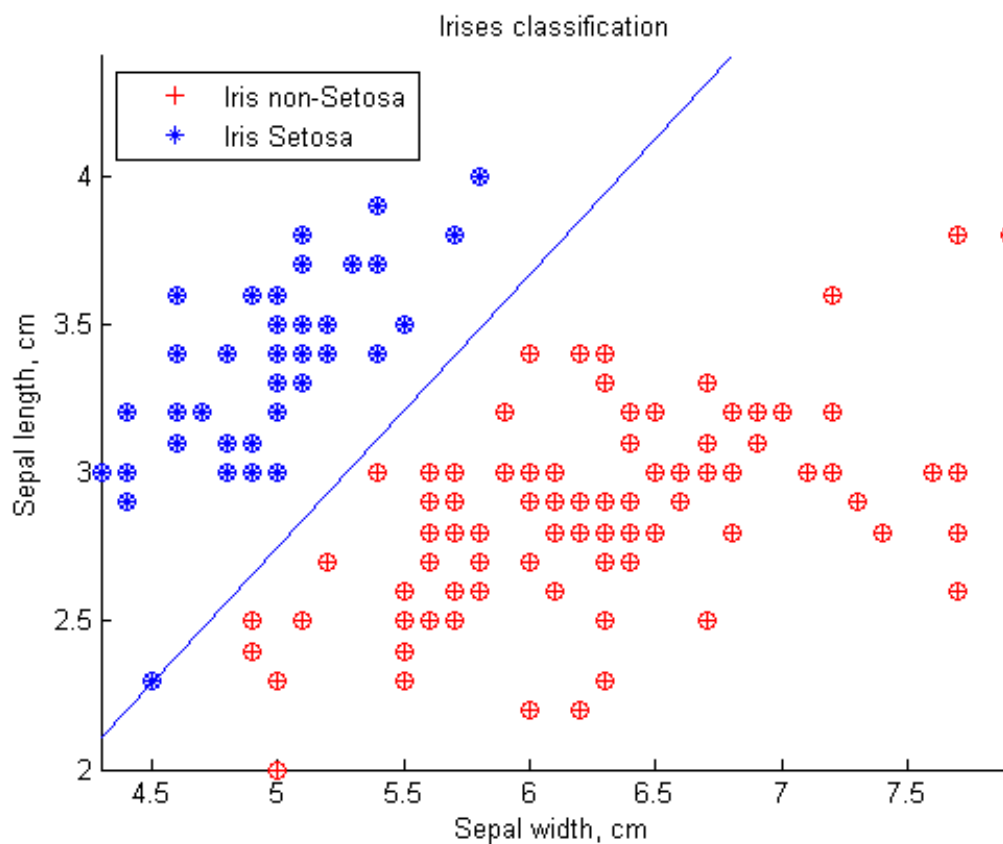


Рисунок 1.2 – Приклад логістичної регресії

– Unsupervised learning [13]. Набори даних не позначаються і упорядковані за подібністю чи розбіжністю.

– Reinforcement learning. Набори даних не маркуються, але після виконання дії або декількох дій система AI отримує зворотний зв'язок.

Робототехніка. Ця область інженерії зосереджена на розробці і виробництві роботів. Роботи часто використовуються для виконання завдань, які людям складно

виконувати або виконувати постійно. Наприклад, роботи використовуються на складальних лініях для виробництва автомобілів або НАСА для переміщення великих об'єктів в космосі. Дослідники також використовують машинне навчання для створення роботів, які можуть взаємодіяти в соціальних мережах [14]. Мета AI – поліпшити робототехніку за допомогою машинного навчання і поліпшити машинне навчання за допомогою робототехніки.

Неймовірно швидка поява штучного інтелекту і робототехніки в нашому суспільстві змусило міжнародні організації задуматися про необхідність встановлення нормативних положень, що регулюють їх використання і зайнятість, з метою запобігання будь-яких потенційних проблем, які можуть виникнути в майбутньому [14]:

- 1) у конструкції роботів повинні бути передбачені механізми відмови (аварійні вимикачі), щоб уникнути небезпечних ситуацій;
- 2) робот не може заподіяти шкоду людству. Робототехніка була розроблена спеціально для того, щоб допомагати і захищати людей;
- 3) не допускається розвиток емоційного зв'язку між людьми і роботами;
- 4) страховий поліс буде обов'язковим для більших одиниць, власники роботів несуть відповідальність за матеріальний збиток, заподіяний їх роботами;
- 5) права і обов'язки роботів будуть класифіковані відповідними законами;
- 6) роботи і машини будуть платити внески на соціальне страхування, їх вихід на ринок праці вплине на робочу силу багатьох компаній, роботи повинні платити податки, щоб субсидувати допомогу з безробіття.

Безпілотні автомобілі. Автономні транспортні засоби використовують комбінацію комп'ютерного зору, розпізнавання зображень і глибокого навчання для розвитку автоматизованих навичок пілотування транспортного засобу, залишаючись на заданій смузі і уникаючи несподіваних перешкод, таких як пішоходи. Технології допомоги водієві в сучасних автомобілях вже допомагають рятувати життя і запобігати травми [14]. Деякі сучасні автомобілі оснащені технологією, яка допомагає водіям уникати занесення на сусідні смуги руху або

небезпечні зміни смуги руху, або попереджає водіїв інших транспортних засобів позаду них, коли вони рухаються заднім ходом, або автоматично гальмує, якщо автомобіль попереду зупиняється або раптово сповільнюється. Ці технології безпеки використовують комбінацію обладнання (датчики, камери і радар) і програмного забезпечення, щоб допомогти транспортним засобам ідентифікувати певні ризики безпеки, щоб вони могли попередити водія аби він міг уникнути аварії. Постійний розвиток автомобільних технологій направлено на забезпечення ще більших переваг в області безпеки і на створення автоматизованих систем керування, які можуть впоратися з усіма завданням водіння [15].

Natural language processing. Це обробка людської мови комп'ютерною програмою. Один з найстаріших і найбільш відомих прикладів NLP – це виявлення спаму, який переглядає рядок теми і текст електронного листа і визначає, чи є воно небажаним. Сучасні підходи до NLP засновані на машинному навчанні. Завдання NLP включають переклад тексту, аналіз тональності і розпізнавання мови [16].

Machine vision. Ця технологія дає машині можливість бачити. Машинний зір фіксує і аналізує візуальну інформацію за допомогою камери, аналого-цифрового перетворення та цифрової обробки сигналів. Його часто порівнюють із зором людини, але машинне зір не пов'язане біологією і може бути запрограмовано, наприклад, бачити крізь стіни. Він використовується в ряді програм, від ідентифікації підпису до аналізу медичних зображень. Computer vision, сфокусований на машинній обробці зображень, часто асоціюється з машинним зором [17].

Переваги штучного інтелекту. По-перше, машина приймає рішення на основі попередніх записів даних. За допомогою алгоритмів ймовірність помилок знижується. Це досягнення, оскільки для вирішення складних завдань потрібні складні обчислення, які можна виконати без помилок. Бізнес-організації використовують цифрових помічників для взаємодії зі своїми користувачами, це допомагає їм значно економити час. Попит на бізнес користувачів задовільний, і їм

не потрібно чекати. Вони запрограмовані на надання максимальної допомоги користувачу.

По-друге, AI не втомлюється і швидко зношується. Штучний інтелект і наука про робототехніку використовуються в гірничодобувній промисловості та інших процесах розвідки палива. Ці складні машини допомагають дослідити дно океану і долати людські обмеження. Завдяки програмуванню роботів, вони можуть виконувати завдання з великими труднощами і більшою відповідальністю [18].

По-третє, цифрова підтримка допомагає в повсякденних справах. Siri слухає нас і виконує завдання одним дотиком. GPS допоможе вам подорожувати по світу. Їжа, одяг, укриття і смартфон. Саме вони пророкують, що ми збираємося друкувати, коротше кажучи, вони знають нас краще, ніж будь-хто. Функція автозаміни розуміє, що ви намагаєтеся сказати, та надає пропозицію найкращим чином. Штучний інтелект широко використовується фінансовими і банківськими установами, тому що він допомагає організовувати дані і управляти ними.

По-четверте, об'єкт, що приймає раціональні рішення. У високорозвинених організаціях є цифрові помічники, які допомагають їм взаємодіяти з користувачами і скорочують потребу в людських ресурсах. Правильні програмні рішення можуть бути прийняті, якщо над ними працювати раціонально. Але люди керуються емоціями. Коли ми думаємо, вони взагалі не відволікаються. У них немає емоційної сторони, і це змушує роботів мислити логічно. Емоції з ними не пов'язані, тому настрої не знижує працездатності. Таким чином, вони завжди продуктивні [19].

По-п'яте, повторювані роботи. Крім того, що завдання повторюються, вони мають однаковий характер і можуть виконуватися за допомогою машинного інтелекту. Машини думають швидше людей і можуть одночасно виконувати різні функції. Їх можна використовувати для виконання небезпечних завдань, а їх параметри регулюються. Це неможливо з людьми, оскільки їх швидкість і час не можуть бути розраховані на основі параметрів [19].

По-шосте, медичні додатки. Це найкращий приклад застосування. Лікарі оцінюють пацієнтів і ризики для їх здоров'я за допомогою штучного машинного

інтелекту. Додатки допомагають інформувати про побічні ефекти різних ліків. В даний час медичні працівники проходять навчання на симуляторах штучної хірургії. Він використовує додаток, який допомагає виявляти і контролювати неврологічні розлади і стимулювати функції мозку. Це також допомагає в радіохірургії. Радіохірургія використовується при оперуванні пухлин і допомагає в операції, не пошкоджуючи навколишні тканини [18].

Моделі не потрібні перерви, як це роблять люди. Вони запрограмовані на довгі години і можуть працювати безупинно і не відволікаючись. Машина не втомлюється, навіть якщо їй доводиться працювати кілька годин поспіль. Це головна перевага перед людьми, яким час від часу потрібен відпочинок, щоб працювати ефективно. Однак на ефективність моделей штучного інтелекту не впливають будь-які зовнішні чинники, і це не заважає безперервній роботі [20].

Ухвалення правильних рішень. Повна відсутність емоцій у машини робить її більш ефективною, оскільки вони здатні приймати правильні рішення за короткий проміжок часу. Кращим прикладом цього є використання в охороні здоров'я. Інтеграція інструментів штучного інтелекту в сектор охорони здоров'я підвищила ефективність лікування за рахунок зведення до мінімуму ризику помилкового діагнозу [20].

Впровадження AI в ризикованих ситуаціях. Про безпеку людини піклуються машини. Вчені використовують складні машини для вивчення дна океану, де виживання людини ускладнено. Це рівень AI. Він досягає місця, недоступного для людей. Таким чином, допомагає вирішувати проблеми одразу.

Недоліки штучного інтелекту розглянемо далі. Висока вартість. AI вимагає високої вартості та величезних витрат, так як це складна машина. Крім вартості установки, її ремонт та обслуговування також вимагають величезних витрат. Програмне забезпечення потребує частого оновлення і відповідає потребам мінливого середовища. При цьому відновлення також вимагає величезного часу.

Немає реплікації людини. Незалежно від того, наскільки розумною стає машина, вона ніколи не зможе відтворити розум людини. Машини раціональні, але

дуже штучні, оскільки не володіють емоціями і моральними цінностями. Вони не знають, що етично, а що законно, і через це не мають власних навичок судження. Вони роблять те, що їм програмують робити, і тому думки про добро чи зло, для них дорівнює нулю. Якщо вони стикаються з незнайомою їм ситуацією, вони діють неправильно або в таких ситуаціях не дієздатні [21].

Ніяких поліпшень з досвідом. Штучний інтелект не може бути поліпшений з досвідом, він може знову виконувати ту ж функцію, якщо їм не дана інша команда. Згодом це може призвести до зносу системи. AI зберігає багато даних, але спосіб доступу і використання сильно відрізняється від людського інтелекту.

Крім того, вони не можуть впоратися з динамічним середовищем і тому не можуть змінити свою реакцію на мінливу середу.

Машини не можуть бути творчими. Вони можуть робити тільки те, чого їх навчають. Хоч вони і допомагають в проектуванні і створенні, вони не можуть зрівнятися з силою людського мозку [22].

Люди чутливі, інтелектуальні і дуже креативні. Вони можуть генерувати ідеї, можуть нестандартно мислити. Вони бачать, чують, думають і відчують, але машина так не може. Їхні думки керуються почуттями, яких абсолютно не вистачає машині. Незалежно від того, наскільки машина розумна, їй не властиві інтуїтивні здібності людського мозку, і вона не може їх відтворити.

Безробіття. Це небезпечно і може мати серйозні наслідки. Завдяки капіталомістким технологіям потреба в людиномістких технологіях в деяких галузях знизилась. Якщо в майбутньому люди не покращать свої навички в найкоротші терміни, ми побачимо, що вони будуть замінені машинами. Основною проблемою стагнації ВВП або його не зростання очікуваними темпами є безробіття. Люди не володіють необхідними навичками, які користуються попитом. Через це існує величезний розрив попиту та пропозиції [23].

Штучний інтелект використовують в безлічі сфер. Наприклад, в охороні здоров'я. Найбільші ставки зроблені на поліпшення результатів лікування пацієнтів і зниження витрат. Компанії застосовують машинне навчання, щоб ставити більш

точні і швидкі діагнози, ніж люди. Одна з найвідоміших медичних технологій – IBM Watson. Він розуміє природну мову і може відповідати на поставлені йому запитання. Система збирає дані про пацієнтів і інші доступні джерела даних, щоб сформулювати гіпотезу, яку потім представляє зі схемою оцінки достовірності [17].

Інші додатки штучного інтелекту включають використання онлайн-віртуальних помічників по здоров'ю і чат-ботів, які допомагають пацієнтам і клієнтам охорони здоров'я знаходити медичну інформацію, призначати зустрічі, розуміти процес виставлення рахунків і виконувати інші адміністративні процеси. Безліч технологій штучного інтелекту також використовується для прогнозування, боротьби і розуміння пандемій, таких як COVID-19 [21].

AI в бізнесі. Алгоритми машинного навчання інтегруються в платформи аналітики та управління взаємовідносинами з клієнтами (CRM), щоб отримати інформацію про те, як краще обслуговувати клієнтів. Чат-боти були вбудовані в веб-сайти для негайного обслуговування клієнтів. Автоматизація робочих місць також стала предметом обговорення серед науковців та IT-аналітиків [24].

AI в освіті. AI може автоматизувати виставлення оцінок, даючи викладачам більше часу. Він може оцінювати учнів і адаптуватися до їх потреб, допомагаючи їм працювати в своєму власному темпі. Викладачі штучного інтелекту можуть надати студентам додаткову підтримку, щоб вони не збилися зі шляху. І це може змінити як студенти навчаються, можливо, навіть замінивши деяких вчителів [25].

AI в фінансах. AI в додатках для особистих фінансів, таких як Intuit Mint або TurboTax, руйнує фінансові інститути. Подібні додатки збирають особисті дані і надають фінансові консультації. Інші програми, такі як IBM Watson, застосовувалися в процесі покупки будинку. Сьогодні програмне забезпечення штучного інтелекту виконує більшу частину торгівлі на Уолл-Стріт [26].

AI в законі. Процес відкриття – просіювання документів – в юриспруденції часто буває непосильним для людей. Використання AI для автоматизації трудомістких процесів в юридичній галузі дозволяє заощадити час і поліпшити обслуговування клієнтів. Юридичні фірми використовують машинне навчання для

опису даних і прогнозування результатів, комп'ютерний зір для класифікації та вилучення інформації з документів і обробку природної мови для інтерпретації запитів на інформацію [27].

AI в виробництві. Виробництво знаходиться в авангарді впровадження робіт в робочий процес. Наприклад, промислові роботи, які колись були запрограмовані для виконання окремих завдань і відокремлені від людей-робітників, все частіше функціонують як коботи: більш дрібні, багатозадачні роботи, які взаємодіють з людьми і беруть на себе відповідальність за більшу кількість частин роботи на складах, виробничих майданчиках і інших робочих місцях [28].

AI в банківській справі. Банки успішно використовують чат-ботів, щоб інформувати своїх клієнтів про послуги та пропозиції і обробляти транзакції, які не потребують втручання людини. Віртуальні помічники використовуються для поліпшення і скорочення витрат на дотримання банківських правил. Банківські організації також використовують штучний інтелект, щоб поліпшити процес прийняття рішень по кредитах, встановити ліміти кредиту та визначити інвестиційні можливості [29].

AI в транспорті. Крім фундаментальної ролі AI в управлінні автономними транспортними засобами, технології штучного інтелекту використовуються в транспорті для управління рухом, прогнозування затримок рейсів і підвищення безпеки та ефективності морських перевезень [30].

Штучний інтелект і машинне навчання знаходяться у верхній частині списку сучасних технологій, які сьогодні використовують постачальники засобів безпеки, щоб виділити свої пропозиції. Штучний інтелект і машинне навчання в продуктах кібербезпеки приносять реальну користь командам безпеки, які шукають способи виявлення атак, шкідливих програм та інших загроз [27].

Організації використовують машинне навчання в програмному забезпеченні для керування інформацією і подіями безпеки (SIEM) в пов'язаних областях для виявлення аномалій і виявлення підозрілих дій, що вказують на загрози.

Аналізуючи дані і використовуючи логіку для виявлення подібності з відомим шкідливим кодом, AI може надавати оповіщення про нові і сприяють усуненню нових атак набагато раніше, ніж співробітники-люди і попередні технологічні ітерації [30].

В результаті технологія безпеки AI різко знижує кількість помилкових спрацьовувань і дає організаціям більше часу для протидії реальним загрозам до того, як буде завдано шкоди.

1.2 Deep Learning та нейронні мережі

Deep Learning (DL) – це підрозділ машинного навчання, що імітує роботу мозку людини при обробці даних і створенні шаблонів для прийняття рішень. Це ключова технологія, що лежить в основі безпілотних автомобілів, що дозволяє їм розпізнавати знак зупинки або відрізнити пішохода від ліхтарного стовпа. Це ключ до голосового управління в споживчих пристроях, таких як телефони, планшети, телевізори і гучномовці. Останнім часом глибокому навчанню приділяється багато уваги, і не дарма. Це досягнення результатів, які раніше були неможливі [31].

При глибокому навчанні комп'ютерна модель вчиться виконувати завдання класифікації безпосередньо із зображень, тексту або звуку. Моделі глибокого навчання можуть досягати найвищої точності, іноді перевищувати продуктивність людського рівня. Моделі навчаються з використанням великого набору помічених даних і архітектур нейронних мереж, які містять багато шарів. DL акцентує увагу на аналіз послідовних шарів все більш значущих уявлень [31]. Під глибиною в глибинному навчанні мається на увазі не більш глибоке розуміння, а багат шаровість моделі (рис. 1.3). Кількість шарів, з яких складається модель даних називають глибиною моделі. Тому глибинне навчання також відоме, як «многошарове» та «ієрархічне» навчання.

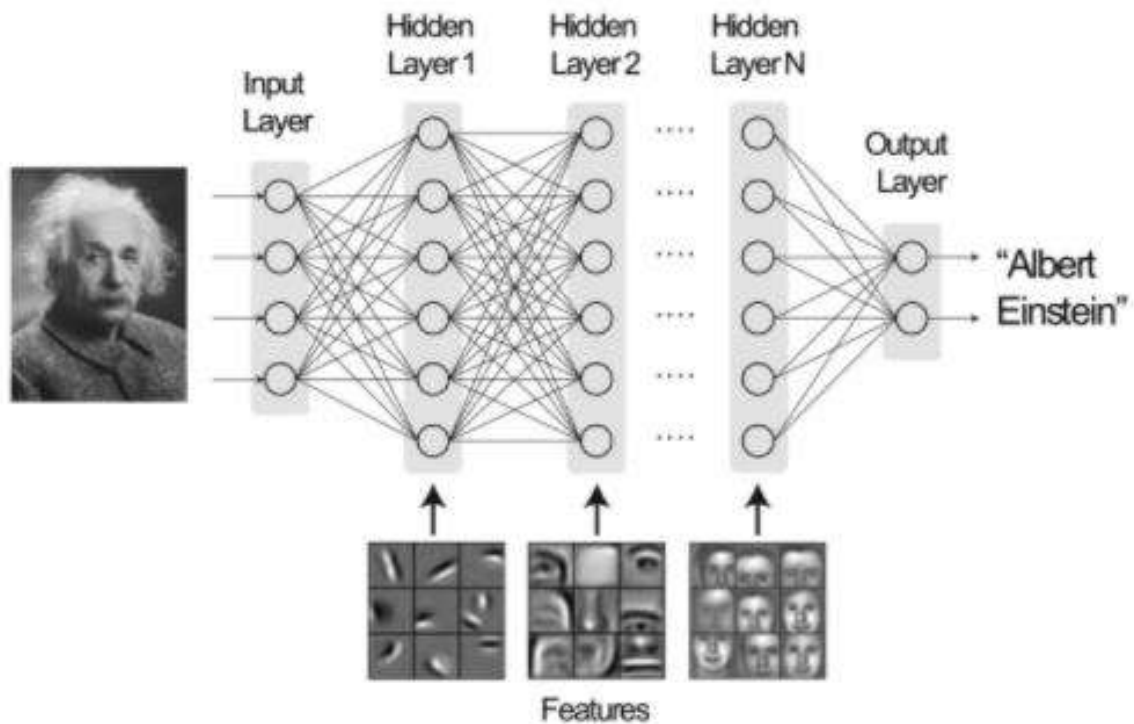


Рисунок 1.3 – Схема нейронної мережі глибокого навчання для класифікації зображень

DL забезпечує точність розпізнавання на більш високому рівні. Це допомагає відповідати очікуванням користувачів і має вирішальне значення для критично важливих додатків, таких як автомобілі без водія. Останні досягнення в даній області покращилися настільки, що глибоке навчання перевершує людей в деяких завданнях, таких як класифікація об'єктів на зображеннях.

Методика глибокого навчання має дві важливі характеристики:

- 1) Поетапно, пошарово конструює все більш складні уявлення.
- 2) Досліджує проміжні представлення спільно, за рахунок чого кожен шар оновлюється відповідно до інформації, отриманої від уявлень інших шарів.

Є дві основні причини, чому ця область стала популярною нещодавно:

- 1) Для deep learning потрібні великі обсяги помічених даних. Наприклад, для розробки автомобілів без водія потрібні мільйони зображень і тисячі годин відео.
- 2) Deep learning вимагає значних обчислювальних потужностей. Високопродуктивні графічні процесори мають паралельну архітектуру, яка

ефективна для глибокого навчання. У поєднанні з кластерами або хмарними обчисленнями це дозволяє командам розробників скоротити час навчання для мережі глибокого навчання з тижнів до годин або менше.

3) Крім обладнання і даних, до кінця 2000-х, нейронним мережам не вистачало ефективного способу навчання. Нейронні мережі залишалися не дуже глибокими – мали один або два шари уявлення, через це вони поступалися більш досконалим поверхневим методам, таким як метод опорних векторів і reinforcement trees. Основна проблема полягала в розповсюдженні градієнта через глибокі пакети шарів. Сигнал зворотнього зв'язку, який використовують для навчання нейронних мереж, загасає в міру збільшення кількості шарів. Все змінилося в 2009-2010 роках, коли з'явилися прості, але важливі алгоритмічні удосконалення, що дозволили поліпшити поширення градієнта: вдосконалені функції активації, кращі схеми оптимізації, такі як RMSProp і Adam. [32]

Більшість методів глибокого навчання використовують архітектури нейронних мереж, а ті натомість базуються на глобальних принципах біологічного Глибинні нейронні мережі автоматизували одне із складних завдань машинного навчання – вибір ознак (features), які є значущими для вирішення завдання з безлічі доступних. Глибокі нейронні мережі автоматично витягують важливі ознаки в процесі навчання. Але за цю можливість доводиться «розплачуватися» дуже великими обчислювальними ресурсами. Проте це стосується лише етапу навчання. Нейронна мережа, що пройшла цей етап працює швидко та потребує мало ресурсів. До того ж нейронна мережа дає ще одну перевагу перед іншими алгоритмами машинного навчання – вона може довчитися на окремих прикладах без повного перенавчання на всіх даних [33].

Нервова система людини – це найбільший обробник інформації в людському тілі (рис. 1.3). Це не єдиний обробник інформації але він найбільший і найефективніший. Нервова система людини складається з кількох компонентів. По-перше – мозок, який відповідає за обробку інформації та прийняття рішень. Інформація до мозку надходить від органів чуття по нервових волокнах [33].

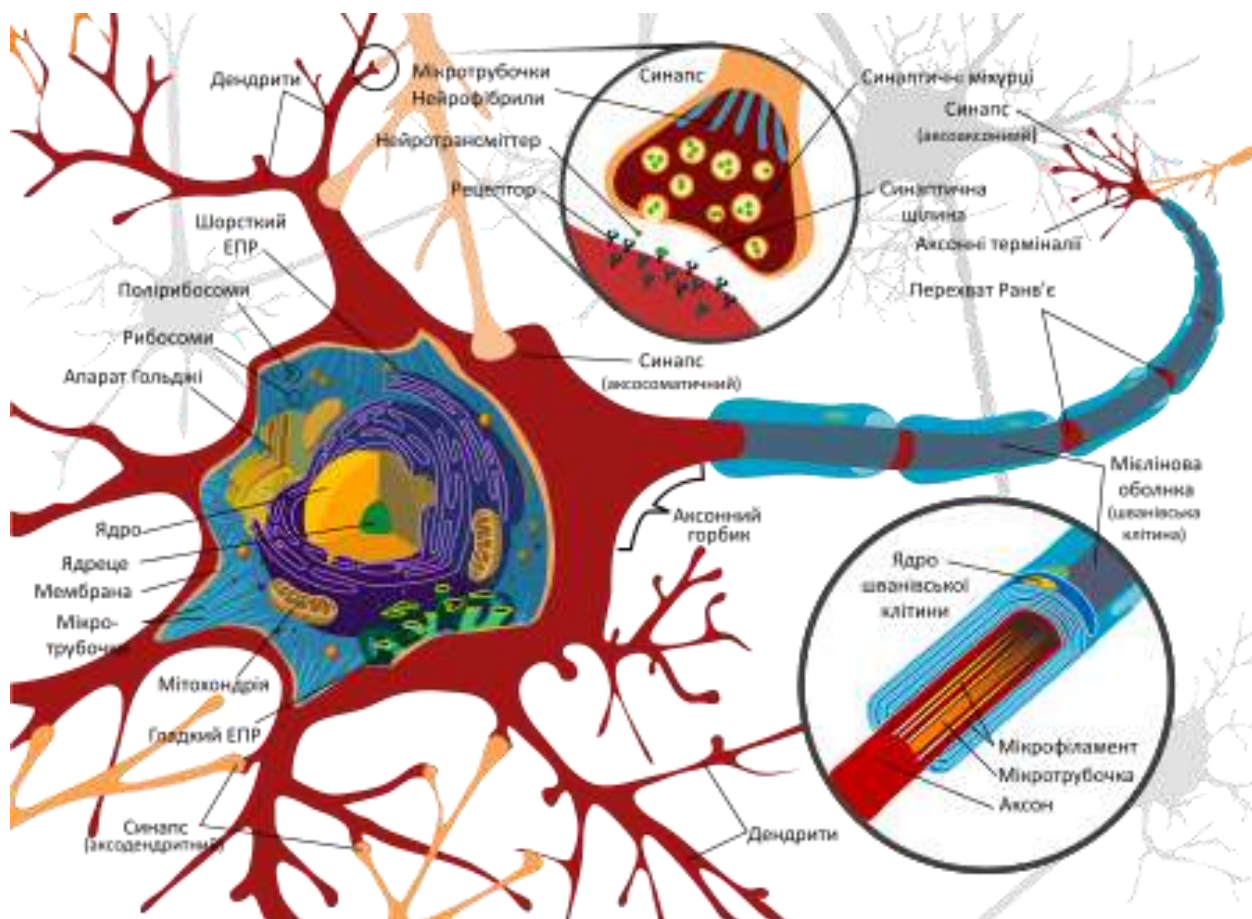


Рисунок 1.4 – Нервова система людини

Мозок обробляє цю інформацію і приймає рішення про те, як зреагувати в даній ситуації. Це рішення передається по нервових волокнах в зворотню сторону до м'язів. Таким чином відбувається обробка інформації в людському тілі. Нервова система складається з декількох компонент. Ці компоненти відповідають за різні завдання. Однак, всі ці компоненти складаються з приблизно однакових клітин які називаються нейронами [31].

Нейрон – це компонента, з якої складається вся нервова система людини. Майже всі нейрони влаштовані приблизно однаково. Отже, у нейрона є ядро, яке називається також тілом нейрона. У тілі нейрона накопичується електричний заряд. З тіла нейрона є безліч відростків. Маленькі відростки називаються дендритами. По цим відросткам проходить сигнал від одного нейрона до іншого. Є великий відросток. Цей відросток називається аксон. По аксону нейрон передає сигнал до інших нейронів. Місце, де аксон одного нейрона з'єднується з дендритами іншого

нейрона, називається "синапсом". Синапс – це місце зіткнення дендрита і аксона. Синапс може бути сильним, а може бути слабким. Якщо синапс сильний, то сигнал, що переданий по аксону нейрона практично повністю перейде до іншого нейрона. Якщо ж синапс слабкий, то практично ніякий заряд не перетне від аксона одного нейрона до другого нейрона. Синапс може змінюватися з часом. Залежно від обставин синапс може ставати сильнішим, а може ставати слабкішим. Саме з налаштуванням синапсу і пов'язане тренування біологічної нейронної мережі і, відповідно, нервової системи людини [33].

У математичної моделі нейрона, тіло нейрона, де накопичується заряд, замінюється на суматор. Крім того, у нейрона є ще й дендрити, по яких сигнал приходить від інших нейронів в тіло нейрона. Відповідно, в математичній моделі нейрона є входи в суматор, які будуть замінювати дендрити. Крім того, у нейрона є аксон, по якому нейрон відправляє сигнали до інших нейронів. Відповідно, в математичної моделі нейрона буде вихід, за яким математична модель нейрона буде відправляти сигнали до інших нейронів. Також варто пам'ятати, що біологічний нейрон накопичує заряд до тих пір, поки цей заряд не досягне якогось значення, і тільки після цього цей заряд йде по аксону до інших нейронів. Це в математичної моделі нейрона робиться за допомогою функції активації. У нейрона є синапси, тобто сила, з якою він з'єднаний з іншими нейронами. Силу синапсів моделюють за допомогою синаптичних ваг, які навішують на входи в нейрон (рис. 1.4) [34].

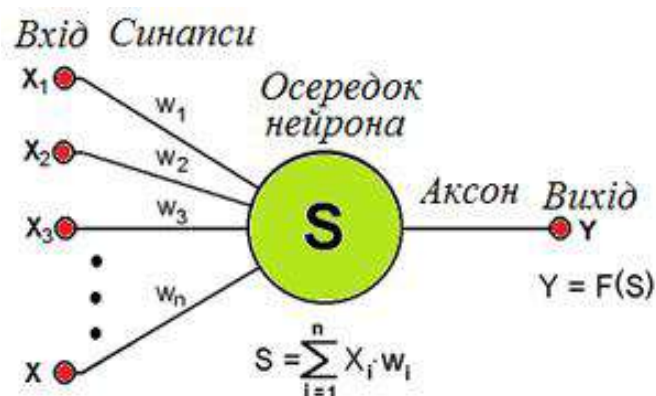


Рисунок 1.5 – Модель штучного нейрона

В математичній моделі нейрона все описується дуже просто (1.1):

$$\sum_{i=1}^n x_i w_i + b \quad (1.1)$$

Вхід в нейрон – це вектор $X = [x_1, x_2, \dots, x_n]$ вхідних значень. Ваги – це вектор параметрів, що навчається. Вони вказують значимість кожної вхідної ознаки для підсумкового результату. Отже, якщо якась вага $w_i = 0$, то ознака x_i є абсолютно незначною для нейронної мережі. І навпаки, чим більше значення, тим сильніше впливає ознака на результат, виданий нейронною мережею. Довжина вектора вагів збігається з довжиною вектора вхідних параметрів. Суматор – функція, яка просто підсумовує добуток ознак на їх ваги. Можна сказати, що це ядро нейрона, яке акумулює заряд. Функція активації – функція, яка приймає на вхід результат суматора і виконує якесь перетворення, щоб перетворити суму зважених входів в адекватний вихід, який можна інтерпретувати з точки зору вирішення поставленого завдання. Так як у випадку малої величини суматорних функції, функція активації може повернути 0, тобто «нічого», то вона є певним аналогом того механізму в реальному нейроні, що відповідає за збудження нейрона (його «активацію») [34]. Для того, щоб описати всі лінійні функції додають зміщення b . Отже, b – це ще один параметр, що настраюється.

Кількість нейронів у шарах в нейронній мережі можна варіювати як завгодно. З чого випливає, що нейронна мережа – досить гнучкий інструмент. Сила нейронних мереж в навчанні. У навчанні вагів зв'язків між нейронами. В ході безлічі ітерацій нейронна мережа порівнює свій вихід зі справжньою відповіддю і, в разі помилки, коригує всі свої ваги таким чином, щоб на наступній ітерації домогтися вірної відповіді. Алгоритм такого навчання називається «алгоритм зворотнього поширення помилки». Це пов'язано з тим, що нейронна мережа розуміє, помилилася вона чи ні, тільки на виході, і потім інформацію про помилку вона поширює в зворотному порядку від виходу до входу [30].

Чим більше в нейронній мережі шарів і нейронів, тим більше зв'язків (а значить, і вагів, що ці зв'язки характеризують) і тим тоншою може бути налаштування нейронної мережі при навчанні. Але також вище буде небезпека підкрутити ваги так, що вони ідеально будуть вирішувати завдання на навчальних даних, але почнуть видавати абсолютно невірну відповідь на нових даних з реального світу. У цьому випадку говорять, що нейронна мережа перенавчилася.

Всього є три важливих типи нейронних мереж:

- повнозв'язні нейронні мережі (fully connected neural networks) (Рис 1.6);
- згорткові нейронні мережі (convolutional neural networks);
- рекурентні нейронні мережі (recurrent neural networks).

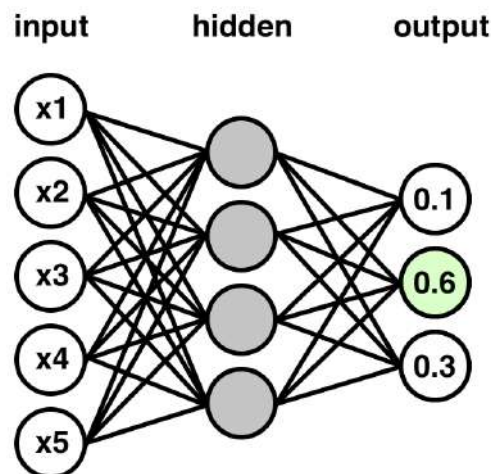


Рисунок 1.6 – Fully connected neural networks

Повнозв'язні нейронні мережі – це ті шари, де всі входи одного рівня підключені до кожного блоку активації наступного рівня. У найбільш популярних моделях машинного навчання останні кілька шарів є повнозв'язними шарами, які об'єднують дані, витягнуті попередніми шарами, для формування остаточного висновку [35].

Згорткові нейронні мережі складаються з декількох шарів штучних нейронів. При подачі значень пікселів штучні нейрони CNN виділяють різні візуальні особливості. Коли вводять зображення в ConvNet, кожен з його шарів генерує

кілька карт активації. Карти активації виділяють характерні риси зображення. Кожен з нейронів приймає в якості вхідних даних фрагмент пікселів, примножує їх значення кольору на свої ваги, підсумовує їх і проганяє через функцію активації. Перший (або нижній) рівень CNN зазвичай виявляє основні особливості, такі як горизонтальні, вертикальні і діагональні краї. Вихідні дані першого шару використовуються в якості вхідних даних для наступного шару, який витягує більш складні елементи, такі як кути і комбінації країв [13]. У міру того, як заглиблюватися в згорткову нейронну мережу, шари починають виявляти високорівневі функції, такі як об'єкти, особи і т. д.

Операція множення значень пікселів на ваги і їх підсумовування називається «згорткою» (звідси і назва). CNN зазвичай складається з декількох згорткових шарів, але також містить інші компоненти. Останній рівень CNN – це рівень класифікації, який приймає вихідні дані останнього згорткового шару в якості вхідних даних (більш високі рівні згортки виявляють складні об'єкти).

На основі карти активації останнього згорткового шару класифікаційний шар виводить набір оцінок достовірності (значення від 0 до 1), які визначають, наскільки ймовірно, зображення належить до «класу». Наприклад, якщо є ConvNet, яка виявляє кішок, собак і коней, виходом останнього шару є ймовірність того, що вхідне зображення містить будь-яке з цих тварин (Рис 1.7) [36]. Всі архітектури, які описані до цього, мали і обробляли інформацію лише про один приклад і не існувало ніякого зв'язку між цими об'єктами. Інформація про один об'єкт ніяк не впливала на обробку іншого.

При обробці тексту, аудіосигналів і часових рядів такий принцип не дуже прийнятний. Тут важливо бачити і враховувати всю предісторію. І чим довше ця предісторія, тим краще. Рекурентна нейронна мережа на момент часу t зберігає всю попередню інформацію про приклади, які пройшли через неї. Але це не абсолютна пам'ять, яка «пам'ятає» все так, як воно було спочатку. Чим довше була попередня послідовність, тим менше залишається інформації про найперших прикладів послідовності. В розумних межах рекурентна нейронна мережа дозволяє

враховувати інформацію про слова, віддалених один від одного дуже далеко. Ефект «пам'яті» досягається за рахунок зворотних блоків [35].

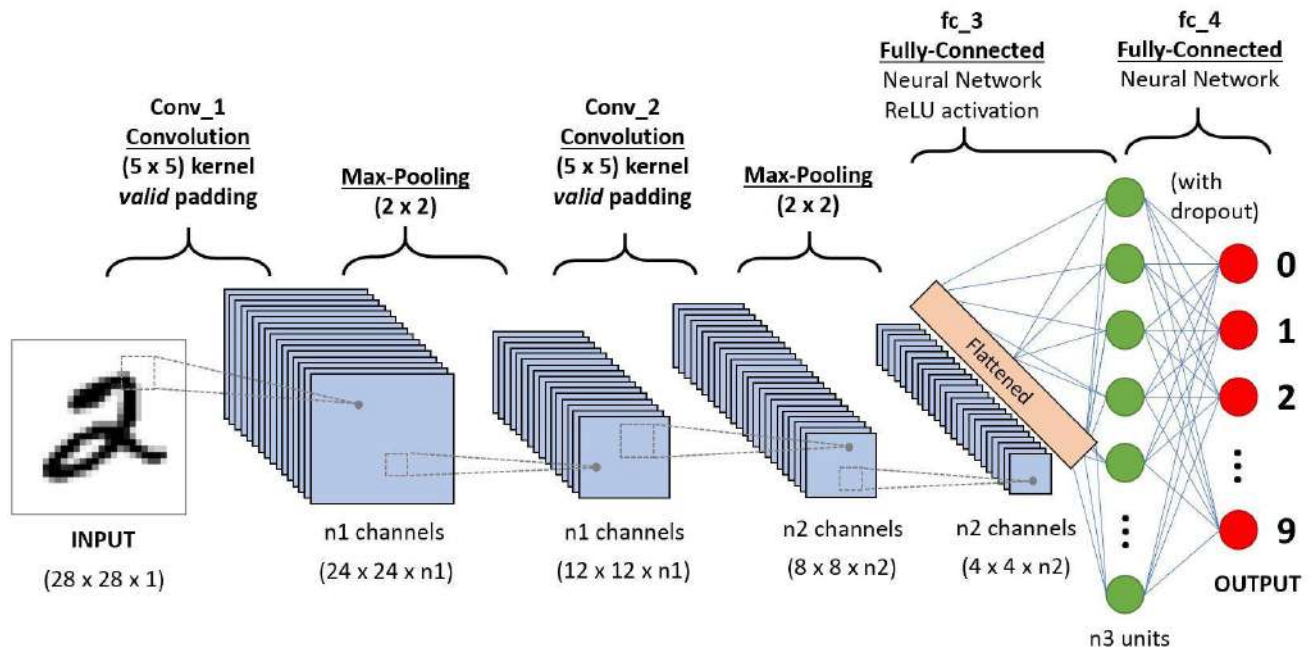


Рисунок 1.7 – Згортова нейронна мережа для класифікації рукописних чисел

Рекурентний шар має зворотний зв'язок, він приймає не тільки вхідні дані, а й дані від самого себе, оброблені в попередній момент часу. По суті, один рекурентний шар, якщо розгорнути його в часі, – це серія звичайних шарів, кожен з яких одночасно приймає на вхід значення нового навчального прикладу і результат обробки попереднього шару. Таким чином, на 1-й шар надійде векторне подання 1-го слова і результат обробки 0-го слова 0-м шаром, а на другий шар потрапить векторне подання 2-го слова і результат обробки 0-го і 1-го слів відповідними шарами. Тільки інформація про 0-ве слово пройде вже через два шари – 0-й і 1-й, а значить, спотвориться більше, ніж інформація про 1-е слово. Значить, кожен наступний блок буде отримувати акумульовану інформацію про всі попередні слова з початку послідовності, але інформація буде тим більш спотворена, чим далі ми пішли від першого слова послідовності (Рис 1.8) [37].

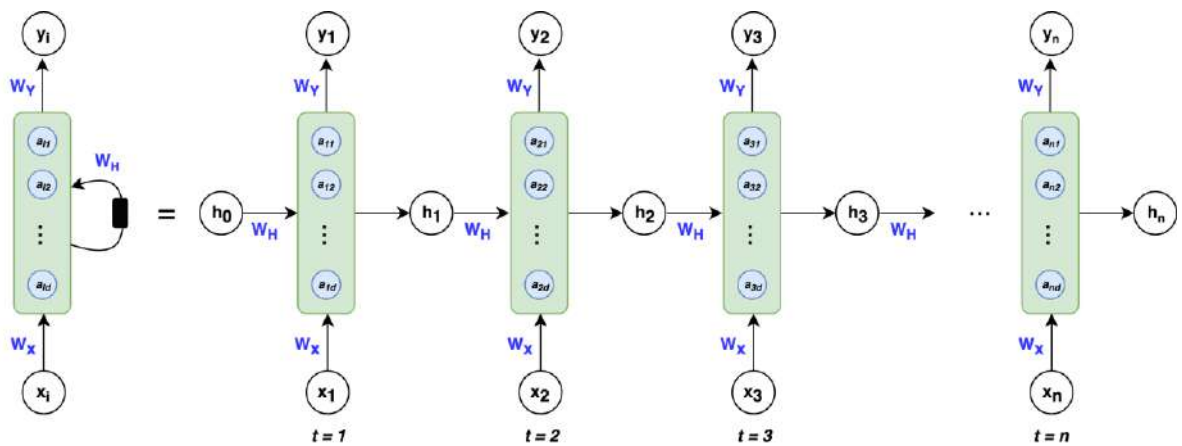


Рисунок 1.8 – Recurrent neural network

З пристрою рекуррентного шару впливає ще один додатковий плюс. Через те, що шар приймає в один момент часу один елемент навчальної послідовності, то йому не важлива довжина вхідної послідовності. Раніше нам завжди потрібно було чітко задати розмір вхідної послідовності. Повнозв'язний шар брав вектор фіксованої довжини, згорткові шари брали матрицю або вектор фіксованого розміру. Рекуррентна нейронна мережа дозволяє не звертати уваги на різницю в довжинах різних послідовностей, що є важливою перевагою. В реальності речення можуть становити як одне слово, так і двадцять [37].

1.3 Аналіз систем розпізнавання обличчя

DeerFace – це система розпізнавання осіб з глибинним навчанням, створена дослідницькою групою в Facebook. Ця система ідентифікує людські обличчя на цифрових зображеннях. DeerFace використовує дев'ятирівневу нейронну мережу з більш ніж 120 мільйонами вагів з'єднань і була навчена на чотирьох мільйонах зображень, завантажених користувачами Facebook. Програмне забезпечення використовує методи 3D-моделювання та штучні нейронні мережі для розпізнавання подібності між двома зображеннями однієї і тієї ж людини, навіть

якщо кут нахилу, освітлення і вираз обличчя не збігаються. За словами дослідників, може визначити, чи являються дві сфотографовані особи однією і тією ж людиною з точністю 97,25%. За даними Facebook, люди в 97,53 відсотках випадків правильно відповідають на один і той же тестовий приклад– всього на чверть відсотка краще, ніж у програмного забезпечення Facebook. Це легка платформа для розпізнавання осіб і аналізу атрибутів людини (вік, стать, емоції і раса). Це гібрид розпізнавання особи сучасними моделями згорткових мереж: VGG-Face, Google FaceNet, OpenFace, Facebook DeepFace, DeepID, ArcFace і Dlib. Бібліотека в основному заснована на Keras і TensorFlow [38].

Сучасний конвеєр розпізнавання осіб складається з 4 загальних етапів: виявлення, вирівнювання, уявлення і перевірка. Deepface обробляє всі ці загальні етапи у фоновому режимі. Функція перевірки в інтерфейсі deepface пропонує перевірити схожість людей. Найголовніша функція Deepface пропонує аналіз особи атрибутів, включаючи age, gender, facial expression (в тому числі гнів, страх, нейтральність, сум, відраза, радість і несподіванка) і race (в тому числі азіатських, білий, близькосхідна, індійський, латино і чорний) передбачення. Функція аналізу в інтерфейсі DeepFace використовується для пошуку демографії особи (рис 1.9) [39]. FaceNet – це система розпізнавання осіб, запропонована дослідниками Google в 2015 році в статті «FaceNet: Unified Embedding for Face Recognition and Clustering». Вона досягла найсучасніших результатів на багатьох тестових наборах даних розпізнавання осіб, таких як Labeled Faces in the Wild (LFW) і Youtube Face Database.



Рисунок 1.9 – Deepface розпізнавання

Вони запропонували підхід, при якому система генерує високоякісне відображення особи із зображень з використанням архітектур глибокого навчання, таких як ZF-Net і Inception. Потім система використовує метод, так званий триплетними втратами, в якості опції втрат для навчання цієї архітектури. По суті, ідея розпізнавання особи полягає в поданні двох зображень у вигляді векторів меншого розміру і визначенні ідентичності на основі подібності, як в оксфордському VGG-Face (рис. 1.10) [40].

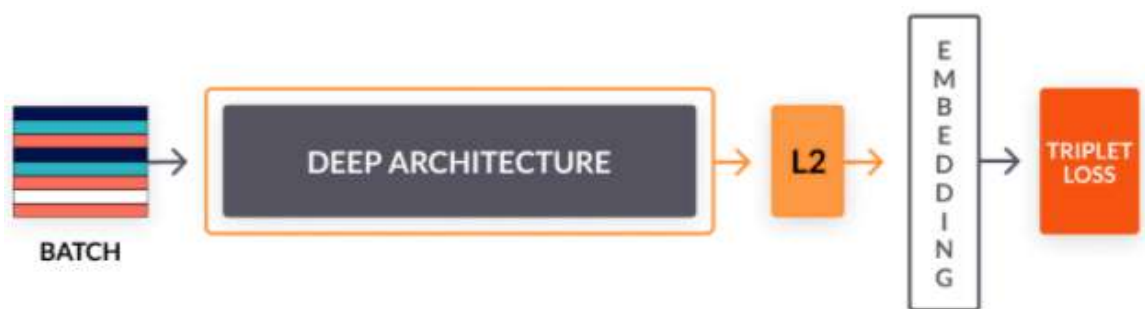


Рисунок 1.10 – Архітектура FaceNet

Згортова нейронна мережа FaceNet покладається на пікселі зображення як на функції, а не на їх вилучення вручну. Основна ідея алгоритму полягає в тому, щоб представити обличчя у вигляді 128-мірного вкладення, зіставляючи вхідні характеристики з векторами (рис. 1.11) [41].



Рисунок 1.11 – Розпізнавання FaceNet

Оскільки ці векторні вкладення представлені в загальному векторному просторі, векторну відстань можна використовувати для обчислення подібності між двома векторами. Це метод розрахунку того, наскільки схожі дві особи. Останнім етапом в архітектурі FaceNet є «потрійна втрата», яка зводить до мінімуму відстань між якорем і відомим позитивним елементом (схожість між двома гранями), в той же час максимізує відстань між якорем і відомим негативним елементом (несхожість).

2 РОЗРОБКА СИСТЕМИ РОЗПІЗНАВАННЯ ЕМОЦІЙ

Система розпізнавання емоцій – це система, що здатна ідентифікувати емоції людини через зображення або відеокамеру. Емоції – це процес, який відображає суб'єктивне ставлення людини до стимулу. Саме емоції спонукають людину до дії, визначають стан її здоров'я, а часом і блокують мислення. Завдяки емоціям ми розуміємо, який психічний стан має людина, на основі цих висновків ми обираємо подальшу модель поведінки [1]. Емоції – це також сигнал бажання та потреби людини у чомусь. Емоції можуть врятувати життя під час надзвичайної ситуації. Завдяки розвитку технологій комп'ютерного зору розпізнавання емоцій стало більш точним та доступним для широкого кола людей

Система ґрунтується на порівнянні поточної емоції з тими, що зберігаються в базі. Дана система складається з 2-ох частин. Спочатку на вхід системи подається зображення чи відео. Якщо система знайшла область обличчя людини, зображення перетворюється та нормалізується щоб системі було легше робити порівняння. Далі нейронна мережа оброблює проходить по нормалізованій області обличчя та робить визначення до якого типу емоції належить вираз обличчя людини.

Узагальнену блок-схему алгоритму роботи системи розпізнавання емоцій наведено на рисунку 2.1.



Рисунок 2.1 – Блок-схема алгоритму роботи системи розпізнавання емоцій

2.1 Інструментальні засоби розробки нейронних мереж

Для програмування глибинної нейронної мережі було обрано мову програмування Python. Python – це високорівнева мова програмування, що була розроблена для зручності читання. У Python часто використовуються англійські ключові слова, тоді як в інших мовах використовуються розділові знаки, і в цій мові програмування менше синтаксичних конструкцій, ніж в інших мовах.

Головні характеристики Python:

- підтримка методів функціонального та структурного програмування, а також ООП;
- велика кількість корисних бібліотек;
- динамічні типи даних дуже високого рівня і підтримка перевірки динамічних типів;
- підтримка автоматичного збору сміття;
- легка інтеграція із такими мовами програмування як C, C ++, COM, ActiveX, CORBA и Java.

Python пропонує лаконічний і читабельний код. У той час як за машинним навчанням і штучним інтелектом стоять складні алгоритми і універсальні робочі процеси, простота Python дозволяє розробникам створювати надійні системи. Розробники повинні докласти всіх зусиль для вирішення проблеми машинного навчання замість того, щоб зосередитися на технічних нюансах мови. Щоб скоротити час розробки, програмісти звертаються до ряду фреймворків і бібліотек Python. Програмна бібліотека – це попередньо написаний код, який розробники використовують для вирішення спільних завдань програмування. Python з його багатим стеком технологій має великий набір бібліотек для штучного інтелекту і машинного навчання [41]. Ось деякі з них:

- Keras, TensorFlow і Scikit-learn для машинного навчання;
- NumPy для високопродуктивних наукових обчислень і аналізу даних;

- Matplotlib для побудови зрозумілих графіків;
- Pandas для універсального аналізу даних.

Спочатку TensorFlow був дослідницьким проектом глибинного навчання групи Google Brain Team, який з тих пір став – завдяки співпраці з 50 командами в Google – бібліотекою з відкритим вихідним кодом, що була розгорнута в екосистемі Google. AI став вирішальним в еволюції взаємодії користувачів зі службами і пристроями (рис. 2.2) [42].

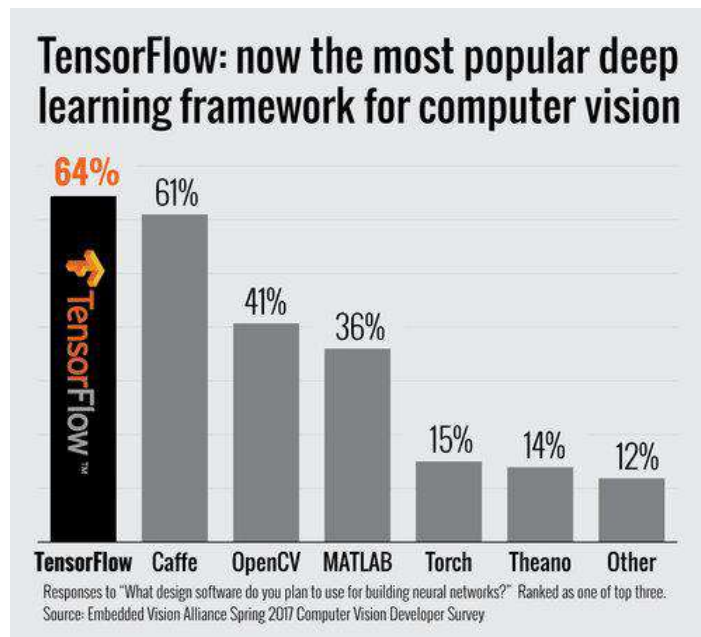


Рисунок 2.2 – Порівняння популярності з іншими бібліотеками

Наявність такого потужного набору бібліотек може дозволити розробникам включити цю потужну еволюцію глибокого навчання в свої продукти. Tensorflow підтримує глибоке навчання, нейронні мережі і загальні чисельні обчислення на процесорах, графічних процесорах і кластерах графічних процесорів. Поточну версію TensorFlow можна знайти на GitHub разом з примітками до випуску. TensorFlow – безумовно, найпопулярніший двигун AI, який використовується сьогодні.

Також важливим інструментом для DL/ML розробника є Google Colab. Це середовище виконання на основі Jupyter Notebook, що дозволяє запускати код

повністю на хмарі. Нейронні мережі потребують багато обчислювальних ресурсів щоб навчати масштабні моделі, але не у всіх є доступ до потужних GPU та TPU. Google Colab підтримує екземпляри як GPU, так і TPU, що робить його ідеальним інструментом для ентузіастів глибокого навчання і аналізу даних через обчислювальні обмеження на локальних машинах. Оскільки до ноутбука Colab можна отримати віддалений доступ з будь-якого комп'ютера через браузер, він також добре підходить для комерційних цілей [43].

NumPy – це бібліотека Python, яка надає просту, але потужну структуру даних: n-мірний масив. Це фундамент, на якому побудована майже вся міць інструментарію Python для науки про дані. NumPy використовує алгоритми, написані на C, які виконуються за наносекунди, а не за секунди. NumPy допомагає скоротити кількість циклів і уникнути плутанини в індексах ітерацій. Тисячі розробників працюють над тим, щоб NumPy залишався швидким, доброзичливим і вільним від помилок [43].

Pandas – це потужний інструмент, який дозволяє перетворювати JSON, CSV, масиви, словники та інші дані в формат рядків і стовпців. Pandas схожий на електронну таблицю, але потрібно працювати з нею, використовуючи код, а не Microsoft Excel. Pandas робить надзвичайно складні перетворення даних легкими і природними. Також включає в себе безліч математичних, аналітичних та інших функцій. Matplotlib – це комплексна бібліотека для створення статичних, анімованих та інтерактивних візуалізацій в Python. Matplotlib видає показники якості публікації в різних форматах друкованих копій і в інтерактивних середовищах на різних платформах. Matplotlib можна використовувати в сценаріях Python, серверах веб-додатків і різних інструментах графічного призначеного для інтерфейсу користувача.

OpenCV – це провідна бібліотека з відкритим вихідним кодом для комп'ютерного зору, обробки зображень і машинного навчання, також підтримує прискорення графічного процесора для роботи в реальному часі. OpenCV безкоштовна як для академічного, так і для комерційного використання. Має

інтерфейси C ++, C, Python і Java і підтримує Windows, Linux, Mac OS, iOS і Android. OpenCV була розроблена для обчислювальної ефективності і з упором на додатки реального часу, бібліотека може використовувати переваги многоядерной обробки.

2.2 Алгоритм розпізнавання області обличчя

Визначення області обличчя відноситься до комп'ютерної технології, яка здатна визначати присутність осіб людей на цифрових зображеннях. Для роботи програми розпізнавання осіб використовують машинне навчання і формули, відомі як алгоритми для виявлення людських облич на великих зображеннях. Ці великі зображення можуть містити безліч об'єктів, які не є особами, наприклад пейзажі, будівлі та інші частини людей (наприклад, ноги, плечі і руки).

Виявлення області – це більш широкий термін, ніж розпізнавання обличчя. Виявлення обличчя просто означає, що система здатна ідентифікувати людське обличчя на зображенні або відео. Розпізнавання обличчя також можна використовувати для автофокусування камер, для підрахунку кількості людей, які увійшли в певну зону. Його навіть можна використовувати в маркетингових цілях. Наприклад, реклама може відображатися в момент розпізнавання особи [44].

Хоча процес досить складний, алгоритми розпізнавання обличчя часто починаються з пошуку людських очей. Очі є так звана ключова точка, і їх найлегше виявити. Після виявлення очей алгоритм може спробувати виявити ділянки обличчя, включаючи брови, рот, ніс, ніздрі і райдужну оболонку. Як тільки алгоритм передбачає, що він виявив область особи, він може застосувати додаткові тести, щоб перевірити, чи дійсно він виявив область [45].

Каскади Хаара є найбільш поширеним алгоритмом в області комп'ютерного зору та використовується для виявлення обличчя на зображенні. Алгоритм Хаара

використовується не тільки для виявлення осіб на зображеннях, але також можна навчити модель виявляти різні об'єкти, такі як автомобілі, будівлі, кухонні інструменти. Алгоритм використовує функції виявлення країв або ліній. Ідея полягає у виборі декількох характеристик (на рисунку 2.3 їх 14) [46].

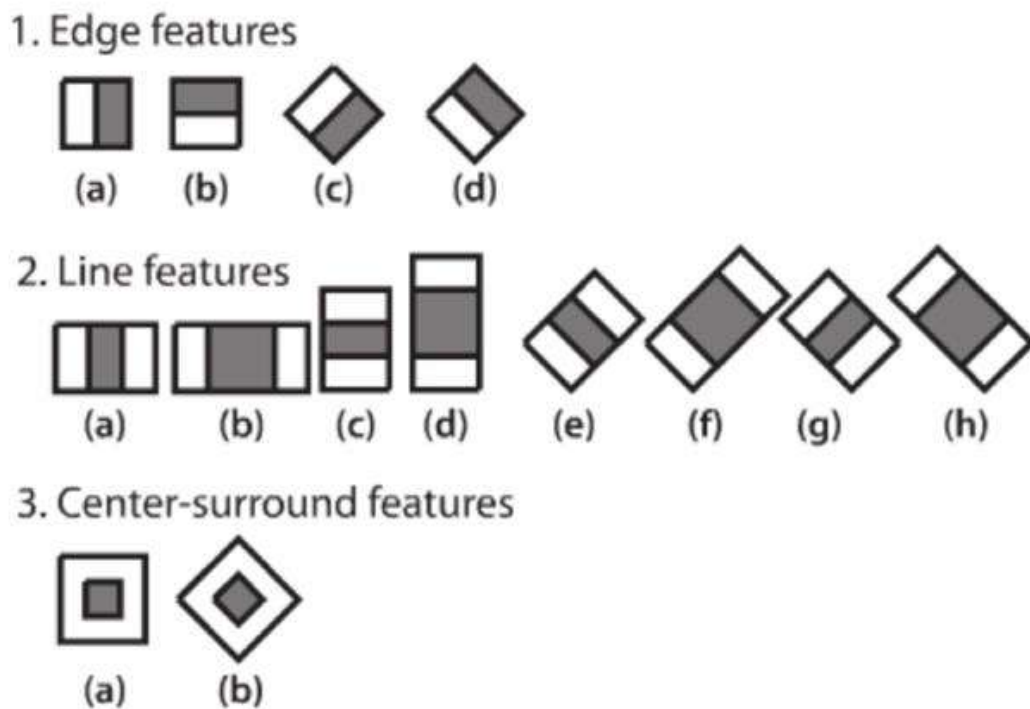


Рисунок 2.3 – Ознаки каскадів Хаара

Припустимо у нас є картинка і на цій картинці ми шукаємо, чи є для кожної ознаки відповідні частини картинки, чи ні. Ознаки Хаара безперервно переміщуються від верхнього лівого кута зображення до нижнього правого, щоб знайти конкретну особливість. Тобто ми перевіряємо, чи є на зображенні відповідність однієї з ознак, чи проявилась одна з цих ознак на зображенні. Подивимося на рисунок 2.4 [47] і побачимо, що ознака 2a (рис 2.3) вказує на наявність носа, 1b – це очі, 1a знайшов вилицю тощо. Коли всі відповідності кожної ознакою знайдені і ці відповідності знайдені одночасно і саме при такому взаємному розташуванні (рис. 2.4), можна розпізнати людське обличчя. Тобто правильна комбінація цих ознак вказує на наявність області обличчя [48].

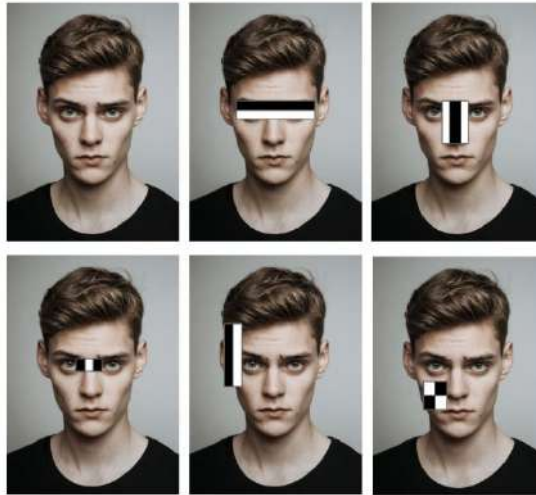


Рисунок 2.4 – Пошук ознак на зображенні

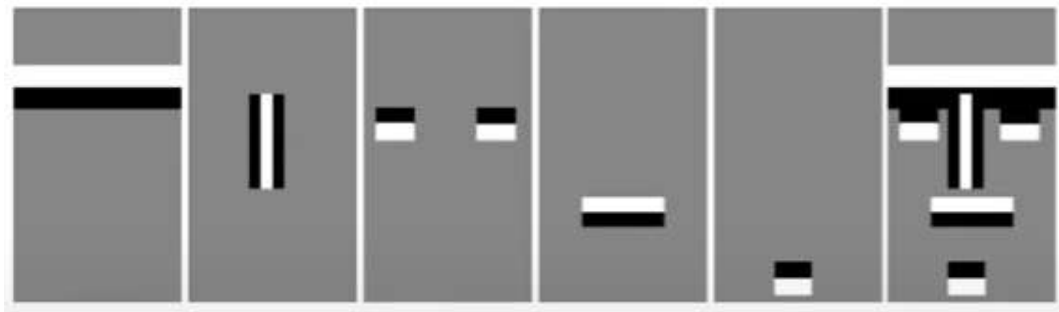


Рисунок 2.5 – Комбінація ознак

OpenCV вже містить безліч попередньо навчених класифікаторів для обличчя, очей, посмішки тощо. Для цього потрібно використати xml файл та функцію:

`detectMultiScale (image, objects, scaleFactor = 1.1, minNeighbors = 3, flags = 0, minSize = new cv.Size (0, 0), maxSize = new cv.Size (0, 0))`, де

`image` – матриця типу `CV_8U`, що містить зображення, де виявляються об'єкти;

`objects` – вектор прямокутників, де кожен прямокутник містить виявлений об'єкт;

`scaleFactor parameter` – вказує, на скільки зменшується розмір зображення при кожному масштабі зображення;

`minNeighbors` – параметр, що вказує, скільки сусідів повинен мати кожен прямокутник-кандидат, щоб зберегти його [49].

2.3 Алгоритм розпізнавання емоцій

Одним із класичних способів визначення емоцій є алгоритм заснований на класифікації ключових точок на зображенні обличчя. Алгоритм розділений на два етапи.

1. Етап навчання, на якому за вихідними даними будується (навчається) модель розподілу ключових ознак [50].

2. Етап застосування, при якому навчальна модель застосовується до реальних даних, у результаті чого вхідні образи відносяться до одного із заданих класів [50].

На вхід алгоритму на етапі навчання навчання поступає масив ключових точок, які наведено на рисунку 2.6, що описує найбільш характерні риси людського обличчя, а саме, брови, очі, ніс, губи, контур обличчя.

За цими даними будується дві статистичні моделі.

Модель форми (2.1) – параметричну лінійна модель, що описує можливі варіанти положення ключових точок:

$$s = [x_1, y_1, \dots, x_n, y_n]^T \quad (2.1)$$

Форма – набір точок на зображенні, що становлять контури досліджуваного об'єкта. Наприклад, форма кисті руки людини може бути задана через точки на контурах долоні і пальців, а форма особи – через точки на бровах, носі, губах і зовнішньому контурі [51].

Модель текстури (2.2) – подібна модель, але описує вже можливі варіації інтенсивності пікселів:

$$s = [t_1, \dots, t_n]^T \quad (2.2)$$

Текстура – інтенсивність пікселів на зображенні всередині зовнішнього контуру форми.

Кількість пікселів всередині зовнішнього контуру форми на різних зображення може відрізнятися, тому всі зображення осіб наводяться до єдиної нормалізованої форми за допомогою кусково-афінного перетворення [51].

На сьогоднішній день ефективність даного методу складає 89,4%. Існує велика кількість алгоритмів, що дозволяють отримати ключові точки з точністю, достатньою для подальшої класифікації за цими точкам емоцій людини. Але, для використання такого підходу необхідно щоб положення особи на зображенні було вирівняно, а в реальних умовах це майже неможливо [51].

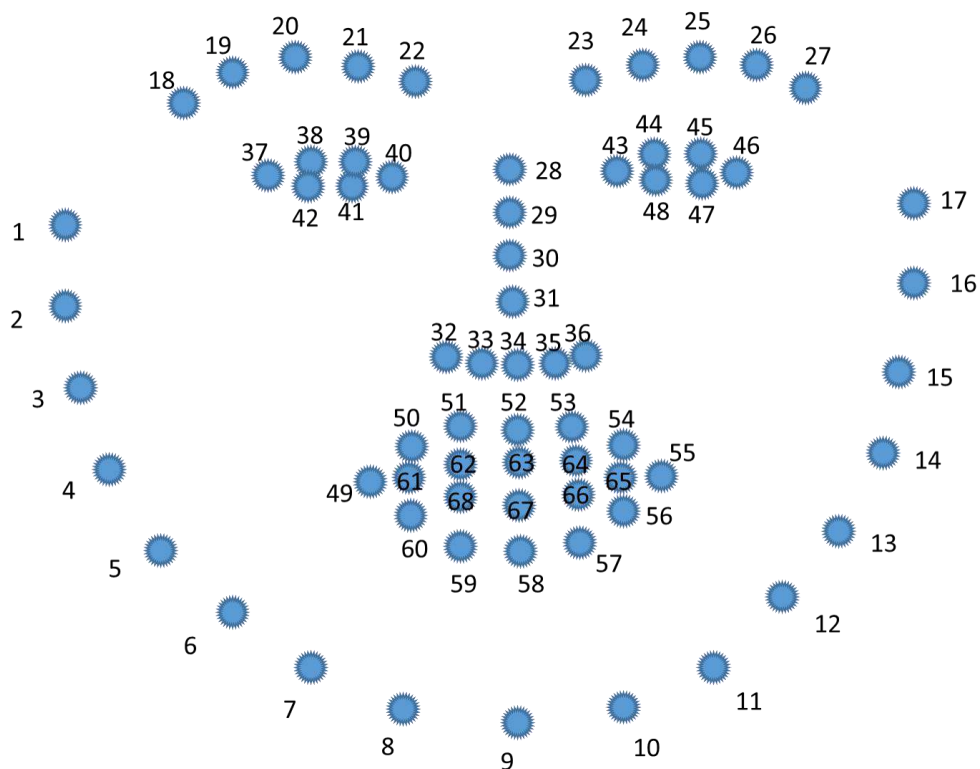


Рисунок 2.6 – Ключові точки

Альтернативою методу ключових точок є використання згортальних нейронних мереж.

2.3.1 Згорткові нейронні мережі

Згорткові нейронні мережі (ConvNet / CNN) є алгоритмом Deep Learning, який може приймати вхідне зображення, призначати важливість (ваги та зміщення) різним об'єктам на зображенні і мати важливість диференціювати одне від одного. Попередня обробка набагато нижча в порівнянні з іншими алгоритмами класифікації [52]. У той час як в примітивних методах характеристики створюються вручну при достатньому навчанні, ConvNets можуть вивчити ці характеристики .

Головна перевага таких мереж від повнозв'язних є зменшення кількості нейронів, що навчаються, а як наслідок, прискорення навчання мережі та зменшення необхідної кількості навчальних даних. За рахунок великої кількості абстрактних шарів, вони забезпечують часткову стійкість до змін масштабу, зсувів, поворотів, зміні ракурсу і іншим спотворенням. Зручне розпаралелювання обчислень, а, отже, можливість реалізації алгоритмів роботи і навчання мережі на графічних процесорах [53] (рис. 2.7).

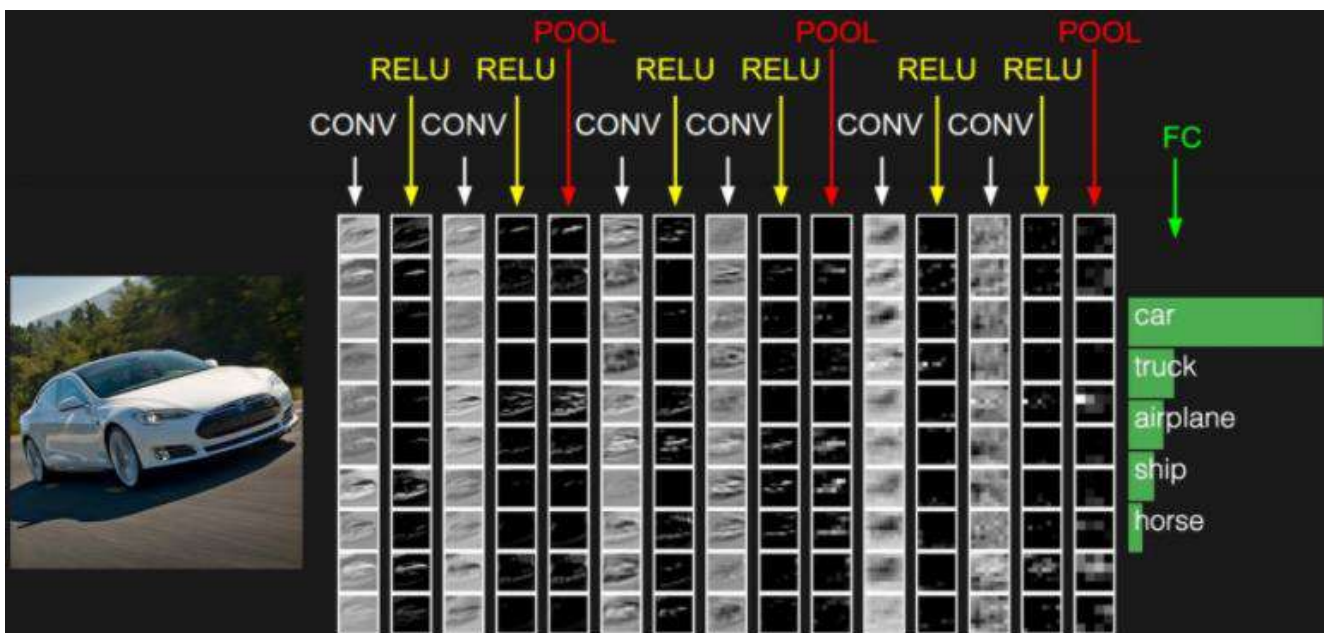


Рисунок 2.7 – Приклад архітектури згорткової мережі

ConvNet представляє собою послідовність рівнів, і кожен рівень ConvNet перетворює один обсяг активацій в інший за допомогою функції, що диференціюється. Всього використовується три основних типи шарів для побудови архітектур ConvNet: шар згортки (конволюції), шар пулу і повнозв'язний шар (як в звичайних нейронних мережах) [52]. Ці шари складаються, щоб сформувати повну архітектуру ConvNet.

Саме від операції згортки ці мережі і отримали свою назву. Для того, щоб зробити згортку потрібно, так зване, ядро згортки. Це матриця такого ж розміру, як і регіон зображення, до якого підключений нейрон. У нашому випадку три на три. Операція згортки виконується так: потрібно взяти лівий верхній елемент в ядрі згортки і помножити його на лівий верхній елемент в регіоні зображення, до якого підключений нейрон. Потім беремо середній елемент в першому рядку ядра згортки і множимо його на середній елемент в першому рядку вхідних сигналів, до яких підключений нейрон. Потім повторюємо цю операцію для всіх елементів ядра згортки і відповідних їм елементів вхідних сигналів. Результати складається і підсумковий результат записується в згортковий шар. Значення в інших нейронах згорткового шару розраховуються за схожою формулою, тільки кожен нейрон підключений до свого регіону вхідних сигналів. Операція згортки використовується не тільки в нейронних мережах, але і в класичних алгоритмах комп'ютерного зору. Тому було придумано багато ядер згортки, які дозволяють реалізувати корисні операції обробки зображень (рис. 2.8) [53].

Перевага згорткової нейронної мережі у тім, що в ній не треба вручну задавати ядра згортки, вони визначаються автоматично в процесі навчання нейронної мережі. Ще один тип шару, який використовуються в згорткових мережах, це шар підвибірки. Нейрони цього шару, також як і шару згортки, підключені не до всіх нейронів попереднього шару, а до обмеженого регіону. Найчастіше використовуються регіон 2 на 2, але іноді зустрічаються і інші варіанти. Шар підвибірки не навчається, він виконує якусь задану операцію, і зараз найбільш популярна операція вибору максимального значення [54].

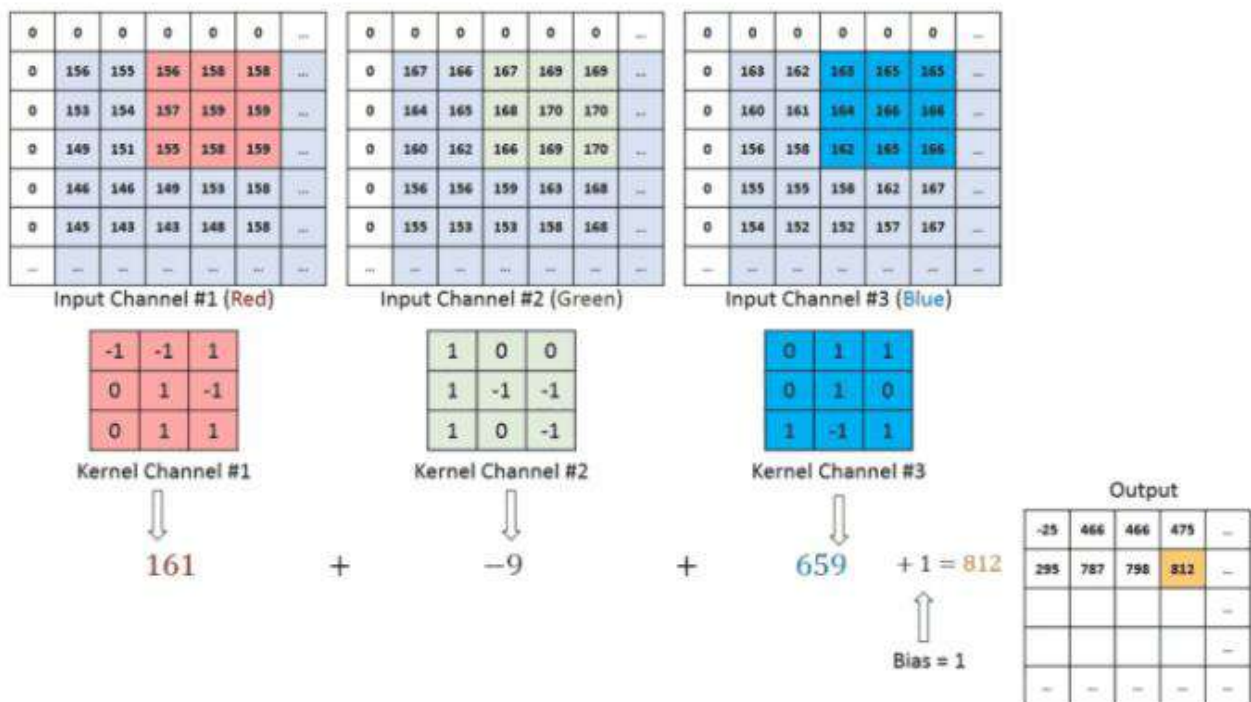


Рисунок 2.8 – Приклад операції згортки

Також можна вибрати середнє значення або виконувати якусь іншу операцію. Для багатьох завдань комп'ютерного зору факт наявності ознаки набагато важливіший, ніж, конкретно, його положення. Max Pooling повертає максимальне значення з частини зображення, покритої ядром. З іншого боку, Average Pooling повертає середнє значення всіх значень з частини зображення, покритої ядром. Max Pooling також діє як пригнічувач шуму. Він повністю відкидає шумні активації, а також виконує усунення шуму разом зі зменшенням розмірності. З іншого боку, Average Pooling просто виконує зменшення розмірності як механізм придушення шуму [54]. Отже, ми можемо сказати, що Max Pooling працює набагато краще, ніж Average Pooling (рис. 2.9).

Повнозв'язний шар вивчає, можливо, нелінійну функцію в цьому просторі. Тепер, коли вхідне зображення перетворено в потрібну форму для багаторівневого персептрона, потрібно згладити зображення в вектор-стовпець. Згладжений висновок подається в нейронну мережу з прямим зв'язком, і зворотнє поширення застосовується до кожної ітерації навчання. Протягом ряду епох модель здатна

розрізняти домінуючі і деякі низькорівневі функції на зображеннях і класифікувати їх за допомогою техніки Softmax Classification.

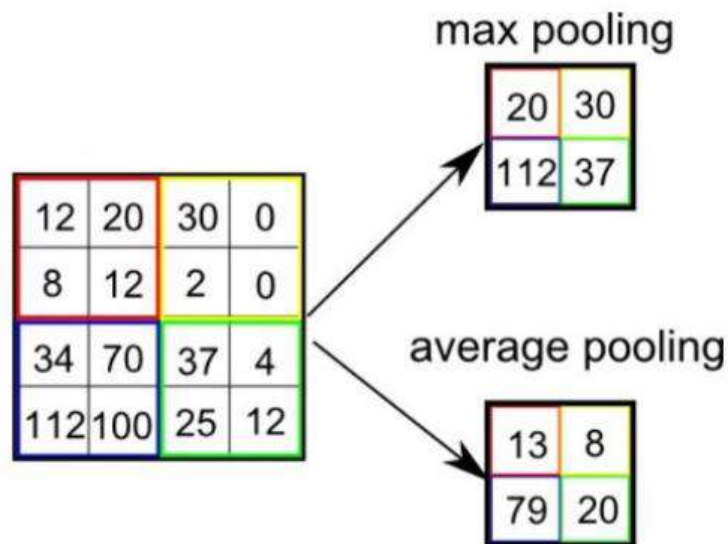


Рисунок 2.9 – Типи підвибірки

2.3.2 Збір навчальних даних та їх нормалізація

Добре підготовлені дані підвищують шанси на якісне навчання нейронної мережі. На практиці, якість моделі залежить більше від підготовлених даних, ніж від оптимізації моделі. Сучасні нейронні мережі здатні самотійно знаходити нелінійну закономірність в даних, тому етап пошуку ознак можна пропустити. Але для цього даних потрібно багато, десятки тисяч і вони повинні бути з лейблами. Також основною вимогою в даних є інформативність. Тобто якщо стоїть завдання класифікувати види тварин, потрібен датасет з видами тільки цих тварин, нічого зайвого. Для початку розробки нейронної мережі знадобиться достатня кількість маркованих даних для навчання, що включають 6 класів основних емоцій П. Екмана. Великі компанії мають ресурси і можливість самотійно збирати дані і позначати їх. Але є загальнодоступні бази даних, які містять базові вирази і широко використовуються для оцінки алгоритмів глибокого навчання. В даній роботі розглянуто основні бази емоцій.

Розширений CohnKanade (СК +) є найбільш широко використовуваною базою даних для оцінки емоцій з лабораторним контролем. Містить близько 600 відео та 5876 зображень із тегами 123 людей у віці від 18 до 50 років. Послідовності мають тривалість від 10 до 60 кадрів, ці кадри показують перехід від нейтральних емоцій до найбільш виражених. На кожному відео позначено сім основних емоцій Екмана: гнів, презирство, огида, страх, щастя, здивування, смуток. Містить в собі постановочні емоції та емоції з реального життя, спонтанні. Зображення переважно одноканальні з однаковим фоном та розміром 640 * 490 пікселів. Але СК + не надає набору даних для алгоритмів навчання, перевірки та тестування. Через це використовувати цей набір даних у роботі неможливо.

ЕМОТІС або ЕМОТІон цей унікальний набір даних містить зображення людей в різноманітних ситуаціях у їхньому природному середовищі та позначений очевидними емоціями, які інтуїтивно ідентифікуються. Більшість зображень зібрані вручну з Інтернету та пошукових запитів. База даних містить близько 30000 зображень різних людей, емоції розділені на 2 типи, 26 дискретних категорій. У тривимірній системі координат емоцій, валентності, збудження та домінування. Цей проект був створений для того, щоб машини зрозуміли, що переживає людина на основі системи координат. Машини з цими можливостями краще взаємодіють з людьми.

Однією з найновіших баз зображень є RAF-DB. Містить в собі 29672 прикладів помічених кольорових зображень. Зображення в основному 100*100 пікселів, але ширина та висота деяких картинок є різною, що не є плюсом для моделі. Зображення як спонтанні так і постановлені, мають 6 класів основних емоцій та 12 класів важких емоцій: лякання, здивування, жахлива огида, сумний гнів, сумний переляк, гнівна відраза, сердите здивування, сумна відраза, гидливе здивування, щасливе здивування, сумне здивування, страшенна злість та щастя що відчуває відразу [55].

FER-2013. Це класична база даних, яка була створена для змагань на платформі машинного навчання Kaggle, і після змагань, дані були опубліковані для

вільного користування. У наборі даних є 35000 зображень. Дані поділяються на дані про навчання, приватні тести та публічні тести. Зображення позначені міткою із 7 семи основних емоцій: щастя, страх, гнів, здивування, огида, нейтральність, смуток. Зображення перетворено в масив чисел від 0 до 255 і розміщено у файлі .csv. Виходячи з цього, очевидна перевага датасету є використання малого обсягу пам'яті. Крім того, цей набір даних можна вільно завантажити на Google Drive. Це заощадить простір. На жаль, розмір зображень становить лише $48 * 48$ пікселів, але всі зображення були зареєстровані автоматично, обличчя знаходиться в центрі і займає однаковий простір на кожному зображенні, що допоможе нейронній мережі легше знайти ознаки.

Після вибору датасету, ці дані необхідно нормалізувати. Нормалізація – це процес попередньої обробки даних після якого значення ознак об'єктів на вхідному векторі буде у діапазоні від 0 до 1. Початковий набір даних має вектор ознак від 0 до 255, але для нейронної мережі такі показники можуть викликати дисбаланс між значеннями ознак. В подальшому це може призвести до нестійкості роботи моделі, погіршити результати навчання і уповільнити процес моделювання.

Спочатку потрібно розглянути структуру даних за допомогою бібліотеки Pandas. У fer2013.csv є 3 колонки (рис. 2.9). У першому стовпці «emotion» – номер емоції, у другому «pixels» – набір масиву пікселів зображень, а в третьому стовпці «Usage» вказується, до якого типу даних належить зображення. Зображення являють собою масиви NumPy $28*28$ зі значеннями пікселів від 0 до 255.

Мітки – це масив цілих чисел в діапазоні від 0 до 6. Вони відповідають класу емоцій, які зображує зображення: 0=Angry, 1=Disgust, 2=Fear, 3=Happy, 4=Sad, 5=Surprise, 6=Neutral. Всього існує три типи вибірки: навчальна, тестова та валідаційна. Навчальна підвибірка містить стандарт, вхідні дані разом із правильним / очікуваним результатом. Модель не бачить тестового набору даних під час навчання, і тому на цьому наборі даних можна оцінити якість моделі. Для підбору гіперпараметрів та вибору найкращої моделі необхідна валідаційна підвибірка.

	emotion	pixels	Usage
0	0	70 80 82 72 58 58 60 63 54 58 60 48 89 115 121...	Training
1	0	151 150 147 155 148 133 111 140 170 174 182 15...	Training
2	2	231 212 156 164 174 138 161 173 182 200 106 38...	Training
3	4	24 32 36 30 32 23 19 20 30 41 21 22 32 34 21 1...	Training
4	6	4 0 0 0 0 0 0 0 0 0 0 3 15 23 28 48 50 58 84...	Training

Рисунок 2.10 – Структура датасету

Дані поділено на 3 підвибірки. Є 3 класи стовпця "Usage": "Train" містить 28709 зображень, "PublicTest" містить 3589 зображень, "PrivateTest". Дані зі стовпця "Train" можна використовувати для навчання моделі, дані з "PrivateTest" для перевірки моделі та "PublicTest" для тестових зображень. Дані потрібно трансформувати так, щоб вони мали таку структуру: вектор ознак $X_i = [x_1, \dots, x_n]$ та мітка (цільова змінна) Y_i (рис. 2.11).

Графік показує кількість зображень, що належать до кожного класу, у навчальній та тестовій підвибірці.

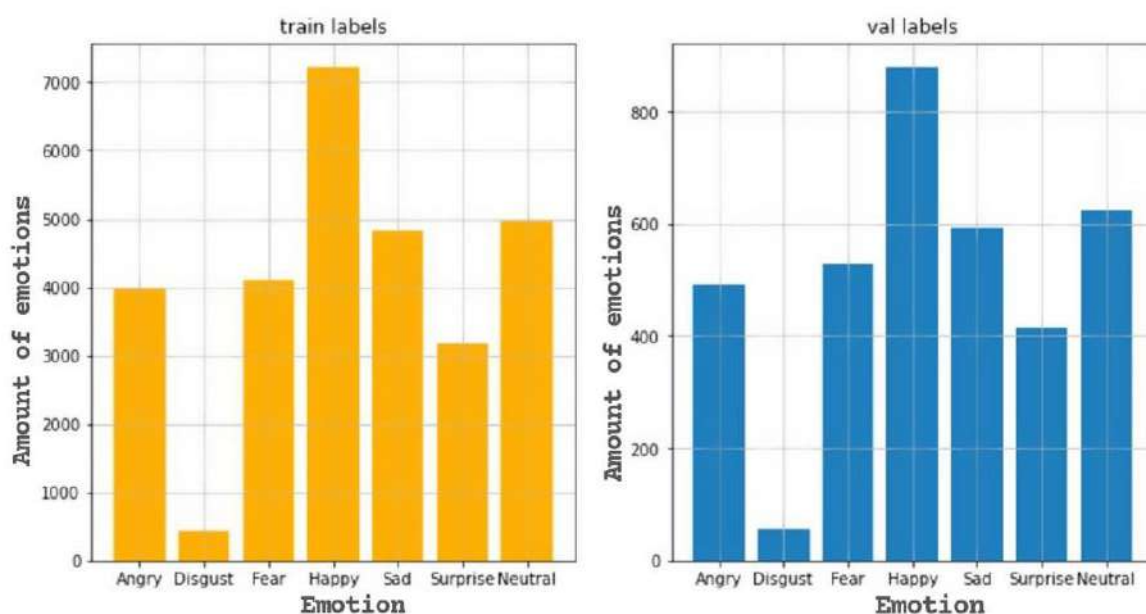


Рисунок 2.11 – Розподілення даних між підвибірками

Оскільки значення пікселів знаходяться в діапазоні від 0 до 255, ці значення потрібно масштабувати (рис. 2.12). Для цього потрібно розділити масив зображень на 255. Важливо, щоб навчальний набір і набір тестів були попередньо оброблені однаково. Крім того, дані потрібно перетворити на тип даних float32. Далі дані повинні бути перетворені в розмір $48 * 48$, а кількість каналів має бути 1.

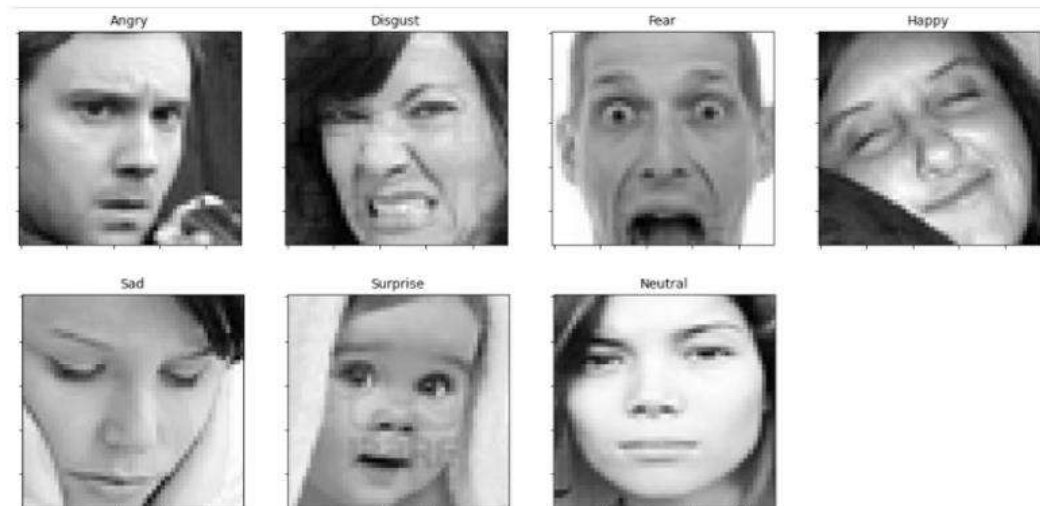


Рисунок 2.12 – Приклад об’єктів даних

У кольорових зображеннях кількість каналів становить 3, а в чорно-білих 1. Масив міток перетворюється в одно-вимірний вектор. Метод `to_categorical()`, перетворює масив або вектор із цілими числами з різними категоріями в новий масив із двійковими значеннями та стовпцями, рівними кількості категорій у даних. Вище представлено нормалізований вхідний вектор даних, перетворений з масиву в зображення (рис. 2.12).

2.3.3 Архітектура згорткової нейронної мережі

Загалом нейронна мережа матиме 3 згорткові шари, 3 шари з макспулінгом, шар сплюснення та 2 повні повнозв’язні шари (рис. 2.13). Перший шар буде згортковим. Саме він відповідає за операцію згортки та вибір ядер згортки. Існують одновимірні та двовимірні аналоги. На першому згортковому шарі кількість різних ядер згортки, що застосовуватиметься, буде дорівнювати 32. У той же час це кількість карт ознак, які буде отримано на наступному шарі після згортки. Далі йде

розмір ядра згортки. Найчастіше використовуються розміри 3×3 , 5×5 і 7×7 , але можуть бути й інші, включаючи неквадратні. В цій архітектурі використовується ядро розміром 3×3 . Розмір вхідних даних визначається з розміру зображення, тобто $48 \times 48 \times 1$. Використовується лише в першому згортковому шарі. Далі вхідний розмір визначається автоматично з вихідного виміру попереднього шару. У випадку згорткового шару розмірність обов'язково містить три параметри: кількість рядків пікселів у матриці зображення, кількість стовпців, кількість кольорових каналів (у випадку чорно-білого зображення це дорівнює 1, а не відсутність каналів). Далі йде відступ, завдяки відступу не втратиться під час операції згортки розмір зображення. Відступ може мати два значення: «valid» та «same». За замовчуванням встановлено значення "valid", що мінімізує зображення. Якщо потрібно, щоб зображення не зменшувалось після операції згортки, тоді встановіть параметр "same", який додає рядки нулів і стовпців навколо зображення, це дозволяє зберегти розмір зображення однаковим. Відступ можна заповнити одиницями та віддзеркалити.

Далі йде шар, який нормалізує вхідні дані – batch normalization. Коли ваги шару змінюються, розподіл його результатів також змінюється, тому наступному шару доведеться починати навчання спочатку. Це заважає навчанню. Ця проблема називається зміщенням змінних. Рішення полягає в тому, що ми нормалізуємо кожен вхід окремо, але не для всіх зображень у наборі даних, а для міні-пакетів. Далі потрібно ввести нові параметри: зсув та розтягнення. Ці параметри також тренуються за допомогою градієнтного спуску.

Функція активації – це функція, яка приймає результат суматора як вхід і виконує певне перетворення, щоб перетворити суму зважених входів на адекватний вихід, який можна інтерпретувати з точки зору вирішення проблеми. Оскільки у випадку невеликого значення функції підсумовування функція активації може повернути 0, тобто «нічого», то це свого роду аналог механізму в реальному нейроні, який відповідає за збудження нейрона (його "активація"). Relu повертає x ,

якщо x додатне, 0 – інакше, як показано на рисунку 2.13. Будь-яка функція може бути апроксимована комбінацією ReLU [56].

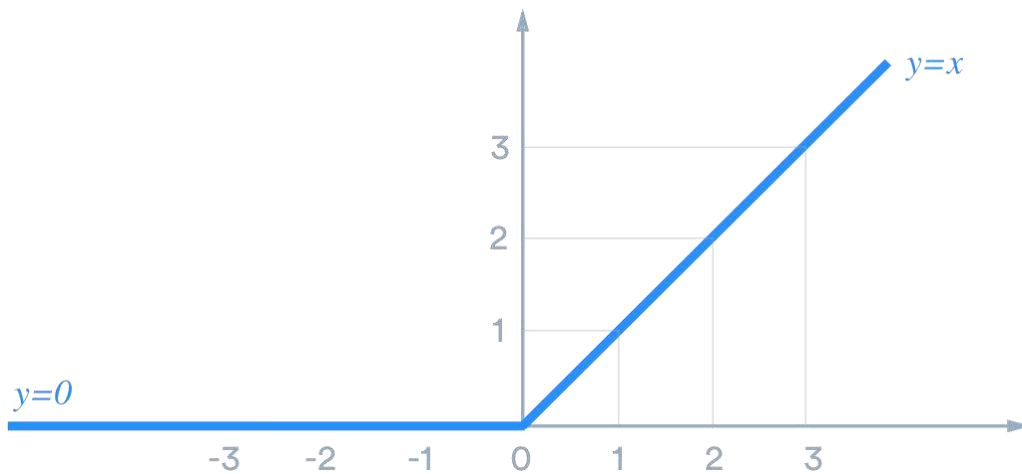


Рисунок 2.13 – Функція активації Relu

Relu (2.3) використовується як функція активації:

$$f(x) = \max(0, x) \quad (2.3)$$

Також вона має перевагу в тому, що ця функція створює розрідженість при активації нейронів. Тобто, на відміну від сигмовидної та гіперболічної функції, не всі нейрони будуть активовані, що зменшить обчислювальну складність. Ця властивість впливає з того, що всі негативні значення зникають. Використання ReLU значно покращує швидкість збіжності градієнтного спуску порівняно з сигмовидною та гіперболічною функцією. Наступний – шар макспулінгу. Шар максимальної підвибірки має розмір 2×2 , можливі й інші значення, найчастіше використовується це значення. Зазвичай згортковий шар, за яким слідує шар об'єднання, називається згортковим блоком. Наступний шар – шар згортки з 64 фільтрами, ядром згортки 3×3 . Функція активації також Relu. Шар макспулінгу 2×2 . Третій згортковий блок це шар згортки з 132 фільтрами, ядром згортки 3×3 . Шар макспулінгу теж 2×2 . Архітектура нейронної мережі візуалізується за допомогою інструменту Tensorboard.

На рисунку 2.14 показано розташування кожного шару на основі вищевказаної архітектури.

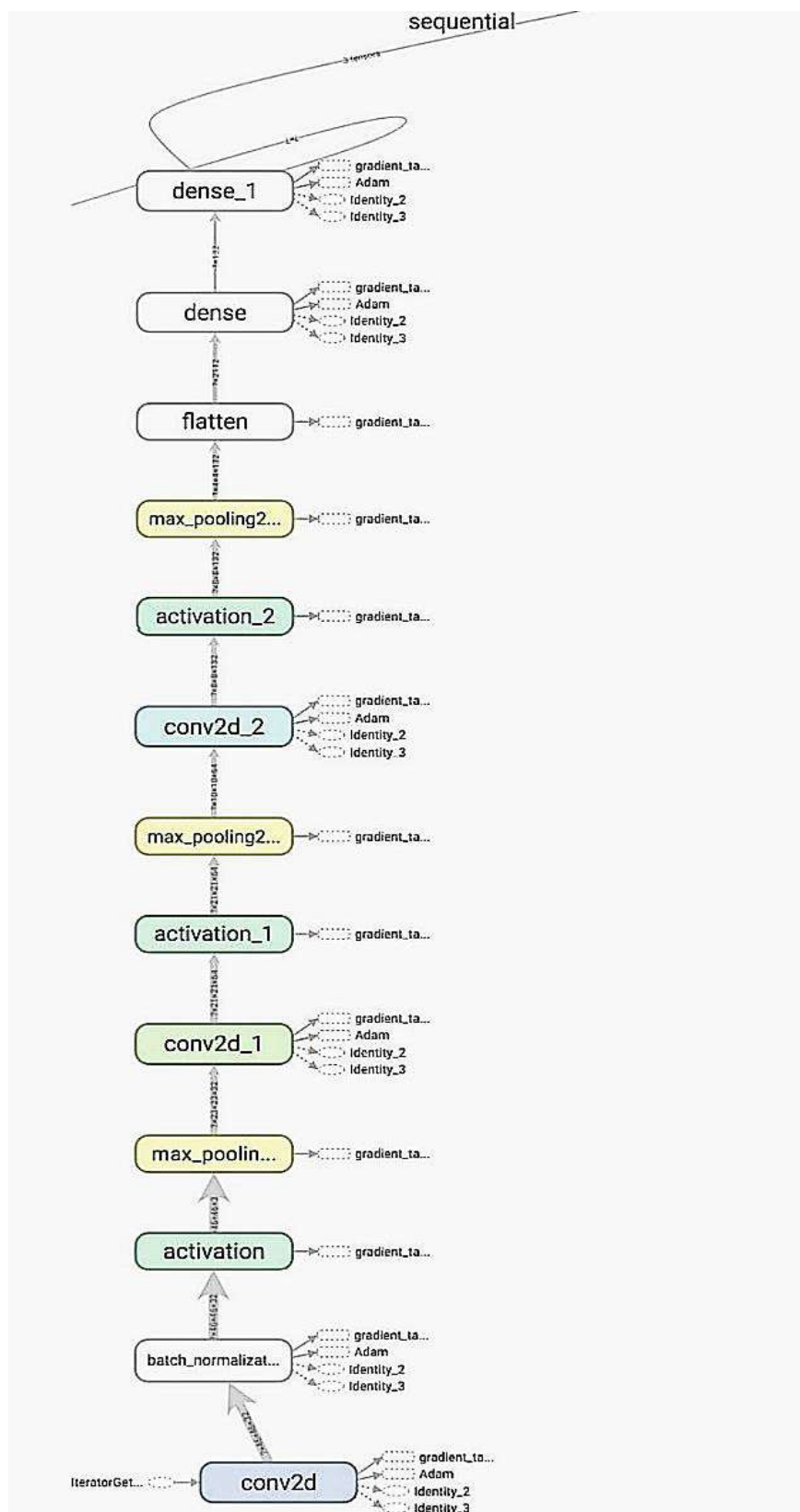


Рисунок 2.14 – Архітектура нейронної мережі

Після двох згорткових блоків йде шар Flatten. Він не має ніяких параметрів, а просто перетворює багатовимірні дані в плоский вектор, який вже можна дати на вхід повнозв'язному шару. Блоки згортки не дадуть нейронній мережі можливість віднесення зображення до певного класу. За це відповідає повнозв'язний шар. Всього в цій архітектурі 2 повнозв'язних шара. У першому повнозв'язному шарі 132 входи в нейрон. У другому 7 входів в нейрон, за кількістю класів, які потрібно передбачити.

Далі йде функція softmax (2.4), що визначає ймовірність приналежності до одного з семи класів.

Функція перетворює вектор z розмірності k в вектор σ тієї ж розмірності, де кожна координата σ_i отриманого вектора представлена дійсним числом в інтервалі $[0,1]$ і сума координат дорівнює 1 [57].

$$\sigma(z)_i = \frac{e^z}{\sum_{k=1}^k e^z} \quad (2.4)$$

2.3.4 Компіляція моделі

Після створення моделі її потрібно налаштувати з урахуванням функції втрат, метрик якості та оптимізаторів. В якості функції втрат була обрана категоріальна перехресна ентропія. Категоріальна кроссентропія (2.5) – це функція втрат, що зазвичай використовується для класифікації 3 або більше класів. Функція просто обчислює різницю для кількісної оцінки між двома розподілами ймовірностей. Категоріальна кроссентропія визначається наступним чином:

$$Loss = \sum_{i=1}^{output\ size} y_i \times \log \hat{y}_i \quad (2.5)$$

Де $\hat{y}_i$ є i -м скалярним значенням на виході моделі, y_i – відповідним цільовим значенням, а розмір виводу – кількістю скалярних значень на виході моделі. Ця втрата є дуже хорошим показником того, наскільки відрізняються два дискретні розподіли ймовірностей один від одного. У цьому контексті y_i – це ймовірність

виникнення події i , а сума всіх y_i дорівнює 1, що означає, що може статися рівно одна подія. Знак мінус гарантує зменшення втрат, коли розподіли наближаються один до одного [58].

В якості оптимізатора використовується Адам з learning rate $lr = 1e-3$. Це хороший алгоритм «за замовчуванням», гарна збіжність. Адам поєднує в собі переваги RMSProp, AdaGrad та SGD + Momentum. Спочатку ми оцінюємо перший імпульс і зважену суму градієнтів, а потім оцінюємо другий імпульс і квадрат градієнтів. Перший імпульс виконує роль швидкості, а другий служить для оптимізації параметрів. Спочатку обидва параметри дорівнюють нулю, тому до них додається зміщення корекції, щоб уникнути занадто великого кроку на самому початку.

Точність (2.6) використовується як показник якості. Точність є одним із показників для оцінки моделей класифікації. Неофіційно точність – це частка прогнозів, яку наша модель отримала правильно. Формально точність має таке визначення [59]:

$$accuracy = \frac{\text{number of correct predictions}}{\text{Total number of predictions}} \quad (2.6)$$

3 НАВЧАННЯ ТА ТЕСТУВАННЯ СИСТЕМИ

3.1 Процес навчання нейронної мережі

Навчальний набір даних використовується для навчання нейронної мережі, а для перевірки передбачень використовується тестовий набір даних. Розмір пакету = 64, кількість епох = 12. Модель працює неефективно, якщо на вхід подається одразу уся вибірка тренувальних зображень. Тому дані були розділені на 64 елементи у кожному пакеті. Щоб мережа могла навчитися, потрібно пройти багато ітерацій кожного пакету. Чим більший обсяг вибіркової партії, тим більше пам'яті

потрібно від комп'ютера. Нижче наведені графіки тренувань. Вони показують, що можна взяти 12 епох для навчання, цього цілком достатньо. Ще трохи, і модель перенавчається (рис. 3.1). Перенавчання – це явище, коли модель що навчається, добре розпізнає приклади з навчальної вибірки, але при цьому не розпізнає або погано розпізнає будь-які інші приклади, які не брали участі в процесі навчання (тобто тестовий набір).

Якщо відбувається перенавчання, то модель не набуває здатності до узагальнення – можливості поширювати виявлені на навчальній множині залежності і закономірності на нові дані. Така модель на практиці не буде корисною, навіть якщо її помилка навчання маленька.

Рисунок 3.1 показує точність передбачень на тестовому наборі даних та навчальному на кожну епоху.

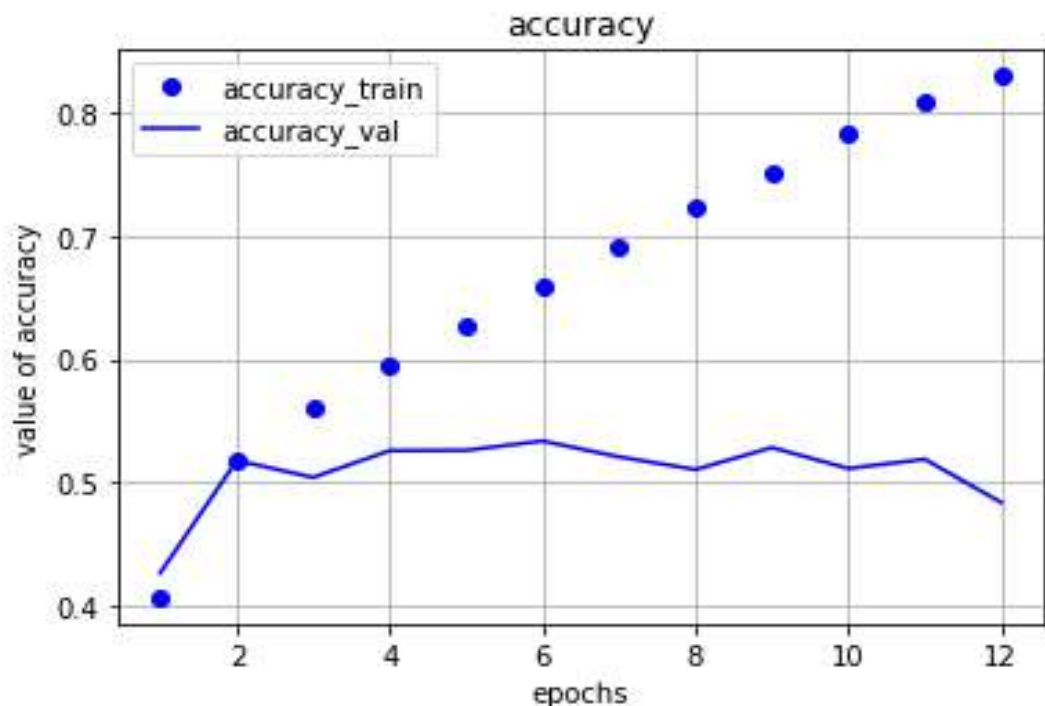


Рисунок 3.1 – Точність

На рисунку 3.2 показано зміну зростання функції збитків на тестових та навчальних даних.

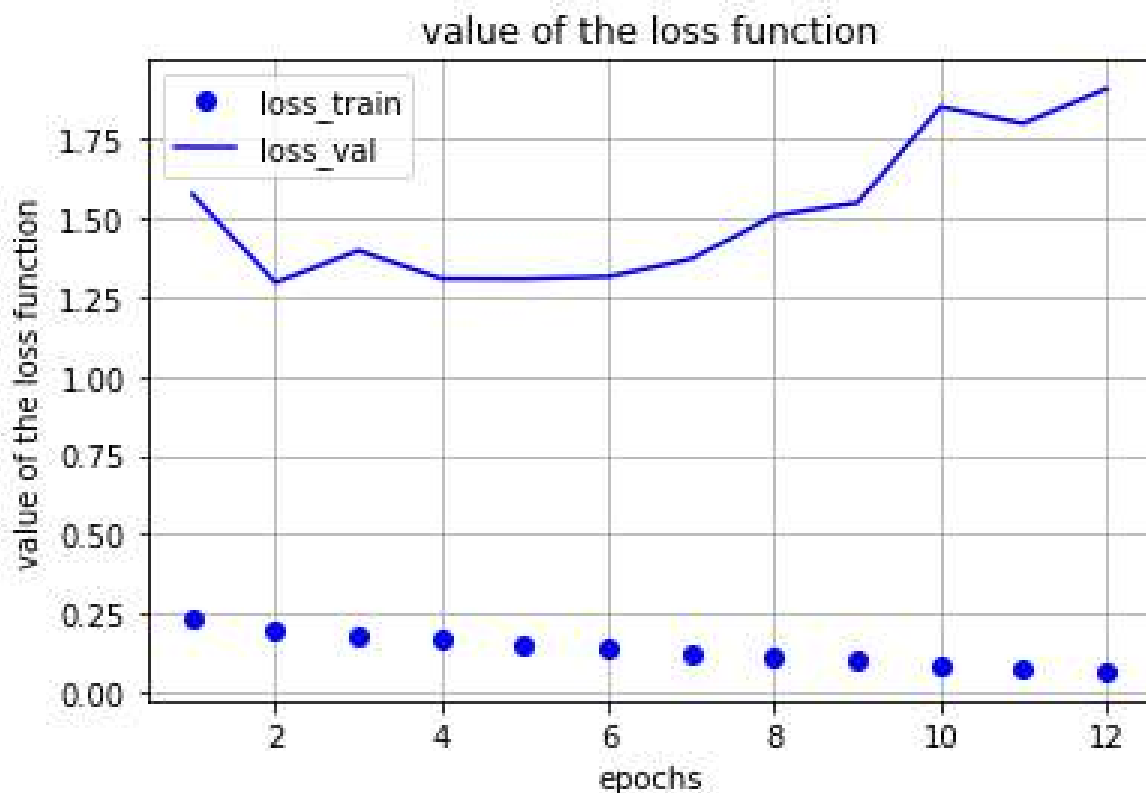


Рисунок 3.2 – Значення функцій втрат

3.2 Тестування працездатності нейронної мережі

Для того щоб зрозуміти на скільки нейронна мережа працездатна, потрібно її протестувати на тестових даних. Для цього потрібно взяти випадковий приклад з тестової вибірки і дати на вхід моделі. Цю операцію потрібно виконати для кожного класу емоцій. Так як для визначення ймовірності приналежності до класу, ми застосували функцію софтмакс, нейронна мережа буде передбачати в діапазоні від 0 до 1. 0 – низька ймовірність, 1 – висока ймовірність.

На рисунках 3.3 – 3.9 показано етап тестування. На рисунку 3.3 нейронна мережа розпізнала емоцію «щастя» з ймовірністю 1,0 на тестовій вибірці. Це означає, що мережа абсолютно точно вивчила емоцію і здатна достатньо точно її розпізнати. На рисунку 3.4 показано, що нейронна мережа розпізнала емоцію

«здивування» з ймовірністю 1,0. Це також є гарним показником якості розпізнавання. На рисунку 3.5 нейронна мережа розпізнала емоцію «гнів» з ймовірністю 0,78 і емоцію «страх» з ймовірністю 0,22. Це тому, що емоції суб'єктивні і можуть бути схожі одна на одну. На малюнку 3.6 показана аналогічна ситуація, модель розпізнала 2 емоції, «страх» з ймовірністю 0,78 і «смуток» з ймовірністю 0,21. На малюнку 3.7 «нейтральна» емоція розпізнана з ймовірністю 0,76, емоція «смуток» – з ймовірністю 0,22 та емоція «гнів» – з низькою ймовірністю 0,05. На малюнку 3.8 згорткова нейронна мережа розпізнала емоцію «смуток» з ймовірністю 0,78, емоцію «гнів» з ймовірністю 0,18 і низьку ймовірність інших емоцій. На малюнку 3.9 нейронна мережа розпізнала емоцію «гнів» з ймовірністю 0,82. Тільки на малюнку 3.9 передбачення виявилися невірними. Це обумовлено тим, що в навчальній вибірці не вистачає достатньої кількості прикладів данної емоції. Це можна вирішити за допомогою аугментації – збільшенням розміру навчальних даних. На інших малюнках нейронна мережа правильно передбачила всі емоції. Вище кожного прикладу прикріплено ярлик кожної емоції.

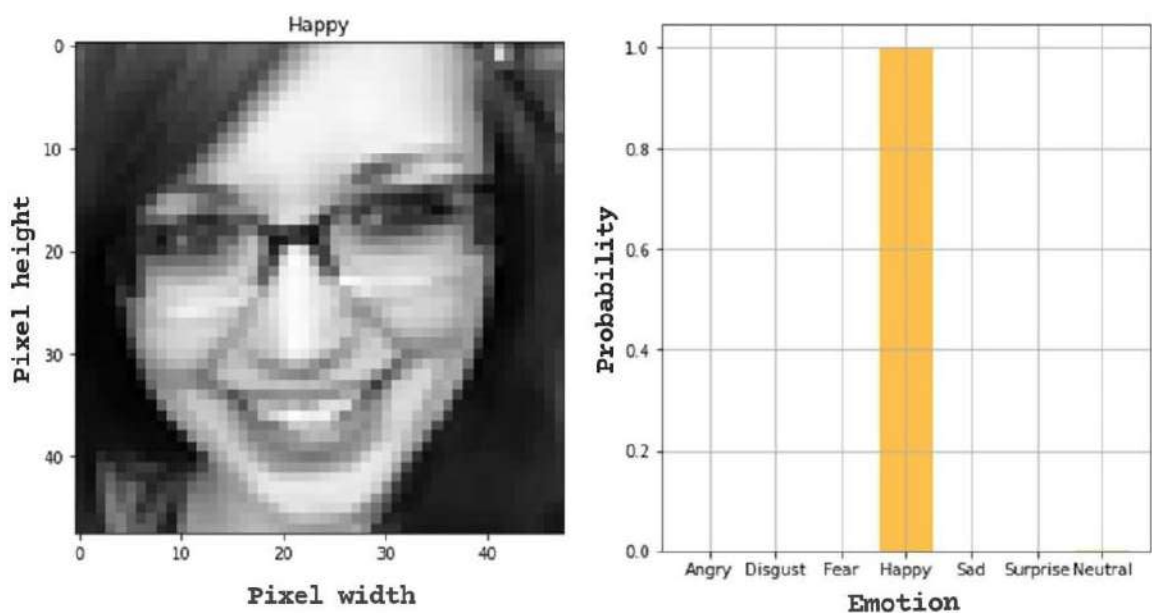


Рисунок 3.3 – Тестування емоції «щастя»

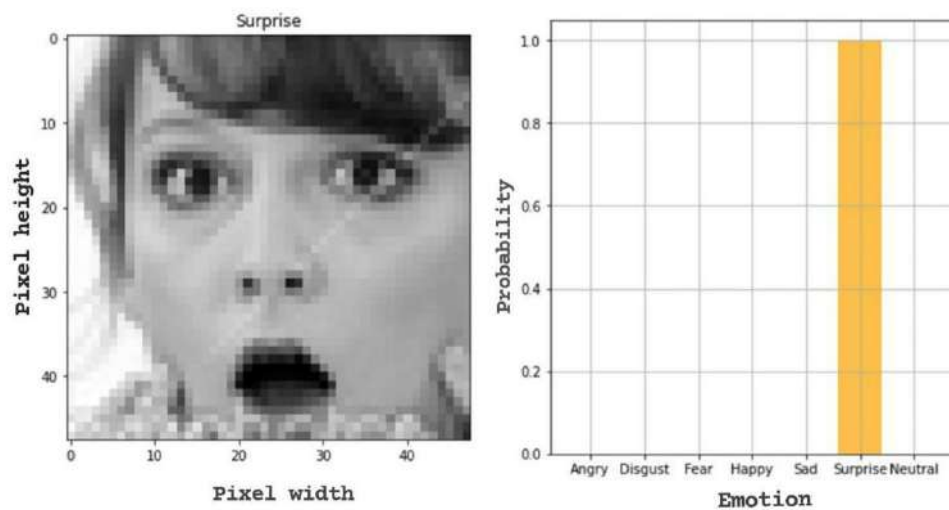


Рисунок 3.4 – Тестування емоції «здивування»

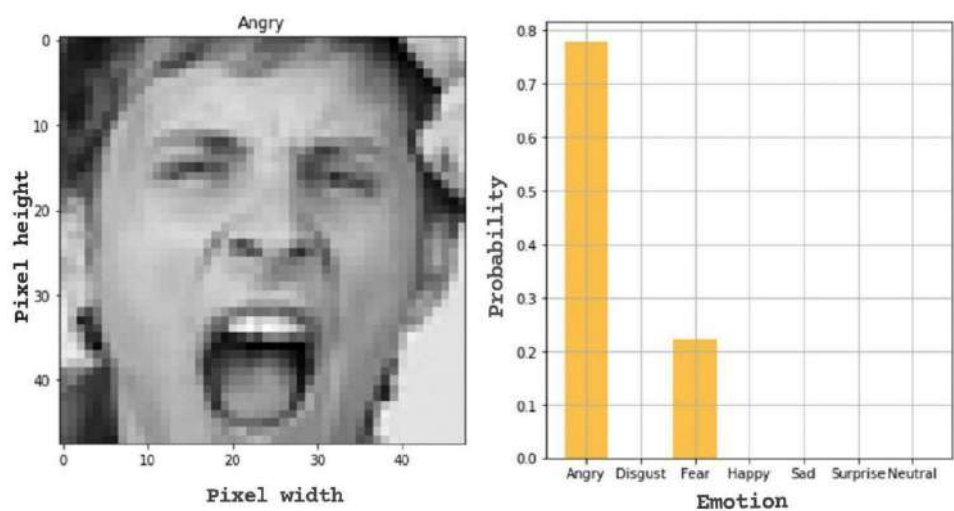


Рисунок 3.5 – Тестування емоції «гнів»

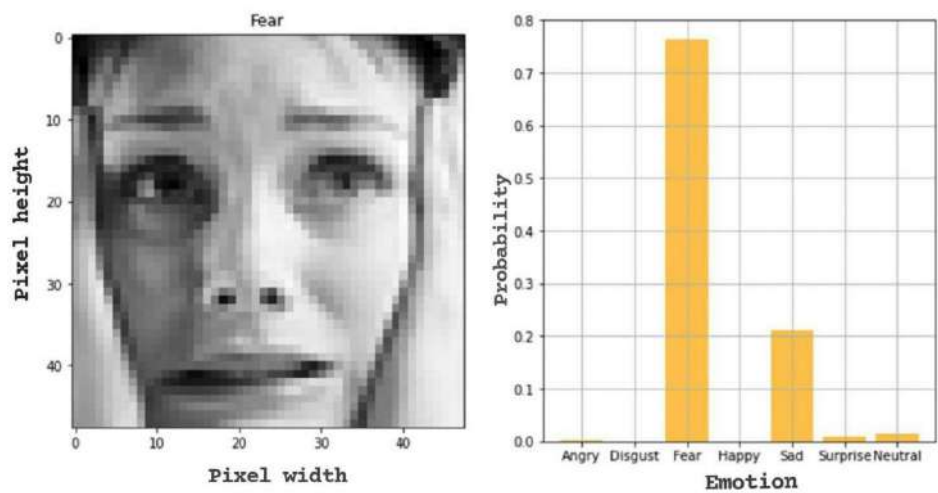


Рисунок 3.6 – тестування емоції «страх»

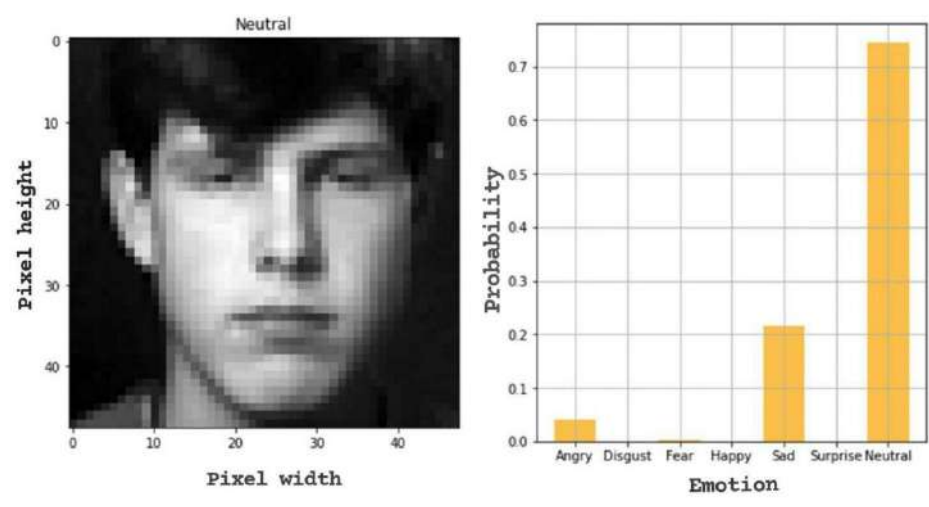


Рисунок 3.7 – Тестування емоції «нейтральність»

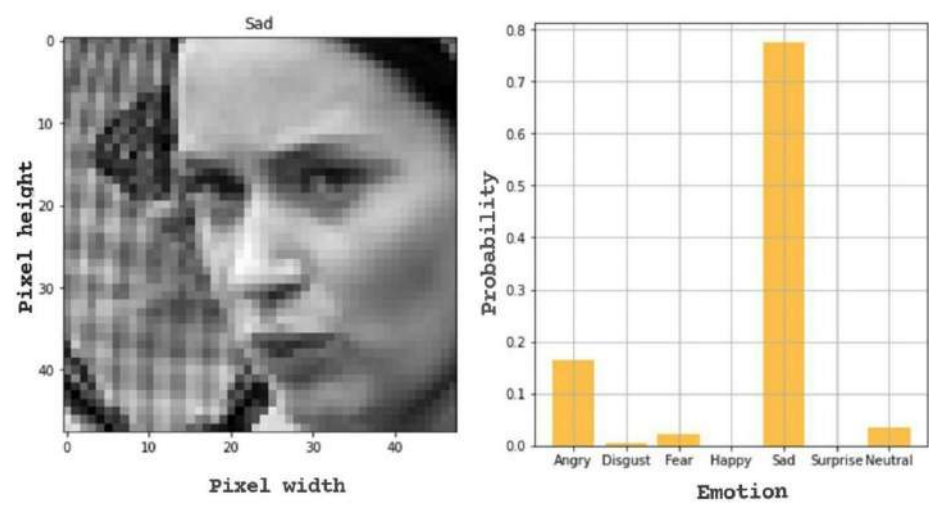


Рисунок 3.8 – Тестування емоції «смуток»

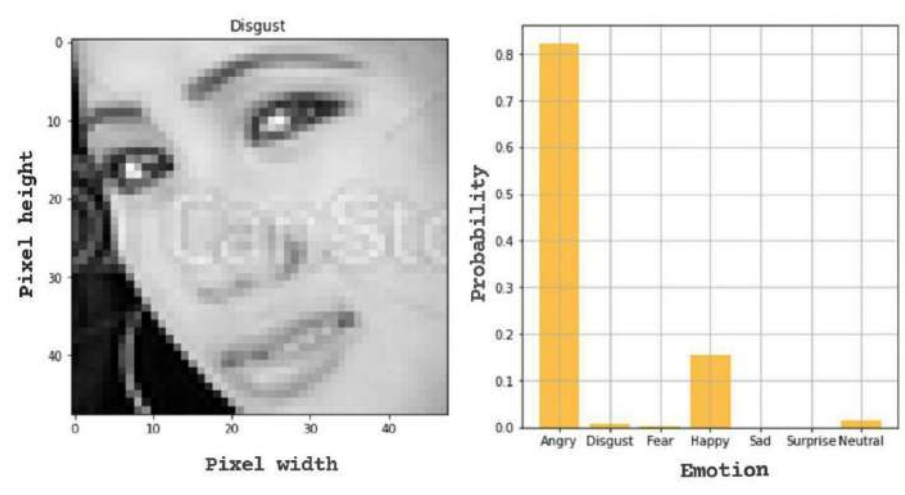


Рисунок 3.9 – Тесування емоції «огида»

3.3 Інтеграція моделі в систему

Після того як на етапі тестування модель виявилася працездатною, можна приступати до її практичного використання (Рис. 3.10). Ідея полягає в тому, що кожен модель нейронної мережі можна інтегрувати в будь-яку систему. Вона швидко виконується і займає мало простору на диску. Можна інтегрувати в соціальні мережі, камери, ігри, веб системи та ін. Спочатку за допомогою фреймворка TensorFlow збережемо модель в json файл та ваги моделі у h5 файл.

В данній роботі система взаємодіє через камеру вбудовану веб-камеру ноутбуку. Імпортується бібліотека numpy. За допомогою можливостей бібліотеки OpenCV, захоплюється відео з камери, пряма трансляція. Для того щоб почати трансляцію, потрібно створити об'єкт VideoCapture. Розпізнається область обличчя за допомогою каскадів Хаара. Далі область обличчя трансформується для моделі у чорно-білу область 48*48. Після цього знаходиться максимальний показник ймовірності емоції до виділеної області та виводиться на екран. Також в бібліотеці OpenCV є можливість малювати лінії. В режимі реального часу малюється квадрат який вказує на виділену область обличчя, а зверху написана емоція яку передбачила модель. Робота системи показана на рисунках 3.10 – 3.11.

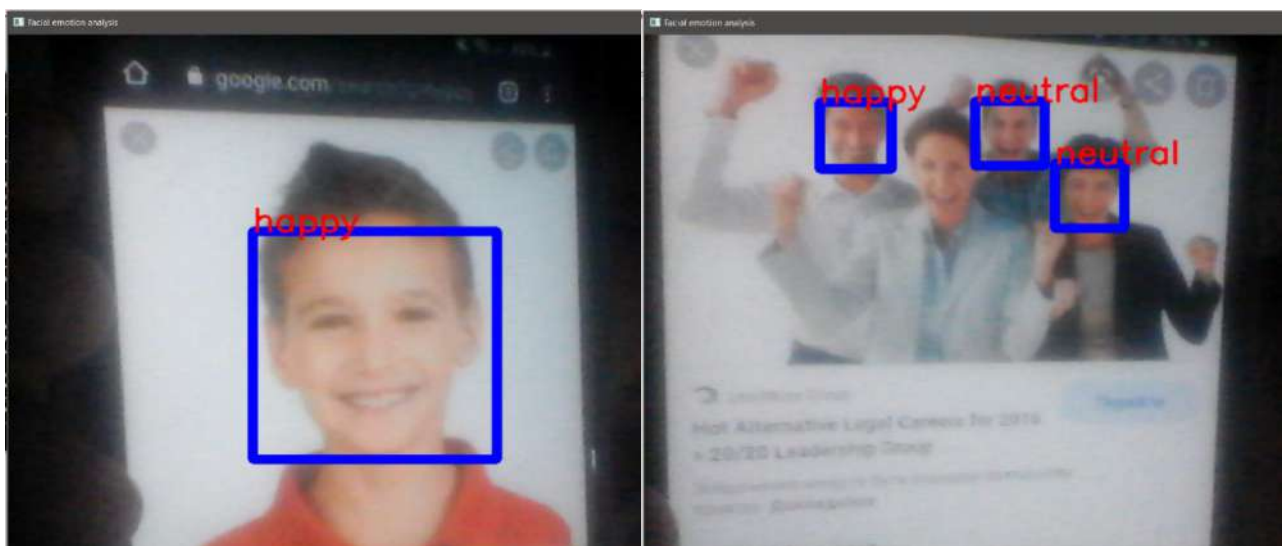


Рисунок 3.10 – Робота системи розпізнавання емоцій в режимі реального часу



Рисунок 3.11 – Робота системи розпізнавання емоцій в режимі реального часу

Система працює коректно. Отримано якісні результати. Таким чином, у дипломі реалізована система розпізнавання емоцій людини, основна ідея якої, полягає у тому, що для пошуку простору ознак виразів та їх класифікації використана глибинна нейронна мережа. Для її навчання було використано 20 000 зображень, які поділені на сім класів емоцій. Зображення були попередньо оброблені та нормалізовані для кращого розпізнавання. Модель нейронної мережі та ваги було збережено для подальшого інтегрування в складні системи.

ВИСНОВКИ

У дипломі реалізована система розпізнавання емоцій за допомогою глибинної нейронної мережі. Вирішення цієї технічно та математично складної проблеми вимагає поглибленого вивчення предметної області, власного аналітичного аналізу, вмінь та навичок у обробці та відборі експериментальних даних, глибоких знань у галузі дискретної математики, геометрії, програмування, статистики, а також психології та багато інших галузей науки [60].

У цій роботі реалізована згорткова нейронна мережа, яка здатна виявляти емоції 7 типів, навіть за умови поганої якості зображень, з предметами на обличчі (окуляри) та під невеликим кутом нахилу. А також згорткова нейронна мережа не потребує великої кількості математичних обчислень на відміну від звичайної нейронної мережі. Для цього було знайдено набір даних у вільному доступі, придатний для цього завдання, всі дані з набору даних були нормалізовані, модель побудована та навчена, обрано метрики з оптимізаторами та функція втрат. Модель також протестована на тестовому наборі даних, а гіперпараметри моделі обрано на основі валідаційних даних.

Перевага полягає в тому, що дана модель була навчена в хмарному сервісі Google, з використанням GPU Tesla K80, швидко тренується і не вимагає великих обчислювальних ресурсів. Навчання даної мережі виявилось оптимальним для наявних ресурсів машини, і в той же час нейронна мережа тренувалася в 10 разів швидше на відеокарті, ніж на центральному процесорному блоці.

Модель згорткової нейронної мережі також здатна самостійно ідентифікувати особливі ознаки та відкинути непотрібні. Крім того, згорткова нейронна мережа здатна адаптуватися до зміни розміру зображення, масштабування, обертання, на відміну від інших алгоритмів нейронних мереж. Модель можна зберегти та інтегрувати в будь-яку систему, наприклад у детектор брехні, соціальні мережі, відеокамери у громадських місцях.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Government Technology [Електронний ресурс]. – Режим доступу: <https://www.govtech.com/computing/Understanding-the-Four-Types-of-Artificial-Intelligence.html>.
2. Learning Hub, 4 Main Types of Artificial Intelligence [Електронний ресурс]. – Режим доступу: <https://learn.g2.com/types-of-artificial-intelligence>.
3. BMC Software, Types of Artificial Intelligence [Електронний ресурс]. – Режим доступу: <https://www.bmc.com/blogs/artificial-intelligence-types/>.
4. The Difference Engine, An Introduction to Artificial Intelligence : The Four Types of AI [Електронний ресурс]. – Режим доступу: <https://www.thedifferenceengine.tech/blog/2019/11/an-introduction-to-artificial-intelligence-the-four-types-of-ai>.
5. Frontiers in Psychology, Cognitive reflection vs. calculation in decision making [Електронний ресурс]. – Режим доступу: <https://www.frontiersin.org/articles/10.3389/fpsyg.2015.00532/full>.
6. Стюарт Р. Artificial Intelligence: A Modern Approach / Р. Стюарт, Н. Пітер., 2006.
7. David B. Bayesian Reasoning and Machine Learning / Barber David., 2012. – (Cambridge University Press).
8. Джером Г. Ф. The Elements of Statistical Learning / Г. Ф. Джером, Т. Роберт, Г. Тревор., 2001.–(Springer).
9. Machine Learning Mastery, Linear Regression [Електронний ресурс]. – Режим доступу: <https://machinelearningmastery.com/linear-regression-for-machine-learning/>.
10. Stat.yale, Linear regression [Електронний ресурс]. – Режим доступу: <http://www.stat.yale.edu/Courses/1997-98/101/linreg.htm>.
11. Machine Learning Mastery, 4 Types of Classification Tasks in Machine

Learning [Электронный ресурс]. – Режим доступа: <https://machinelearningmastery.com/types-of-classification-in-machine-learning/>.

12. Statistics solutions, Logistic Regression [Электронный ресурс]. – Режим доступа: <https://www.statisticssolutions.com/what-is-logistic-regression/>.

13. Machine Learning Mastery, Supervised and Unsupervised Machine Learning Algorithms [Электронный ресурс]. – Режим доступа: <https://machinelearningmastery.com/supervised-and-unsupervised-machine-learning-algorithms/>.

14. Builtin, Robotics Technology [Электронный ресурс]. – Режим доступа: <https://builtin.com/robotics>.

15. Industry Week, What is the Importance of Unmanned Vehicles to our Economy? [Электронный ресурс]. – Режим доступа: <https://www.industryweek.com/technology-and-iiot/emerging-technologies/article/21960764/what-is-the-importance-of-unmanned-vehicles-to-our-economy>.

16. Towards Data Science, Your Guide to Natural Language Processing (NLP) [Электронный ресурс]. – Режим доступа: <https://towardsdatascience.com/your-guide-to-natural-language-processing-nlp-48ea2511f6e1>.

17. Forbes, What is Machine Vision And How Is It Used In Business Today? [Электронный ресурс]. – Режим доступа: <https://www.forbes.com/sites/bernardmarr/2019/10/11/what-is-machine-vision-and-how-is-it-used-in-business-today/?sh=3f62a99d6939>.

18. HCL Tech, What are the advantages of Artificial Intelligence? [Электронный ресурс]. – Режим доступа: <https://www.hcltech.com/technology-qa/what-are-the-advantages-of-artificial-intelligence>.

19. Towards Data Science, Advantages and Disadvantages of Artificial Intelligence [Электронный ресурс]. – Режим доступа: <https://towardsdatascience.com/advantages-and-disadvantages-of-artificial-intelligence-182a5ef6588c>.

20. Educba, Advantages of Artificial Intelligence [Электронный ресурс]. – Режим доступа: <https://www.educba.com/advantages-of-artificial-intelligence/>.
21. Proschool, What are the disadvantages of AI? [Электронный ресурс]. – Режим доступа: <https://www.proschoolonline.com/blog/what-are-the-disadvantages-of-ai>.
22. Analytix Labs, 10 Advantages and Disadvantages of Artificial Intelligence [Электронный ресурс]. – Режим доступа: <https://www.analytixlabs.co.in/blog/advantages-disadvantages-of-artificial-intelligence/>.
23. Edureka, What are the Advantages and Disadvantages of Artificial Intelligence? [Электронный ресурс]. – Режим доступа: <https://www.edureka.co/blog/what-are-the-advantages-and-disadvantages-of-artificial-intelligence/>.
24. Business World Innovative Technologies, What is Artificial Intelligence (AI) in Business? [Электронный ресурс]. – Режим доступа: <https://www.businessworldit.com/ai/artificial-intelligence-in-business/>.
25. eLearning Industry, 5 Main Roles Of Artificial Intelligence In Education [Электронный ресурс]. – Режим доступа: <https://elearningindustry.com/5-main-roles-artificial-intelligence-in-education>.
26. Insider, Artificial Intelligence in Financial Services: Applications and benefits of AI in finance [Электронный ресурс]. – Режим доступа: <https://www.businessinsider.com/ai-in-finance>.
27. Emerj, AI in Law and Legal Practice – A Comprehensive View of 35 Current Applications [Электронный ресурс]. – Режим доступа: <https://emerj.com/ai-sector-overviews/ai-in-law-legal-practice-current-applications/>.
28. McKinsey&Company, AI in production: A game changer for manufacturers with heavy assets [Электронный ресурс]. – Режим доступа: <https://www.mckinsey.com/business-functions/mckinsey-analytics/our-insights/ai-in-production-a-game-changer-for-manufacturers-with-heavy-assets>.
29. Insider, The impact of artificial intelligence in the banking sector & how AI

is being used in 2021 [Электронный ресурс]. – Режим доступа: <https://www.businessinsider.com/ai-in-banking-report>.

30. Forbes, How AI Can Transform The Transportation Industry, AI in transportation: the basics clarified [Электронный ресурс]. – Режим доступа: <https://www.forbes.com/sites/cognitiveworld/2019/07/26/how-ai-can-transform-the-transportation-industry/?sh=7d61e12a4964>.

31. K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, arXiv preprint arXiv:1409.1556, 2014.

32. Y. Sun, X. Wang, X. Tang, Deep learning face representation by joint identification-verification, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), arxiv.org/abs/1406.4773, 2014, pp. 1891-1898.

33. Y. I. Tian, T. Kanade, J. F. Cohn, Recognizing action units for facial expression analysis, in: IEEE Transactions on pattern analysis and machine intelligence, volume 23(2), IEEE, 2001, pp. 97–115. doi:10.1109/34.908962.

34. Sainath, T., Mohamed, A.-R., Kingsbury, B. & Ramabhadran, B. Deep convolutional neural networks for LVCSR. In Proc. Acoustics, Speech and Signal Processing 8614–8618 (2013).

35. Collobert, R., et al. Natural language processing (almost) from scratch. J. Mach. Learn. Res. 12, 2493–2537 (2011).

36. LeCun, Y. et al. Handwritten digit recognition with a back-propagation network. In Proc. Advances in Neural Information Processing Systems 396–404 (1990).

37. LeCun, Y., Bottou, L., Bengio, Y. & Haffner, P. Gradient-based learning applied to document recognition. Proc. IEEE 86, 2278–2324 (1998).

38. Simard, D., Steinkraus, P. Y. & Platt, J. C. Best practices for convolutional neural networks. In Proc. Document Analysis and Recognition 958–963 (2003).

39. Garcia, C. & Delakis, M. Convolutional face finder: a neural architecture for fast and robust face detection. IEEE Trans. Pattern Anal. Machine Intell. 26, 1408–1423 (2004).

40. Osadchy, M., LeCun, Y. & Miller, M. Synergistic face detection and pose estimation with energy-based models. J. Mach. Learn. Res. 8, 1197–1215 (2007).

41. Garcia, C. & Delakis, M. Convolutional face finder: a neural architecture for fast and robust face detection. *IEEE Trans. Pattern Anal. Machine Intell.* 26, 1408–1423 (2004).
42. TensorFlow [Электронный ресурс]. – Режим доступа: <https://www.tensorflow.org/>.
43. NumPy [Электронный ресурс]. – Режим доступа: <https://numpy.org/>.
44. A K. Jain, P. Flynn, A A Ross, *Handbook of Biometrics*, New York: Springer, 2010.
45. P. N. Belhumeur, 1. P. Hespanha, D. 1. Kriegman, *Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection*. *European Conference on Computer Vision*, 1996.
46. L. Cuimei, Q. Zhiliang, J. Nan and W. Jianhua, "Human face detection algorithm via Haar cascade classifier combined with three additional classifiers", *Electronic Measurement & Instruments (ICEMI) 2017 13th IEEE International Conference on*, pp. 483-487, October 2017.
47. V. Sing, V. Shokeen and B. Singh, "Face detection by haar cascade classifier with simple and complex backgrounds images using opencv implementation", *International Journal of Advanced Technology in Engineering and Science*, vol. 11, no. 12, pp. 33-38, 2013.
48. Y. Li, X. Xu, N. Mu and L. Chen, "Eye-gaze tracking system by haar cascade classifier", *Industrial Electronics and Applications (ICIEA) 2016 IEEE 11th Conference on*, pp. 564-567, June 2016.
49. OpenCV, Python Tutorials [Электронный ресурс]. – Режим доступа: https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_tutorials.html.
50. Bowyer, K. W., Chang, K., & Flynn, P. (2006). A survey of approaches and challenges in 3D and Multi-modal 3D + 2D face recognition. *Computer Vision and Image Understanding*, 101(1), 1–15.
51. Lowe, D. (2004). Distinctive image features from scale-invariant key points. *International Journal of Computer Vision*, 60(2), 91–110.

52. Cs231n, Convolutional Neural Network [Электронный ресурс]. – Режим доступа: <https://cs231n.github.io/convolutional-networks/#overview>.
53. TowardsDataScience, A Comprehensive Guide to Convolutional Neural Networks [Электронный ресурс]. – Режим доступа: <https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53>.
54. O. Abdel-hamid, L. Deng and D. Yu, Exploring Convolutional Neural Network Structures and Optimization Techniques for Speech Recognition, pp. 3366-3370, August 2013.
55. Shan Li, Weihong Deng, JunPing Du; Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 2852-2861.
56. Paper with code, Rectified Linear Units [Электронный ресурс]. – Режим доступа: <https://paperswithcode.com/method/relu>.
57. Medium, Softmax [Электронный ресурс]. – Режим доступа: <https://medium.com/@congyuzhou/softmax-3408fb42d55a>.
58. Github, Understanding Categorical Cross-Entropy Loss, Binary Cross-Entropy Loss, Softmax Loss, Logistic Loss, Focal Loss and all those confusing names [Электронный ресурс]. – Режим доступа: https://gombbru.github.io/2018/05/23/cross_entropy_loss/.
59. Thoughtco, Accuracy and precision [Электронный ресурс]. – Режим доступа: <https://www.thoughtco.com/difference-between-accuracy-and-precision-609328>.
60. International Workshop on Computer Modeling and Intelligent Systems 2021, Emotion Recognition Under Special Condition [Электронный ресурс]. – Режим доступа: <https://www.youtube.com/watch?v=KrIdbdiYwps&t=6984s>.

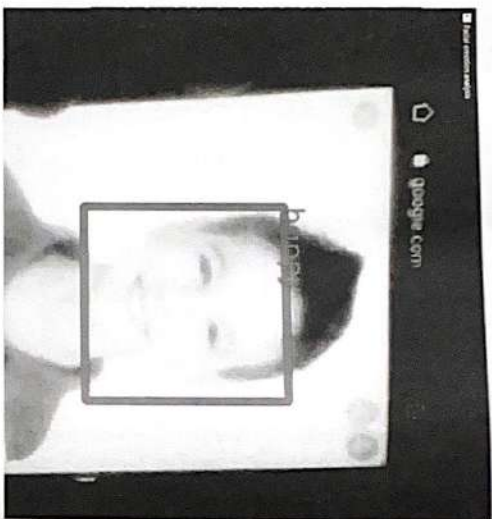


Рисунок 1 - Результат работы системы

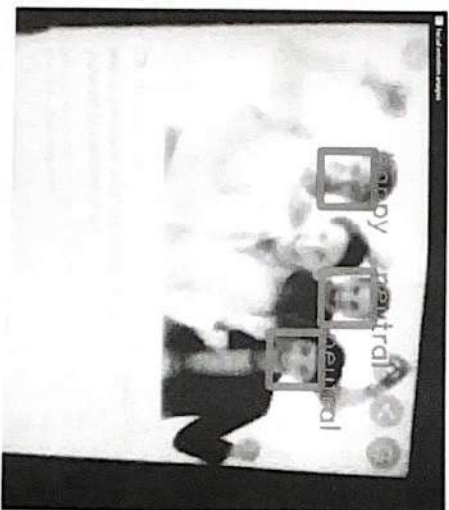


Рисунок 3 - Результат работы системы



Рисунок 2 - Результат работы системы

[illegible]



						13.02070849.00042 AI			
						Розробка системи розпізнавання емоцій на обличчі людини			
						Блок-схема алгоритма системи розпізнавання емоцій			
Ім.	Лист	№ докум.	Підпис	Дата		Тит.	Маса	Масшт.	
Розроб		Трохименко О.Е.	565/1	06.05.18					
Перевір		Ткаченко М.Ю.	02.05.18						
Тестув									
						Лист 1		Листів 1	
Інженер		Шербат Н.В.	06.05.18			НУ "Запорізька політехніка"			
Контролює		Куберикова І.В.	06.05.18			КНТ-527			

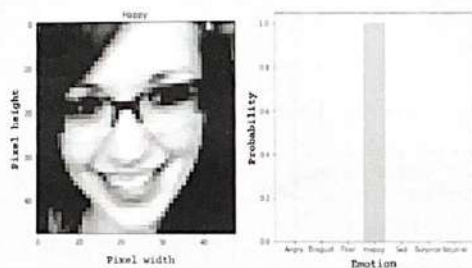


Рисунок 1 - Тестування емоції «щастя»

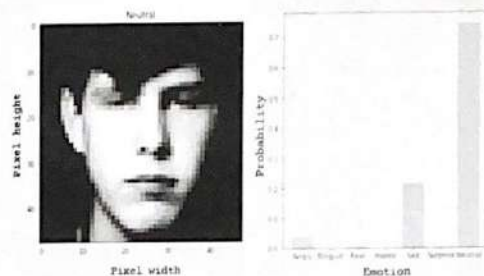


Рисунок 5 - Тестування емоції «нейтральність»

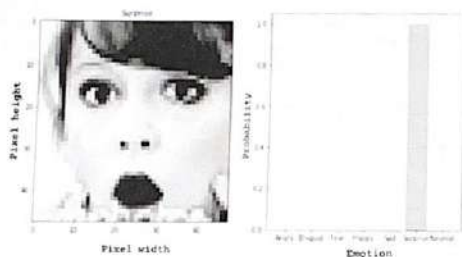


Рисунок 2 - Тестування емоції «здивування»

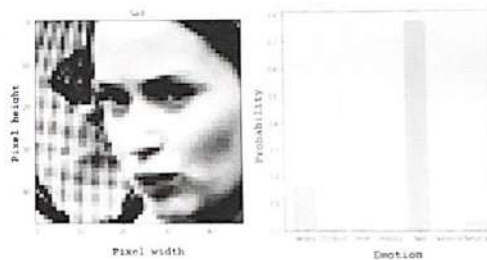


Рисунок 6 - Тестування емоції «смуток»

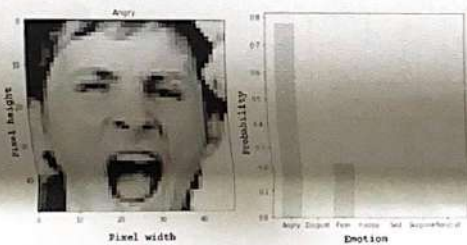


Рисунок 3 - Тестування емоції «гнів»

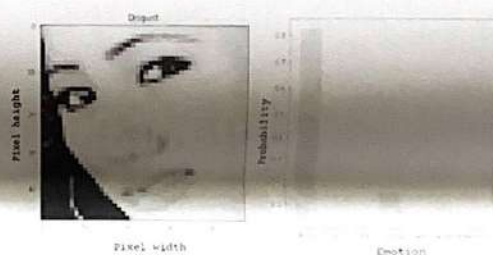


Рисунок 7 - Тестування емоції «огоида»

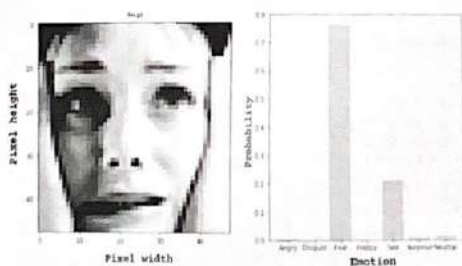
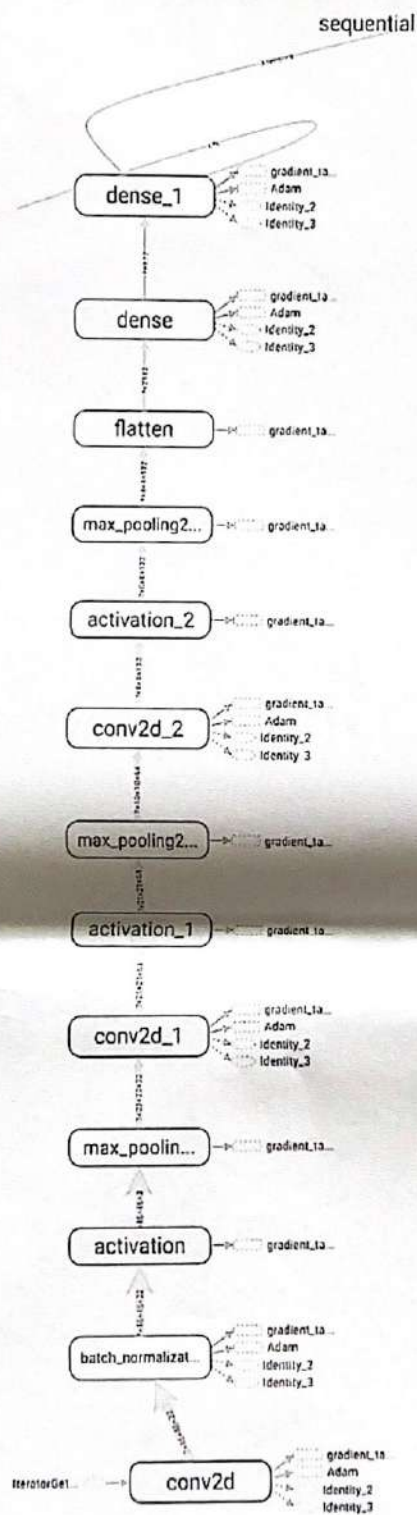


Рисунок 4 - Тестування емоції «страх»

					13.02070849.00042 ПЛЗ		
					Розробка системи розпізнавання емоцій на обличчі людини Тестування емоцій на моделі		
Лист	Лист	№ докум.	Підпис	Дата			
Розроб	Трохименко О.Е.	01.12.13	Л.С.П.				
Перевір	Трохименко М.Ю.						
Технік							
Начальн	Шербаєв Н.В.						
Листов	Кудерметов Р.Р.						
					Лист 1		
					Листів 1		
					НУ "Запорізька політехніка"		
					КНТ-527		



Ім'я	Піст.	№ докум.	Підпис	Дата
Розроб.	Протязь О.Е.	6304/16	2016.07.21	
Перевір.	Ткаченко М.Ю.			
Голов.				
П.с.п.	Шербас Н.В.			
Контроль	Кудриченко Р.Р.			

13.02070849.00042 ПЛ4

Розробка системи розпізнавання
емоцій на обличчі людини
Архітектура нейронної мережі

Лист	Маса	Масштаб
Лист 1	Лист 1	
НУ "Львівська політехніка"		
КНТ-527		