Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук
(повна назва)

Кафедра Штучного інтелекту
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти другий (магістерський)

Дослідження нейромережевих технологій для розпізнавання емоцій людини у реальному часі
(тема)

Виконав:
студент 2 курсу, групи СШІм-21-1
Кравець А. В.
(прізвище, ініціали)

Спеціальність 122 Комп'ютерні науки
(код і повна назва спеціальності)

Тип програми освітньо-наукова
(освітньо-професійна або освітньо-наукова)

Освітня програма Системи штучного інтелекту
(повна назва спеціалізації)

Керівник проф. Кулішова Н. Є.
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри _____ В.О. Філатов
(підпис) (прізвище, ініціали)

2023 р.

Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук
(повна назва)

Кафедра Штучного інтелекту
(повна назва)

Рівень вищої освіти другий (магістерський)

Спеціальність 122 Комп'ютерні науки
(код і повна назва)

Тип програми освітньо-наукова
(освітньо-професійна або освітньо-наукова)

Освітня програма Системи штучного інтелекту (СШІ)
(повна назва)

ЗАТВЕРДЖУЮ:
Зав. кафедри _____
(підпис)
«_____» _____ 20 ___ р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові Кравцю Артему Вікторовичу
(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження нейромережевих технологій для розпізнавання емоцій людини у реальному часі

затверджена наказом університету від 31 березня 20 23 р. № 306Ст

2. Термін подання студентом роботи до екзаменаційної комісії 23 травня 20 23 р.

3. Вихідні дані до роботи Операційна система – Windows 10, Частота процесору 4.9 ГГц, Середовище розробки – PyCharm, Мова програмування – Python, Бібліотеки і модулі, що використовувалися: OpenCV, NumPy, pandas, Keras, matplotlib.

4. Перелік питань, що потрібно опрацювати в роботі
1) Аналіз предметної області та постановка задачі
2) Аналіз існуючих підходів до розпізнавання емоцій людини
3) Теорія розпізнавання емоцій людини на відео за допомогою згорткових нейронних мереж
4) Програму для розпізнавання емоцій людини на відео у режимі реального часу

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (п.5 включається до завдання за рішенням випускової кафедри) Рисунок 1 – Розширений нелінійний синапс, Рисунок 2 – Розширений нейро-нечіткий нейрон, Рисунок 3 – Багатовимірний розширений нейро-нечіткий нейрон, Рисунок 4 – Основні операції CNN, Рисунок 5 – Приклад згортання зображення 5 × 5 із ядром 3 × 3, Рисунок 6 – Результати максимального та середнього об'єднання для зображення, Рисунок 7 – Виключення в NN, Рисунок 8 – Розташування функції softmax, Рисунок 9 – Приклад моделі на 68 ознакових точок, Рисунок 10 – 15 – Результати роботи програми.

6. Консультанти розділів роботи (п.6 включається до завдання за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Отримання завдання на кваліфікаційну роботу	03.04.2023	виконано
2	Аналіз предметної області і постановка задачі	04.04.2023-11.04.2023	виконано
3	Дослідження сфер застосування	12.04.2023-19.04.2023	виконано
4	Дослідження та аналіз основних нейромережевих	20.04.2023-27.04.2023	виконано
	технологій		
5	Аналіз теорії алгоритму CNN	28.04.2023-05.05.2023	виконано
6	Розробка програми	06.05.2023-11.05.2023	виконано
7	Проведення експериментів	12.05.2023-14.05.2023	виконано
8	Аналіз отриманих результатів	15.05.2023-17.05.2023	виконано
9	Написання пояснювальної записки	18.05.2023-21.05.2023	виконано
10	Попередній захист	22.05.2023	виконано
11	Захист перед ЕК	23.05.2023	

Дата видачі завдання 3 квітня 2023 р.

Студент _____
(підпис)

Керівник роботи _____ проф. Кулішова Н. Є.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ

Записка пояснювальна: 51 с., 15 рис., 2 дод., 27 джерел.

НЕЙРОМЕРЕЖЕВІ ТЕХНОЛОГІЇ, ГЛИБИННЕ НАВЧАННЯ, ЗГОРТКОВА НЕЙРОННА МЕРЕЖА, НЕЧІТКА СИСТЕМА, РОЗПІЗНАВАННЯ ОБРАЗІВ, РОЗПІЗНАВАННЯ ЕМОЦІЙ.

Об'єкт дослідження – архітектура та навчання нейромережевих технологій для розпізнавання емоцій людини у реальному часі.

Предмет дослідження – оцінка ефективності нейромережевих технологій для розпізнавання емоцій людини у реальному часі на основі аналізу відеоданих..

Мета роботи – дослідження рівня ефективності нейромережевих технологій для розпізнавання емоцій людини у реальному часі на основі аналізу відеоданих, а також розроблення та валідація методу розпізнавання емоцій на основі нейромережевих технологій.

Методи дослідження – огляд літератури з проблематики застосування нейромережевих технологій для розпізнавання емоцій людини у реальному часі на основі аналізу відеоданих. Під час виконання атестаційної роботи проведено порівняння популярних нейромережевих підходів для обробки двовимірних даних на основі аналізу відеоданих, оцінка їх ефективності та порівняння зі стандартними методами. Було проведено тестування нейромережевих підходів та виділені їх основні переваги та недоліки. На основі результатів був проведений аналіз швидкодії та обчислені метрики для оцінки якості класифікації емоцій людини.

ABSTRACT

Explanatory note: 51 p., 15 fig., 2 ann., 27 sources.

NEURAL NETWORK TECHNOLOGIES, DEEP LEARNING, CONVOLUTIONAL NEURAL NETWORK, FUZZY SYSTEM, PATTERN RECOGNITION, EMOTION RECOGNITION.

The object of the research is the architecture and training of neural network technologies for recognizing human emotions in real time.

The subject of the research is the evaluation of the effectiveness of neural network technologies for recognizing human emotions in real time based on the analysis of video data.

The purpose of the work is to investigate the level of effectiveness of neural network technologies for recognizing human emotions in real time based on the analysis of video data, as well as to develop and validate a method for recognizing emotions based on neural network technologies.

Research methods – a review of the literature on the application of neural network technologies to recognize human emotions in real time based on video data analysis. During the certification work, a comparison of popular neural network approaches for two-dimensional data processing based on video data analysis, evaluation of their effectiveness and comparison with standard methods was carried out. Neural network approaches were tested and their main advantages and disadvantages were reviewed. On the basis of the results, speed analysis and metric calculation were performed to assess the quality of human emotion classification.

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ

ШНМ – штучна нейронна мережа;

CNN – convolutional neural network – згорткова нейронна мережа;

DNN – deep neural network – глубинна нейронна мережа;

FNN – fuzzy neural network – нечітка нейронна мережа;

LSTM – long-short term memory – довга короткочасна пам’ять;

MENFN – multidimensional extended neo-fuzzy neuron – багатовимірний розширений нейро-нечіткий нейрон;

ML – machine learning – машинне навчання;

NFN – neo-fuzzy neuron – нейро-нечіткий нейрон;

RNN – recurrent neural network – рекурентна нейронна мережа;

SGD – stochastic gradient descent – стохастичний градієнтний спуск.

ВСТУП

Інформаційні технології стають невід'ємною частиною життя багатьох людей; вони активно впроваджуються в освіту, бізнес, охорону здоров'я, розваги. Багато з цих технологій є інтерактивними і реалізують постійну двосторонню взаємодію людини з комп'ютером або мобільним пристроєм. Одним із перспективних напрямків розвитку інтерфейсів для такої взаємності є підхід, який використовує розпізнавання людей, їх віку, статі, стану здоров’я, емоційного стану на відео в реальному часі. Ця складна технічна проблема вже знаходить свої рішення. Як математична задача, завдання розпізнавання емоційного стану користувача по відео зводиться до виявлення характерних ознак і кластеризації зібраних даних.

Розпізнавання та інтерпретація емоцій стали важливими завданнями в різних сферах людської діяльності, зокрема в психології, медицині, маркетингу, бізнесі, електронній комерції та соціальних медіа. Цей прорив пов’язаний зі значним збільшенням потоку даних, а також зі збільшенням обчислювальних потужностей, що дозволяє обробляти та аналізувати цю велику кількість даних.

Потік даних – це неперервна, упорядкована (неявно за часом прибуття чи явно за часовою міткою) послідовність елементів, що надходять в режимі реального часу. Створення таких потоків є серйозним, трудомістким завданням, що істотно збільшує вартість розробки проектів і тривалість реалізації. В даному випадку неможливо контролювати порядок, за яким елементи надходять на обробку, а також неможливо зберегти усю послідовність повністю. Більш того, ці елементи надходять з високою швидкістю, що призводить до величезних або навіть нескінчених об’ємів даних. Серед особливостей, що є спільними для потоків даних, можна виділити наступні:

- елементи надходять неперервно та послідовно;
- зазвичай потоки генеруються зовнішніми ресурсами, через це

системи, що використовуються для обробки потоків зазвичай не мають безпосереднього доступу до джерела даних і не можуть його контролювати;

− вихідні характеристики потоків даних не піддаються контролю і зазвичай непередбачувані;

− чутливість до змін у розподілі даних через динамічний характер реального середовища

− елементи потоків даних підвержені помилкам.

Зважаючи на ці особливості потоків даних, традиційні алгоритми машинного навчання непридатні для обробки таких даних через обмеження ресурсів з боку пам’яті та часу роботи. Для подолання цих обмежень були та продовжують розроблятися нові алгоритми, здатні обробляти неперервні потоки даних. Відсутність структури у даних робить цю задачу ще більш складною. Останніми роками нейромережеві підходи зарекомендували себе як потужний інструмент для обробки та аналізу неструктурованих даних:

− обробка природомовного тексту застосовується для вилучення значення з документів, електронних листів, статей, постів в соціальних мережах;

− алгоритми розпізнавання образів застосовується для знаходження різноманітних об’єктів у каталогах цифрових зображень;

− перетворення мови в текст можна використовувати для перетворення аудіо в текст.

Розробка архітектур нейронних мереж та алгоритмів їх навчання, що здатні швидко опрацьовувати потокові неструктуровані дані є логічним кроком у напрямку збільшення частки даних, що можуть бути аналізовані та використані організаціями для оптимізації тих чи інших процесів.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАДАЧІ

Стандартним підходом у машинному навчанні [1], [2], [3], особливо при використанні глибинних нейронних мереж, стало так зване пакетне навчання. При використанні даного підходу передбачається, що модель будується в автономному режимі відразу для всього набору даних [4], [5], [6] і ніколи не оновлюється в майбутньому. Крім того, глибинні нейронні мережі зазвичай налаштовуються у режимі багатоепохового навчання, що потребує багато часу.

1.1 Аналіз структури нейромережі

Штучна нейронна мережа (ШНМ) – математична модель, а також її програмне або апаратне втілення, побудована за принципом організації та функціонування біологічних нейронних мереж – мереж нервових клітин живого організма.

ШНМ складається з штучних нейронів (artificial neuron), кожен з яких представляє собою спрощену модель біологічного нейрона. Все, що робить штучний нейрон – це приймає сигнали з багатьох входів, обробляє їх єдиним чином і передає результат на багато інших штучні нейрони, тобто робить те ж саме, що і нейрон біологічний. Біологічні нейрони пов'язані між собою аксонами, місця стиків називаються синапсами. У синапсах відбувається посилення або ослаблення електрохімічного сигналу. Зв'язки між штучними нейронами називаються синаптичними, або просто синапсами. У синапсу є один параметр – ваговий коефіцієнт, залежно від його значення відбувається ту чи іншу зміну інформації, коли вона передається від одного нейрона до іншого. Саме завдяки цьому вхідна інформація обробляється і перетворюється в результат, а навчання нейронної мережі засноване на експериментальному підборі такого вагового коефіцієнта для кожного синапсу, який і призводить до отримання необхідного результату.

Нейрони вхідного шару отримують дані ззовні (наприклад, від сенсорів системи розпізнавання осіб) і після їх обробки передають сигнали через синапси нейронів наступного шару. Нейрони другого шару (його називають прихованим, тому що він безпосередньо не пов'язаний ні з входом, ні з виходом ШНМ) обробляють отримані сигнали і передають їх нейронам вихідного шару. Оскільки мова йде про імітацію нейронів, то кожен процесор вхідного рівня пов'язаний з декількома процесорами прихованого рівня, кожен з яких, в свою чергу, пов'язаний з декількома процесорами рівня вихідного. Така, найпростіша ШНМ здатна до навчання і може знаходити прості взаємозв'язку в даних. ШНМ, здатна знаходити не тільки прості взаємозв'язку, а й взаємозв'язку між взаємозв'язками має набагато складнішу структуру. У ній може бути кілька прихованих шарів нейронів, що перемежовуються шарами, які виконують складні логічні перетворення. Кожен наступний шар мережі шукає взаємозв'язку в попередньому. Такі ШНМ здатні до глибокого (глибинного) навчання. Саме завдяки переходу на використання нейромережі з глибоким навчанням компанії Google вдалося різко підвищити якість роботи свого популярного продукту «Перекладач».

1.2 Активаційні функції нейромереж

В теорії нейронних мереж активаційною називається функція, аргументом якої є виважена сума входів штучного нейрона, а значенням – вихід нейрона:

$$y = F(S), \qquad S = \sum_{i=1}^{N} w_i * x_i \tag{1.1}$$

де S – зважена сума входів нейрона;

N – число входів нейрона;

wi – вага i-го входу нейрона;

xi – значення, яке надходить по i-му входу;

f(S) – активаційна функція;

y – вихідне значення нейрона (i, відповідно, активационной функції).

Від виду і форми використовуваної активаційний функції залежить вибір алгоритму навчання мережі, а також якість її навчання на конкретному навчальній множині. Параметри активационной функції підбираються експериментально в процесі навчання.

1.3 Активаційна функція

Першою активаційною функцією, використовуваної в моделі нейрона, запропонованої У. Маккаллохом і У. Питтсом, була функція одиничного стрибка, або функція Хевісайда. Вона задається формулою:

$$f(x) = \begin{cases} 0, & S < 0 \\ 1, & S \geq 0 \end{cases} \quad (1.2)$$

Таким чином, поки зважена сума S не перевищить деякий поріг θ.

Нейрон перебуває в «загальмованому» стані, і на його виході завжди буде 0. Поріг θ називають порогом активації або збудження, оскільки, як тільки сума його перевищить, нейрон переходить в «порушену» стан і формує на виході 1. У цьому випадку нейрон називається бінарним.

Недолік цієї функції очевидний – вона робить область значень виходу нейрона обмеженою і, по суті, зводить всі можливості такої нейронної мережі до вирішення завдання бінарної класифікації. Щоб апроксимувати більш складні залежності, потрібно збільшувати число бінарних нейронів.

Цю проблему частково вирішує застосування функції з лінійним порогом (1.3):

$$f(x) = \begin{cases} 0, \ S < 0 \\ aS, \ 0 \geq S \geq \theta \\ 1, \ S \geq \theta \end{cases} \quad (1.3)$$

де а – параметр крутизни.

Використання активаційної функції у вигляді лінійного порогу розширює область значень виходу нейрона, але при цьому він все ще залишається лінійним перетворювачем, що значно знижує аппроксимирующие можливості мережі. Крім цього, наявність двох точок розриву, де функції не диференційована, унеможливлює використання лінійного порогу в градієнтних алгоритмах навчання, де використовується похідна активаційної функції.

Тому при навчанні багатошарових нейронних мереж найбільш часто використовуються Сигмоїдальні активаційні функції, названі так за їх характерну

S – образну форму. До таких засобів належать гіперболічний тангенс і логістична функція, що задаються відповідними формулами:

$$th(S) = \frac{e^{aS} - e^{-aS}}{e^{aS} + e^{-aS}}$$
$$f(S) = \frac{1}{1+e^{-aS}} \quad (1.4)$$

де а – параметр навчання.

Це монотонно зростаючі функції, що диференціюються на всій області визначення, що робить їх застосовними в алгоритмах навчання, що використовують похідні активационной функції. Зазвичай все нейрони мережі мають однакову активаційну функцію.

1.4 Постановка завдання

Розпізнавання емоцій є важливим завданням в багатьох сферах нашого життя, таких як медицина, психологія, маркетинг, реклама, розваги та безпека. Наприклад, в медицині відомо, що емоції можуть впливати на здоров'я пацієнта та лікування, тому розпізнавання емоцій може допомогти в діагностиці та лікуванні психічних захворювань.

У маркетингу та рекламі розпізнавання емоцій може бути корисним для аналізу реакції клієнтів на рекламні матеріали, щоб зрозуміти їх емоційну реакцію та підлаштувати матеріали для більш ефективного просування продукту або послуги.

Також, у безпеці розпізнавання емоцій може бути використано для виявлення підозрілих осіб на вулицях або в аеропортах, а також для попередження насильницьких інцидентів.

Отже, розпізнавання емоцій є важливою задачею, яка може мати позитивний вплив на багато сфер нашого життя.

На сьогоднішній день існує багато нейромережових технологій для розпізнавання емоцій людини у реальному часі. Одними з найпопулярніших нейромережевих технологій є згорткові нейромережі (CNN). Ці нейромережі здатні виконувати операції над зображеннями та виділяти характерні риси обличчя, що допомагає розпізнати емоції на зображенні. Вони зазвичай використовуються, оскільки зберігають просторову інформацію. Для розпізнавання емоцій на зображеннях використовуються різноманітні архітектури CNN, такі як VGG, ResNet, Inception тощо. Ці архітектури навчаються на великих наборах даних, таких як CK+, FER2013, AffectNet тощо, та зазвичай дають високу точність на тестових наборах.

Також існують рекурентні нейромережі (RNN). Вони здатні працювати з досить великими послідовностями даних. Найпопулярнішою архітектурою є довга короткочасна пам’ять (LSTM). Ця архітектура здатна

пам'ятати інформацію з попередніх станів та захищати від проблеми зниклих градієнтів, що дозволяє їй точніше розпізнавати емоції.

Крім цих підходів ще можна розглядати нечіткі нейронні мережі (FNN) [7], [8], які зазвичай використовуються для моделювання складних систем, де точні значення не відомі або важко вимірювати. MENFN (багатовимірний розширений нейро-нечіткий нейрон) – це приклад архітектури нечітких нейронних мереж, який використовується для моделювання нечітких систем заснованих на правилах.

MENFN може бути використаний для розпізнавання емоцій, але його ефективність може бути обмежена в порівнянні з CNN і глибинними нейронними мережами (DNN). Це може бути пов'язано з обробкою вхідних даних та здатністю до адаптації до нових даних.

1.5 Глибинне навчання

На відміну від методів машинного навчання, методи глибинного навчання, такі як DNN та CNN, мають декілька переваг у порівнянні з методами машинного навчання для задачі розпізнавання емоцій на відео у реальному часі [9], [10].

По-перше, глибинні нейронні мережі, особливо CNN, мають здатність розпізнавати складні візуальні ознаки у зображеннях. Вони можуть виконувати функції, які людина здатна виконувати при сприйнятті візуальної інформації, наприклад, розпізнавати гранулування та зміну форми об'єктів. Це дає їм перевагу у розпізнаванні емоцій на відео.

По-друге, глибинні нейронні мережі мають здатність до самонавчання, що дозволяє їм виявляти складні залежності між вхідними даними і вихідними результатами. Це означає, що глибинні нейронні мережі можуть навчатися на великій кількості даних і покращувати свою точність в розпізнаванні емоцій з часом.

Отже, глибинні нейронні мережі, зокрема DNN та CNN, є більш ефективними для розпізнавання емоцій на відео у реальному часі порівняно з методами машинного навчання. Але ми всеодно розглянемо обидва методи більш детально, щоб у цьому впевнитися. З методів штучного навчання ми розглянемо FNN, а з методів глибинного навчання CNN.

2 АНАЛІЗ ІСНУЮЧИХ ПІДХОДІВ ДО РОЗПІЗНАВАННЯ ЕМОЦІЙ ЛЮДИНИ

2.1 Нечіткі нейронні мережі

Нечіткі нейронні мережі (FNN) є потужним інструментом для моделювання та обробки нечіткої інформації. Вони базуються на нечіткій логіці, яка дозволяє працювати з неоднозначними та нечіткими вхідними даними. У FNN використовуються нечіткі наближення, такі як нечіткі множини, нечіткі правила та нечіткі відношення, для представлення та обробки інформації.

Головною перевагою FNN є їх здатність працювати з нечіткими концептами та нечіткими зв'язками, які часто присутні в реальних задачах. Вони можуть моделювати невизначеність та неоднозначність, що дозволяє їм ефективно працювати зі складними системами та приймати розумні рішення навіть при обмежених та нечітких вхідних даних.

2.1.1 Стандартний та розширений нейро-нечіткий нейрон

Нейро-нечіткий нейрон (NFN) був вперше запропонований на початку 1990-х років Учіно та Ямакавою [11], [12], [13] для спрощення моделювання складних нелінійних систем. NFN є обчислювально простим, має високу точність апроксимації та здатність мінімізувати обраний критерій навчання в реальному часі.

Останнім часом з'явилися публікації про застосування NFN результатів у різних задачах. У [14], [15], [16] запропоновано різні архітектури NFN і відповідні алгоритми навчання. Практичні завдання, пов’язані з дослідженням вібрації асинхронних двигунів, функціонування підшипників, оптимізації чисельності колоній бактерій та проблем класифікації успішно вирішувалися за допомогою NFN [17], [18], [19], [20].

Стандартний нейро-нечіткий нейрон побудований на так званих нелінійних синапсах – елементах, які реалізують нечіткий висновок Такагі-Сугено нульового порядку [21], [22].

Ця форма відповідає перетворенню, яке виконує синапс:

$$f_i(x_i) = \sum_{i=1}^{h} w_{li}\mu_{li}(x_i) \tag{2.1}$$

де w_{li} – це синаптичні ваги;

$\mu_{li}(x_i)$ – функція приналежності до синапсу, яка фазифікує вхідний компонент x_i;

l – кількість ваг;

l = 1, 2, …, h, i – кількість синапсів, i = 1, 2, …, n.

Можливості синапсів були покращені за допомогою так званого розширеного нелінійного синапсу (ENS), це показано на рисунку 2.1.

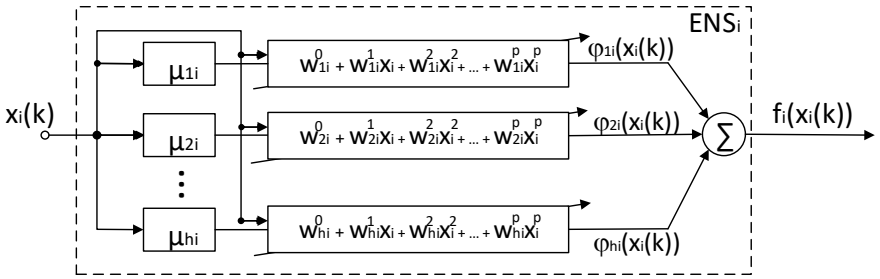

Рисунок 2.1 – Розширений нелінійний синапс

Розширений нелінійний синапс нейро-нечітких нейронів реалізує нечіткий висновок довільного порядку. Для цього використовуються додаткові змінні:

$$y_{li}(x_i) = \mu_{li}(x_i)\left(w_{li}^0 + w_{li}^1 x_i + w_{li}^2 x_i^2 + \ \ldots \ + \ w_{li}^p x_i^p\right), \tag{2.2}$$

$$f_i(x_i) = \sum_{i=1}^{h} w_{li}\mu_{li}(x_i)\,(w_{li}^0 + w_{li}^1 x_i + \\ + w_{li}^2 x_i^2 + \ldots + w_{li}^p x_i^p) = w_{1i}^0\mu_{1i}(x_i) + \\ + w_{1i}^1 x_i \mu_{1i}(x_i) + \ldots + w_{1i}^p x_i^p\,\mu_{1i}(x_i) + \\ + \ldots + w_{hi}^p x_i^p\,\mu_{hi}(x_i), \quad (2.3)$$

$$w_i = (w_{1i}^0, w_{1i}^1, \ldots, w_{2i}^0, \ldots, w_{2i}^p, \ldots, w_{hi}^p)^T, \quad (2.4)$$

далі ми можемо записати:

$$f_1(x_1) = w_i^T \tilde{u}_i(x_i), \quad (2.5)$$

$$y = \sum_{i=1}^{n} f_1(x_1) = \sum_{i=1}^{n} w_i^T\, \tilde{\mu}_\iota(x_i) = \tilde{w}^T \tilde{\mu}(x), \quad (2.6)$$

де $\tilde{w}^T = (w_1^T, \ldots, w_1^T, \ldots, w_n^T)^T$,

$$\tilde{\mu}(x) = (\tilde{\mu}_1^T(x_1), \ldots, \tilde{\mu}_i^T(x_i), \ldots, \tilde{\mu}_n^T(x_n))^T. \quad (2.7)$$

Таким чином, ENS реалізує вихід форми: якщо x_i – це x_{li}, то $f(x_i)$ – $w_{li}^0 + w_{li}^1 x_i + \ldots + w_{li}^p x_i^p$, l = 1, 2, … , h, що повторюється з формулюванням висновку r-го порядку Такагі-Сугено.

Синапси – нейро-чіткі нейронні структурні блоки, які реалізують відображення:

$$y = \sum_{i=1}^{n} f_i(x_i), \quad (2.8)$$

де x_i – елемент вектора вхідних даних $x = (x_1, \ldots, x_i, \ldots, x_n)^T \in R^n$;

i – номер компонента;

n – розмірність вектору;

y – скалярний вихід NFN.

У розширеному нелінійному синапсі на B-сплайни використовуються як функція належності.

Таким чином розширений нейро-нечіткий нейрон, що отримує вхідний вектор $x(k) = (x_1(k), \dots, x_i(k), \dots, x_n(k))^T$ (k = 1, 2, … – поточний відлік дискретного часу), генерує результуюче скалярне значення:

$$y(k) = \sum_{i=1}^{n} \sum_{l=1}^{h} w_{li}(k-1)\mu_{li}(x_i(k)) \qquad (2.9)$$

де $w_{li}(k-1)$ – це значення синаптичних ваг, отримані в результаті навчання на основі попередніх k-1 спостережень.

На рисунку 2.2 показано, як комбінуються елементи розширеного нейро-нечіткого нейрона [14].

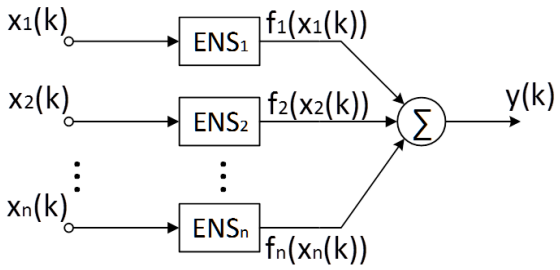

Рисунок 2.2 – Розширений нейро-нечіткий нейрон

2.1.2 Багатовимірний розширений нейро-нечіткий нейрон

Розглянута архітектура розширеного нео-нечіткого нейрона отримала подальший розвиток у роботі Бодяньского-Кулішової-Ху-Тищенко [23], де розглядався багатовимірний розширений нейро-нечіткий нейрон (MENFN). У MENFN вхідний сигнал вектора $x = (x_1, \dots, x_i, \dots, x_n)^T \in R^n$, генерує відповідь вихідного вектора. Ця структура містить кілька шарів. Вхідний рівень складається з розширених нейро-нечітких нейронів; проміжний

рівень з елементів, що відхиляють негативні значення, вихідний рівень нормалізує вихідні значення та об'єднує їх у результуючий вектор. Для вивчення розробленої архітектури використовувався алгоритм, заснований на градієнтній процедурі. Критерій навчання подається у вигляді:

$$E(k) = \frac{1}{2}(d(k))^2 = \frac{1}{2}e^2(k) =$$
$$= \frac{1}{2}(d(k) - \sum_{i=1}^{n}\sum_{l=1}^{n} w_{li}\,\mu_{li}(x_i(k)))^2, \qquad (2.10)$$

після чого алгоритм навчання дорівнює:

$$\begin{cases} w(k) = w(k-1) + r^{-1}(k)e(k)\mu\big(x(k)\big) \\ r(k) = ar(k-1) + \left\|\mu\big(x(k)\big)\right\|^2, 0 \le a \le 1. \end{cases} \qquad (2.11)$$

де $d(k)$ – зовнішні дані для навчання;

$e(k)$ – помилка навчання;

η – параметр темпу навчання.

Залежно від значення α, алгоритм Учіно-Ямакави [12] перетворюється на алгоритм Гудвіна-Раміджа-Кейнса [24] або однокроковий алгоритм Качмаржа-Відроу-Гоффа [25].

Незважаючи на свої універсальні властивості, даний алгоритм навчання не забезпечує виконання жорстких вимог до навчання системи в реальному часі на невеликій кількості вибірок навчальних даних, які присутні в постановці задачі розпізнавання емоційного стану користувача у відеоряді.

Для прискорення навчання MENFN було запропоновано використовувати ентропійно-інформаційний критерій навчання [26]:

$$E_j(t) = \frac{1}{2}\Big(1 + d_j(t)\Big)\,ln\frac{1 + d_j(t)}{1 + y_j(t)} +$$

$$+\frac{1}{2}\Big(1-d_j(t)\Big)\,ln\frac{1-d_j(t)}{1-y_j(t)}. \quad (2.12)$$

У Чихоцького-Анбехауна [26] було зазначено, що цей критерій стає істотно ефективним, якщо гіперболічний тангенс вибрано як функцію активації для $y_j(t)$:

$$y_j(x) = \tanh(\widetilde{w}^T x) \quad (2.13)$$

Тоді диференціювання [13] по w_{ij}, враховуючи [14], дає простий алгоритм навчання у вигляді:

$$\frac{dw_{ij}}{dt} = \eta e_j(t) x_i, \quad (2.14)$$

де $e_j(t) = d_j(t) - y_j(t)$ є локальною помилкою навчання.

У дискретному випадку цей простий вираз набуває вигляду:

$$w_j(k+1) = w_j(k) + \eta(k) e_j(k) x(k) \quad (2.15)$$

Ця простота запису забезпечує як обчислювальну простоту алгоритму, так і високу швидкість навчання, необхідну для онлайн програм.

У результаті архітектура багатовимірного розширеного нейро-нечіткого нейрону набула наступного вигляду, як на рисунку 2.3.

Перший шар складається з розширених нейро-нечітких нейронів ENFN, кількість яких відповідає розмірності вихідного вектора. Кількість нелінійних синапсів, що формує кожен із розширених неофазі нейронів, відповідає розмірності вектора вхідних ознак. На наступньому рівні реалізується функція:

$$V_j(k) = \Psi\left(y_j(k)\right) = \tanh\left(y_j(k)\right) \tag{2.16}$$

Останній рівень MENFN виявляє максимуми в обчислених значеннях алгоритмів навчання $v_j(k)$:

$$\tilde{y}(k) = \sup_{j=1}^{m}\left\{v_j(k)\right\}, \tag{2.17}$$

це необхідно, коли навчальний вектор заданий у діапазоні [0, 1].

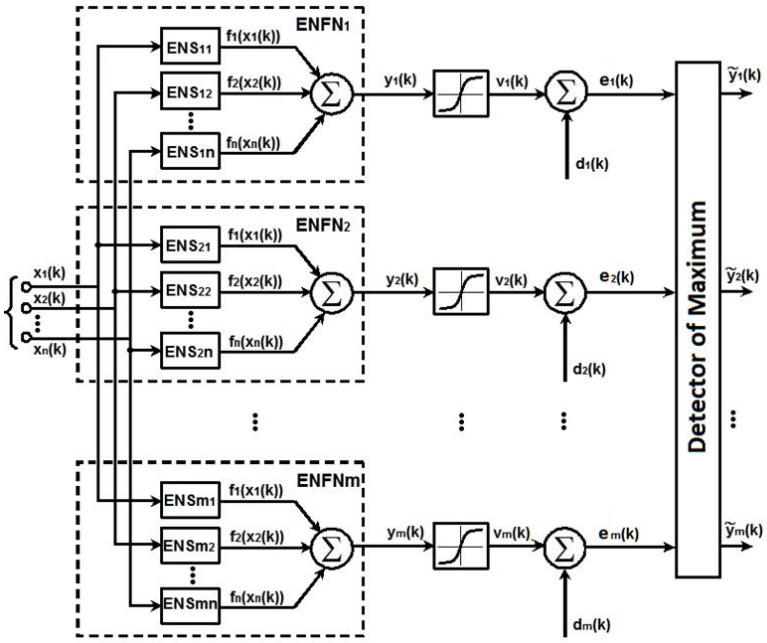

Рисунок 2.3 – Багатовимірний розширений нейро-нечіткий нейрон із функціями активації $y_j(x) = \tanh(\tilde{w}^T x)$

2.2 Рекурентні нейронні мережі

LSTM (Long Short-Term Memory) є популярною формою рекурентних нейронних мереж (RNN), яка використовується для моделювання послідовних даних та розпізнавання залежностей в часових рядах. LSTM

має здатність враховувати довгострокові залежності із зворотним поширенням помилки, що дозволяє цій моделі ефективно працювати зі вхідними даними, що мають довготривалі залежності між часовими кроками.

Одним із головних компонентів LSTM є комірка пам'яті (memory cell), яка зберігає та оновлює стан пам'яті на кожному кроці часу. Це дозволяє LSTM «зпам'ятовувати» інформацію, що була важлива на попередніх кроках часу та використовувати цю інформацію для прийняття рішень. LSTM також має спеціальні вентилі (gates), які контролюють потік інформації в комірку пам'яті та її вивід. Вентилі дозволяють LSTM вибирати, яку інформацію треба зберегти, забути або вивести на основі вхідних даних.

У контексті розпізнавання емоцій, LSTM може бути використана для аналізу послідовних даних, таких як послідовності аудіо- або текстових даних, що відображають емоційний вислів людини. LSTM може "запам'ятовувати" інформацію про попередні емоційні стани та використовувати цю інформацію для більш точного розпізнавання поточного емоційного стану. Також LSTM може працювати зі звуковими данними, що дає можливість аналізувати мовлення та розпізнавати емоційні компоненти в голосі.

Модель LSTM може вивчати часові характеристики відео з інформації послідовності. Мережа LSTM складається з блоку пам'яті, що містить структуру воріт. Формула його розрахунку така:

$$\begin{cases} i_t = \sigma(w_{xi}x_t + w_{hi}h_{t-1} + w_{ci}c_{t-1} + b_i) \\ f_t = \sigma\left(w_{xf}x_t + w_{hf}h_{t-1} + w_{cf}c_{t-1} + b_f\right) \\ c_t = f_t c_{t-1} + i_t tanh(w_{xc}x_t + w_{hc}h_{t-1} + w_{ci}h_{t-1} + b_c) \\ o_t = \sigma(w_{xo}x_t + w_{ho}h_{t-1} + w_{co}c_{t-1} + b_o) \\ h_t = o_t \tanh(c_t) \end{cases} \quad (2.18)$$

де σ – сигмовидна функція активації, i, f, o та c відповідно представляють вхідні вентилі, забувальний вентиль, вихідний вентиль та вектор стану комірки, W представляє матрицю вагів (наприклад, w_{hi} представляє матрицю вагів між прихованим шаром і вхідним вентилем), b представляє зсув (наприклад, b_i представляє вектор зсуву вхідного вентиля).

Модель LSTM може вирішити проблему згасання градієнта та вибуху градієнта в стандартній мережі RNN. Однак відео має багато зайвих відеокадрів. Ефективне виділення інформації про ключовий кадр відео допоможе підвищити швидкість розпізнавання завдання розпізнавання емоцій у відео. Щоб вирішити вищевказані проблеми, існує ідея механізму уваги, яка приймає модель LSTM, засновану на механізмі уваги в задачі розпізнавання емоцій по відео.

$H \in R^{d\times N}$ визначається матрицею вихідного вектора прихованого шару $[h_1, \dots, h_n]$ створеного моделлю LSTM, де d є розмірністю вихідного вектора прихованого шару, а N – кількість вихідних векторів прихованого шару. Механізм уваги створюю вектор ваги уваги α та зважене представлення прихованого шару r.

$$\begin{cases} M = \tanh(W_h) \\ \alpha = softmax(\omega^T\ M) \\ r = H\alpha^T \end{cases} \tag{2.19}$$

де $M \in R^{d\times N}$, $\alpha \in R^N$, $r \in R^d$, $W_h \in R^{d\times d}$, $\omega \in R^d$ – відповідні вагові матриці.

Формула розрахунку виходу власного вектора прихованого шару:

$$h^* = \tanh(W_p r + W_x h_N) \tag{2.20}$$

де $h^* \in R^d$, $W_p \in R^{d\times d}$, $W_x \in R^{d\times d}$ – відповідні вагові матриці.

h^* можна розглядати як вектор тимчасових ознак відео. Цей вектор використовується як вхідний сигнал шару softmax. Розподіл ймовірностей типів емоцій розраховується наступним чином:

$$y = softmax(W_s h^* + b_s) \tag{2.21}$$

де W_s та b_s – це ваги та зміщення шару softMax відповідно.

2.3 Згорткові нейронні мережі

Згорткові нейронні мережі (CNN) – це нейронні мережі, які відносяться до алгоритмів глибокого навчання. CNN зазвичай використовується для обробки зображень і володіє властивостями, що дозволяють виявляти та інтерпретувати властивості зображення з високою точністю. Це досягається за допомогою виконання операцій згортки (convolution) та підсумування (pooling) над зображенням, щоб отримати його репрезентацію у вигляді вектору ознак.

Активаційна функція в CNN зазвичай використовується для додавання нелінійності в мережу, щоб вона могла навчитися складним залежностям між вхідними та вихідними даними. Зазвичай використовуються такі функції, як ReLU (Rectified Linear Unit), Sigmoid, Tanh та інші.

Архітектура CNN складається з однієї або декількох шарів згортки та підсумування, за якими можуть слідувати шари повнозв’язних нейронів (Fully Connected Layers). Перші шари зазвичай виконують низькорівневу обробку зображення, таку як виявлення ліній та країв, після чого наступні шари агрегують ці ознаки для визначення більш складних форм, таких як форми об’єктів.

Одним з найпоширеніших підходів при навчанні CNN є зворотне поширення помилки (Backpropagation), яке використовується для оновлення

ваг шарів мережі. Також можуть використовуватися методи оптимізації, такі як Stochastic Gradient Descent (SGD), для пошуку оптимальних ваг для кращого розпізнавання зображень.

CNN є дуже потужним інструментом для розпізнавання образів. Вона використовується в багатьох областях, де необхідне розпізнавання образів, таких як:

- комп’ютерний зір: CNN використовується в комп’ютерному зорі для автоматичного розпізнавання образів, таких як лиця, транспортні засоби, будівлі тощо.
- медицина: В медицині CNN використовується для аналізу зображень, зокрема для діагностики різних захворювань та хвороб, таких як рак, діабет, артрит, серцеві захворювання тощо.
- реклама: CNN використовується в рекламі для аналізу зображень та відео та підбору рекламного контенту на основі визначення споживацьких потреб.
- автомобільна промисловість: В автомобільній промисловості CNN використовується для розпізнавання дорожніх знаків, пішоходів та інших об'єктів на дорозі, що сприяє безпеці руху транспортних засобів.
- відеоігри: Відеоігри використовують CNN для покращення графіки та створення більш реалістичних образів та персонажів.
- технічне зорове спостереження: CNN використовується для технічного зорового спостереження в промисловості, зокрема для контролю якості продукції та визначення дефектів на виробничій лінії.
- автономні системи: CNN використовується в автономних системах, таких як дрони та роботи, для розпізнавання об’єктів та навігації у середовищі.

У нашому випадку ми розглядаємо CNN для розпізнавання емоцій на відео. Та у цій області CNN мають багато переваг.

По-перше, CNN є ефективними в роботі з великими обсягами відеоданих, оскільки вони можуть автоматично виявляти та

використовувати локальні залежності та шаблони в зображеннях. Вони використовують згортки та пулінг для отримання репрезентативних функцій, що дозволяє впевнено виявляти обличчя та інші важливі емоційні ознаки.

По-друге, CNN може автоматично вивчати важливі ознаки зображень, такі як форма обличчя, вирази очей та рота, що є ключовими факторами при розпізнаванні емоцій. Вони можуть ефективно впоратись зі змінами в освітленні, масштабуванні та інших варіаціях зображень, що часто спостерігаються в реальних відеоданих.

По-третє, CNN можуть використовувати просторову та часову інформацію для аналізу емоцій на відео. Завдяки використанню 3D-згорток та LSTM-шарів, CNN можуть моделювати динаміку емоцій у відеоряді та розпізнавати зміни емоційного стану протягом часу. Це особливо корисно для аналізу руху обличчя та зміни виразів під час відтворення емоцій.

Загалом, CNN є потужними інструментами для розпізнавання емоцій на відео завдяки їх здатності автоматично виявляти та використовувати локальні шаблони та ознаки, працювати зі змінами у зображеннях та моделювати динаміку емоцій. Вони відіграють важливу роль у дослідженні розпізнавання емоцій людини на відео у реальному часі.

2.4 Аналіз та підсумки розглянутих методів

FNN мають кілька переваг у контексті розпізнавання емоцій, але мають деякі обмеження порівняно з CNN.

Одна з основних переваг FNN полягає в їх здатності працювати з нечіткою та невизначеною інформацією, що є типовим для емоційних виразів людини. Вони можуть моделювати нечіткі зв'язки та використовувати нечіткі правила для обробки емоційних даних. Це дозволяє їм бути гнучкими та пристосовуватись до різних виразів обличчя та емоційних варіацій.

Однак, порівняно з CNN, FNN можуть мати меншу ефективність у виявленні та розпізнаванні просторових шаблонів у зображеннях, що є важливими для розпізнавання емоцій на зображеннях обличчя. FNN зазвичай мають меншу кількість параметрів та шарів порівняно з CNN, що може обмежити їх здатність до адекватного представлення складних емоційних ознак та залежностей у даних.

Крім того, CNN мають спеціалізовані шари, такі як згорткові шари та пулінг, які ефективно виявляють локальні шаблони та ознаки в зображеннях. Це дозволяє їм бути більш потужними в аналізі образів обличчя та виразів, що сприяє кращому розпізнаванню емоцій на зображеннях та відео.

Отже, FNN є корисними для розпізнавання емоцій, зокрема завдяки їх здатності працювати з нечіткими даними та використовувати нечіткі правила. Проте, в порівнянні з CNN, FNN можуть бути менш ефективними у виявленні просторових шаблонів та складних залежностей у зображеннях обличчя, що може вплинути на точність розпізнавання емоцій.

Щодо RNN, зокрема LSTM, вони мають свої переваги у розпізнаванні емоцій на відео.

Одна з основних переваг RNN, зокрема LSTM, полягає в їх здатності моделювати та ураховувати залежності в часових послідовностях, таких як послідовності відеокадрів. Вони можуть аналізувати динаміку емоційних змін та залежності між різними кадрами, що дозволяє краще розпізнавати емоційні стани на відео. LSTM має комірки пам'яті, які дозволяють запам'ятовувати та використовувати інформацію з попередніх кроків часу, що є важливим для ефективного аналізу послідовних даних.

Крім того, RNN, включаючи LSTM, можуть працювати з різними типами вхідних даних, такими як текстові описи або аудіо, що дозволяє аналізувати емоційний вислів в різних форматах. Вони можуть використовувати цю інформацію, щоб отримати більш повне розуміння емоційної ситуації та забезпечити більш точне розпізнавання емоцій на

відео.

Однак, RNN, включаючи LSTM можуть бути більш обчислювально витратними через потребу у зворотному поширенні помилки протягом всієї послідовності даних. Це може вплинути на швидкість тренування та розпізнавання емоцій у реальному часі.

Крім того, RNN, включаючи LSTM, можуть мати обмежену здатність виявляти просторові шаблони та локальні ознаки, що можуть бути важливими для розпізнавання емоцій на зображеннях обличчя. Вони можуть бути менш ефективними в аналізі просторових залежностей і зображень, що може впливати на точність розпізнавання емоцій на відео.

Отже, RNN, зокрема LSTM, мають переваги в аналізі часових послідовностей та різних типів вхідних даних у розпізнаванні емоцій на відео. Однак, їх обчислювальна складність та обмеженість у виявленні просторових шаблонів порівняно з CNN можуть впливати на їхню ефективність у деяких сценаріях.

Зважаючи усі вищеперечислені факти, ми можемо сказати, що CNN найкращий вибір для нашої задачі, та далі розглянути дану модель більш детально.

3 ТЕОРІЯ РОЗПІЗНАВАННЯ ЕМОЦІЙ ЛЮДИНИ НА ВІДЕО ЗА ДОПОМОГОЮ ЗГОРТКОВИХ НЕЙРОННИХ МЕРЕЖ

CNN – це алгоритм глибинного навчання, який приймає вхідне зображення, призначає важливість (вагові значення та зміщення, які можна дізнатися) різним аспектам або об’єктам зображення та здатний розрізняти зображення. Необхідність попередньої обробки в CNN набагато нижча, ніж в інших алгоритмах класифікації.

На рисунку 3.1 показані операції CNN. Архітектура CNN аналогічна структурі зв’язку нейронів у людському мозку та була натхненна організацією зорової кори. Однією з функцій CNN є скорочення зображень до форми, яку легше обробляти, не втрачаючи функцій, які є критично важливими для якісного прогнозування. Це важливо при розробці архітектури, яка не тільки добре вивчає функції, але й масштабується до масивних наборів даних. Головними CNN операціями є згортка, об’єднання, пакетна нормалізація та вилучення.

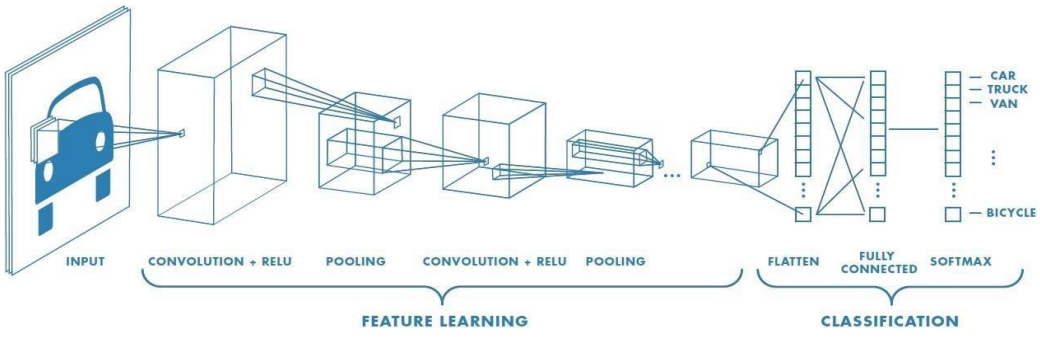

Рисунок 3.1 – Основні операції CNN

3.1 Операція згортки

Мета операції згортки полягає в тому, щоб отримати високорівневі характеристики, такі як границі, із вхідного зображення. Кожний шар згортки розглядає певні ознаки. Перший шар згортки аналізує такі ознаки,

як границі, колір, орієнтація градієнта та прості текстури. На наступному другому шарі аналізуються більш складні текстури та візерунки. На останньому шарі аналізуються об’єкти та частини об’єктів.

Елемент, який бере участь у виконанні операції згортки, називається ядром. Ядро фільтрує все, що не є важливим для карти ознак, зосереджуючись лише на певній інформації. Фільтр рухається вправо з певною довжиною кроку, доки не проаналізує всю ширину. Потім він повертається ліворуч від зображення з такою ж довжиною кроку та повторює процес, доки не буде пройдено все зображення.

На рисунку 3.2 представлено приклад згортання зображення розміром 5 × 5 (показано зеленим кольором) з наступним ядерним фільтром 3 × 3:

$$\begin{matrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{matrix} \tag{3.1}$$

Довжина кроку вибирається як одиниця, тому ядро зміщується дев’ять разів, кожного разу виконуючи матричне множення ядра та частини зображення під ним.

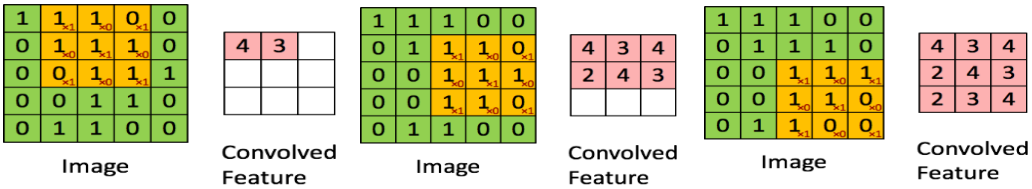

Рисунок 3.2 – Приклад згортання зображення 5 × 5 із ядром 3 × 3, щоб отримати згорнуту функцію 3 × 3

Згорнута функція може мати такі ж розміри, як вхід або ядро. Це робиться за допомогою того самого або дійсного доповнення. Однакове заповнення – це коли згорнута функція має розміри вхідного зображення, а дійсне – коли ця функція має розміри ядра.

3.2 Операція об’єднання

Під час операції об’єднання зменшується просторовий розмір згорнутого об’єкта. Це робиться для того, щоб зменшити обчислення, необхідні для обробки даних і виділення домінуючих ознак, які є інваріантними щодо обертання та положення. Існує два типи об’єднання, а саме: максимальне об’єднання та середнє об’єднання. Максимальне повертає максимальне значення з частини зображення, охопленої ядром, тоді як середнє повертає середнє значення відповідних значень. На рисунку 3.3 показано результати, отримані шляхом об’єднання максимальних і середніх значень зображення.

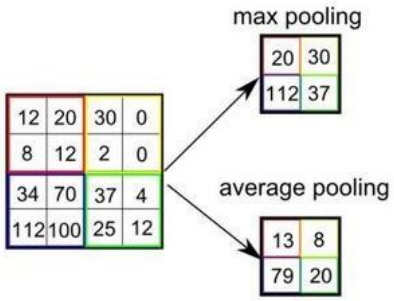

Рисунок 3.3 – Результати максимального та середнього об’єднання для зображення

3.3 Повністю зв’язаний шар

Нейрони в повністю зв’язаному шарі мають зв’язки з усіма нейронами попереднього шару. Цей шар знаходиться в кінці CNN. У цьому шарі вхідні дані з попереднього шару зведені в одновимірний вектор і для отримання результату застосована певна функція активації.

3.4 Виключення

Виключення використовується, щоб уникнути перенавчання. Перенавчання в моделі ML відбувається, коли точність навчання значно перевищує точність тестування. Виключення означає ігнорування нейронів під час навчання, щоб вони не розглядалися під час конкретного проходу вперед або назад, залишаючи зменшену мережу. Ці нейрони вибираються випадковим чином, і приклад показано на рисунку 3.4. Швидкість виключення – це ймовірність навчання даного вузла на рівні, де 1.0 означає відсутність вилучення, а 0.0 означає, що всі виходи з рівня ігноруються.

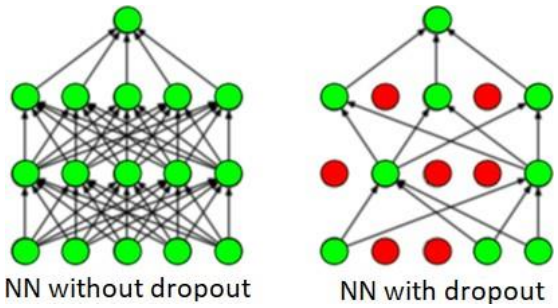

Рисунок 3.4 – Виключення в NN

3.5 Пакетна нормалізація

Навчання мережі є більш ефективним, коли розподіли вхідних даних рівня однакові. Варіації в цих розподілах можуть зробити модель упередженою. Пакетна нормалізація використовується для нормалізації вхідних даних до потрапляння у перші шари моделі.

3.6 Функції активації

Softmax та експоненціальна лінійна одиниця (ELU) є функціями активації, які зазвичай використовуються в CNN.

Функція softmax визначається як:

$$\frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \tag{3.2}$$

де z_i – це вхідні значення, а K – кількість вхідних значень.

Ця функція перетворює дійсні числа на ймовірності, оскільки вона гарантує, що сума вихідних значень дорівнює 1 і знаходиться в діапазоні від 0 до 1. Softmax використовується в повнозв'язному рівні запропонованих моделей, тому результати можна інтерпретувати як розподіл ймовірностей для п'яти емоції. На рисунку 3.5 показано розташування функції softmax.

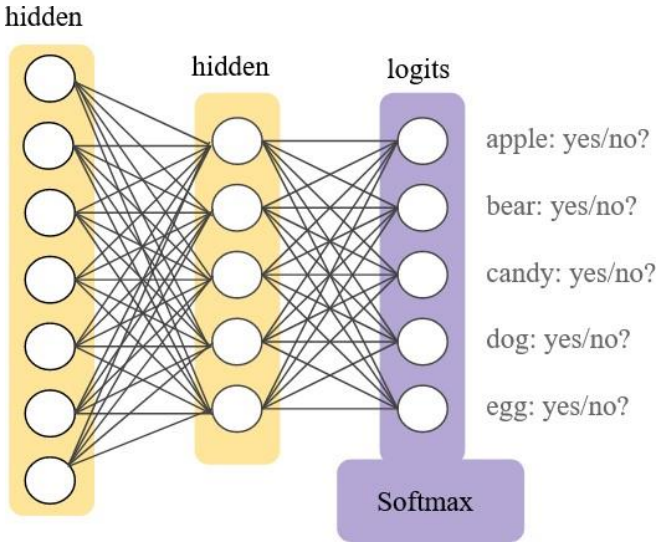

Рисунок 3.5 – Розташування функції softmax

Функція ELU виглядає наступним чином:

$$\begin{cases} x, if\ x > 0 \\ \alpha(e^x - 1), if\ x < 0 \end{cases} \tag{3.3}$$

де x – вхідне значення, а α – нахил.

Ця функція насичується до від’ємного значення, коли x від’ємне, а α контролює насиченість. Це зменшує інформацію, що передається на наступний рівень.

Окрім цих двох популярних функцій існує ще ReLU (Rectified Linear Unit). Ця функція активації використовується широко у CNN. Вона є простою та ефективною, дозволяючи передавати позитивні значення без зміни, а від'ємні значення обнуляються. ReLU допомагає виявляти та активувати релевантні емоційні ознаки.

Наступною популярною функцією активації є Leaky ReLU. Ця варіація ReLU дозволяє невеликий потік від'ємних значень. Це може допомогти запобігти "мертвим" нейронам, коли значення ваги стає від'ємним, і нейрон не активується. Leaky ReLU може бути корисним варіантом, якщо модель має проблему з навчанням.

Останньою популярною функцією активації можна назвати Сігмоід. Ця функція активації обмежує значення на діапазон від 0 до 1. Вона добре підходить для бінарних класифікаційних задач, де необхідно визначити наявність або відсутність певної емоції.

Усі ці функції активації можуть бути використані у різних комбінаціях та шарах CNN в залежності від архітектури моделі та вимог задачі розпізнавання емоцій.

4 ПРОГРАМА ДЛЯ РОЗПІЗНАВАННЯ ЕМОЦІЙ ЛЮДИНИ НА ВІДЕО У РЕЖИМІ РЕАЛЬНОГО ЧАСУ

На основі аналізу найпопулярніших методів розпізнавання емоцій людини у попередніх розділах, було зроблено висновок, що найбільш ефективно буде використовувати CNN. У цьому розділі ми перевіримо цей метод на практиці та проаналізуємо отримані результати.

4.1 Аналіз використовуваних інструментів та бібліотек

Дане програмне рішення було реалізовано мовою Python, адже ця мова программування має простий, але виразний синтаксис, великий вибір бібліотек, та особливо бібліотек з алгоритмами машинного навчання, а також високу культуру документації.

Як інтегровану середу розробки було обрано PyCharm, бо він має досить простий та зрозумілий інтерфейс, а також надзвичайно велику кількість налаштувань.

Бібліотеки, які були використані при створенні програми для розпізнавання емоцій:

- OpenCV – це бібліотека з відкритим кодом для комп’ютерного зору. Це надає машині можливість розпізнавати обличчя, образи або предмети;
- NumPy – це бібліотека для мови програмування Python, яка підтримує великі багатовимірні масиви та NumPy матриці разом із великою колекцією математичних функцій високого рівня для роботи з цими масивами;
- Pandas – це програмна бібліотека, написана на мові програмування Python для обробки та аналізу даних;
- Keras – це бібліотека програмного забезпечення з відкритим кодом, яка надає інтерфейс Python для штучних нейронних мереж;

- Matplotlib – це бібліотека для побудови графіків для мови програмування Python та її розширення чисельної математики NumPy.

4.2 Аналіз робочої станції

Апаратне забезпечення персонального комп’ютера складається з процесора Intel i7 12700, що має 12 ядер та максимальну тактову частоту 4,9 ГГц. Обсяг оперативної пам’яті 32 Гб, частота 3600 МГц. Відеоадаптер – GTX 3080TI з 10240 шейдерними блоками, 12 ГБ відеопам’яті та тактовою частотою приблизно 1665 МГц. Відео буде записуватися за допомогою iPhone XR, який оснащений камерою, здатною знімати у форматі Full HD з частотою 60 кадрів у секунду.

4.3 Розгляд програмного рішення

Програмне рішення складається з наступних ключових етапів. На першому етапі ми створюємо CNN модель за домогою Keras бібліотеки та знаходимо найкращі параметри моделі. Для цього ми спочатку задаємо усі шари для нашої моделі. Вказуємо усі необхідні параметри, такі як: оптимізатор, функцію втрат, метрики, функцію активації. Далі ми створюємо ще більше навчальних даних за допомогою ImageDataGenerator. На наступному кроці ми вже навчаємо модель. Для запобігання перенавчання, ми також використовуємо EarlyStopping та ModelCheckpoint. Наприкінці ми зберігаємо найкращу модель.

Після цього ми можемо приступити до розпізнавання емоцій. Але для цієї задачі нам напочатку потрібно розпізнати образ людини, а саме її обличчя. Для цього ми скористаємося бібліотекою OpenCV, де застосуємо каскади Хаара та натреновану модель обличчя на 68 ознакових точок, як зображено на рисунку 4.1.

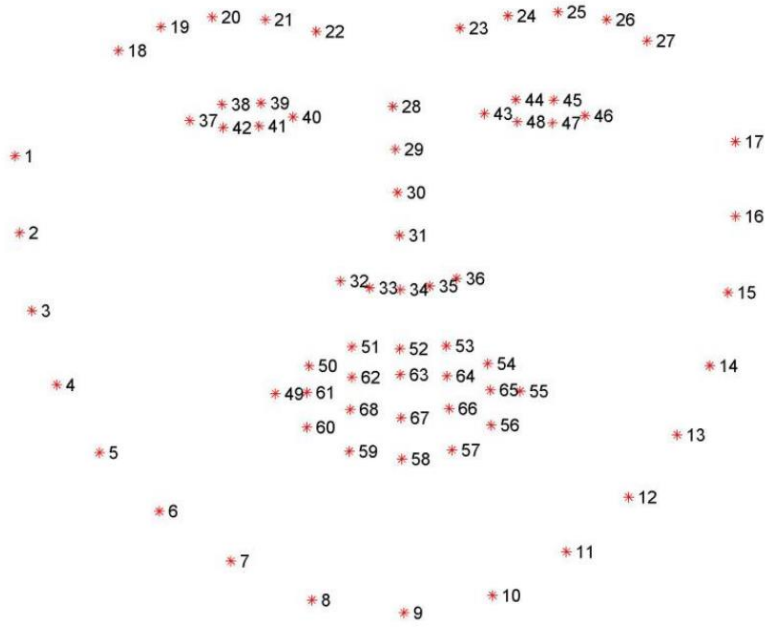

Рисунок 4.1 – Приклад моделі на 68 ознакових точок

На останньому етапі, виділивши обличчя, ми робимо передбачення емоції людини на основі натренованої за допомогою CNN на першому етапі моделі. З кодом програми можна ознайомитися у додатку А.

4.4 Результати роботи програми

Для даної роботи я виділив наступні основні види емоцій людини: злість (рисунок 4.2), щастя (рисунок 4.3), страх (рисунок 4.4), огида, сум (рисунок 4.5), здивування (рисунок 4.6), нейтральна (рисунок 4.7).

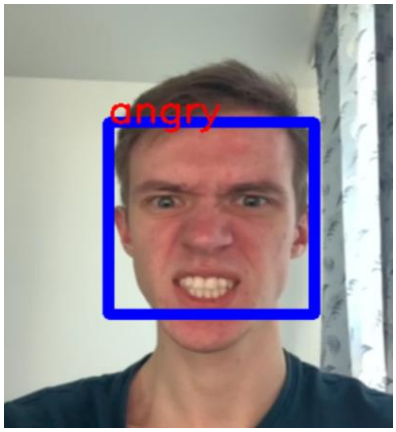

Рисунок 4.2 – Емоція злості

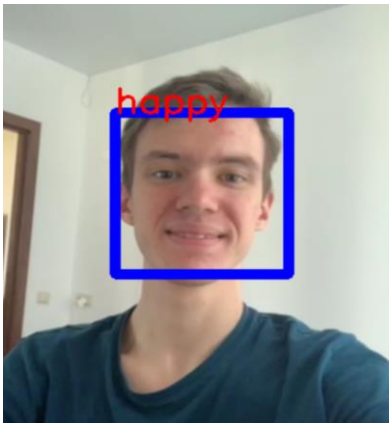

Рисунок 4.3 – Емоція щастя

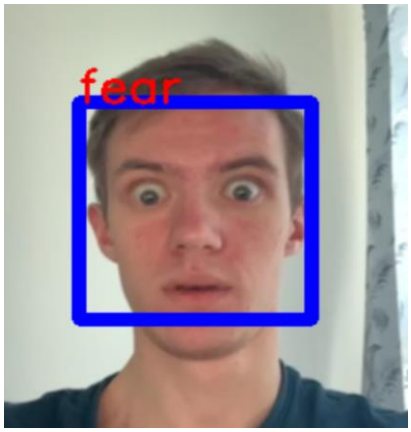

Рисунок 4.4 – Емоція страху

Рисунок 4.5 – Емоція суму

Рисунок 4.5 – Емоція здивування

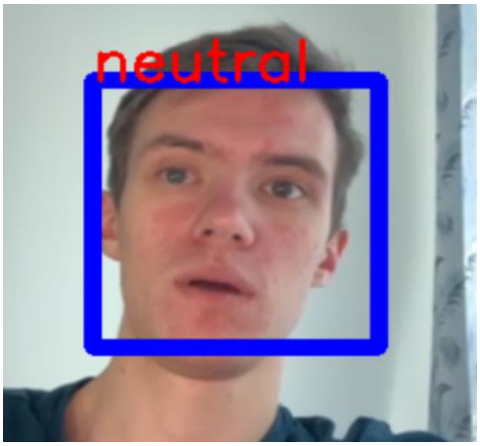

Рисунок 4.7 – Нейтральна емоція

ВИСНОВКИ

В ході виконання даної кваліфікаційної роботи була розглянута предметна область і визначена формальна постановка задачі. Були розглянуті сфери застосування технології розпізнавання емоцій людини на відео у режимі реального часу, а також було проведено аналіз методів та підходів, що найчастіше використовуються для цього завдання.

Була сформована ціль – визначити найефективніший підхід та створити за допомогою нього програмне рішення для розпізнавання емоцій людини.

Було обрано достатньо детальну модель обличчя на 68 ознакових точок. Серед відомих методів запропоновано обрати метод згорткових нейронних мереж.

Було створено програмне рішення, за допомогою якого можна визначати емоції людини на відео у режимі реального часу. Програмна реалізація підтвердила, що обраний метод забезпечує необхідну точність та швидкодію.

Дана технологія щодня розвивається завдяки розвитку нейронних мереж та комп’ютерного зору, а також зі збільшенням обчислювальних потужностей, що доступні людині.

Проаналізувавши результати роботи програми, можна прийти до висновку, що згорткові нейронні мережі мають великий потенціал у розпізнаванні емоцій на відео. Вони добре справляються зі сприйняттям візуальних ознак та шаблонів у зображеннях обличчя. Вони можуть виявляти складні просторові залежності та емоційні ознаки, що сприяє точному розпізнаванню емоцій на відео.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Bodyanskiy Ye., Kulishova N., Chala O., The Extended Multidimensional Neo-Fuzzy System and its Fast Learning in Pattern Recognition Tasks. Data, vol.3, 2018, p. 63.

2. Kulishova N. Ye., Bodyanskiy Ye. V., Tkachenko V. Ph., Feature vector generation for the facial expression recognition using neo-fuzzy system. Radio Electronics, Computer Science, Control, vol.3, 2018, p. 88–96.

3. Bodyanskiy Ye., Kulishova N., Malysheva D. The Extended Neo-Fuzzy System of Computational Intelligence and its Fast Learning for Emotions Online Recognition. Proc. of the 2018 IEEE Second International Conference on Data Stream Mining & Processing (DSMP). (Lviv, Ukraine August 21-25, 2018). 2018. p. 473–478.

4. Challenges in benchmarking stream learning algorithms with real-world data / V. M. A. Souza et al. Data mining and knowledge discovery. 2020. Vol. 34, no. 6. p. 1805–1858. URL: https://doi.org/10.1007/s10618-020-00698-5 (date of access: 25.03.2022).

5. Domingos P., Hulten G. Mining high-speed data streams. The sixth ACM SIGKDD international conference, Boston, Massachusetts, United States, 20–23 August 2000. New York, New York, USA, 2000. URL: https://doi.org/10.1145/347090.347107 (date of access: 25.03.2022).

6. Discussion and review on evolving data streams and concept drift adapting / I. Khamassi et al. Evolving systems. 2016. Vol. 9, no. 1. P. 1–23. URL: https://doi.org/10.1007/s12530-016-9168-2 (date of access: 25.03.2022).

7. Otto P., Bodyanskiy Y., Kolodyazhniy V. A new learning algorithm for a forecasting neuro-fuzzy network. Integrated Computer-Aided Engineering. 2003. Vol. 10, no. 4. p. 399–409. URL: https://doi.org/10.3233/ica-2003-10409 (date of access: 25.03.2022).

8. Bodyanskiy Y., Kolodyazhniy V., Stephan A. An adaptive learning algorithm for a neuro-fuzzy network. Computational intelligence. theory and

applications. Berlin, Heidelberg, 2001. P. 68–75. URL: https://doi.org/10.1007/3-540-45493-4_11 (date of access: 25.03.2022).

9. Golomb L.A., Lawrence D.T. and Sejnowski T.J. SexNet: A neural network identifies sex from human faces // Advances in Neural Information Processing Systems. Morgan Kaufmann Publishers. San Mateo. USA, 1991. 77–83 p.

10. Rowley H. A. Neural Network-Based Face Detection. IEEE Computer Society Conference on Computer Vision and Pattern Recognition – San Francisco, CA, 1996. 208 p.

11. Miki J., Yamakawa J., Analog implementation of neo-fuzzy neuron and its on-board learning, in Computational Intelligence and Applications, Ed. N.E. Mastorakis, Piraeus: WSES Press, 1999, 144 – 149 p.

12. Uchino E., Yamakawa J., Soft computing based signal prediction, restoration and filtering, in Intelligent Hybrid Systems: Fuzzy Logic, Neural Networks and Genetic Algoritms, Ed. Da Ruan, Boston: Kluwer Academic Publishers, 1997, 331 – 349 p.

13. Yamakawa J., Uchino E., Miki J., Kusanagi H., A neo-fuzzy neuron and its application to system identification and prediction of the system behavior, Proc. 2-nd Int. Conf. on Fuzzy Logic and Neural Networks, Iizuka, Japan, 1992, 477 – 483 p.

14. Bodyanskiy Ye., Kulishova N., Extended neo-fuzzy neuron in the task of images filtering, Radioelectronics. Computer Science. Control, № 1(32), 2014, 112 – 119 p.

15. Bodyanskiy Ye., Victorov Y., The cascade of neo-fuzzy architecture and its online learning algorithm, Int. Book Series Inf. Sci. Comput., 17(1), 2010, 110 – 116 p.

16. Bodyanskiy Ye., Kokshenev I., Kolodyazhniy V., An adaptive learning algorithm for a neo-fuzzy neuron, Proc. of the 3rd Conference of the European Society for Fuzzy Logic and Technology, 2005, 375 – 379 p.

17. Zurita D., Delgado M., Carino J.A., Ortega J.A., Clerc G., Industrial

Time Series Modelling by Means of the Neo-Fuzzy Neuron, IEEE Access, vol. 4, 2016, 6151 – 6160 p.

18. Pandit M., Srivastava L., Singh V., On-line voltage security assessment using modified neo-fuzzy neuron based classifier, IEEE Int. Conf. Ind. Technol., 2006, 899 – 904 p.

19. Kim H.D., Optimal learning of neo-fuzzy structure using bacteria foraging optimisation, Proceedings of the ICCA, 2005.

20. Silva A.M., Caminhas W., Lemos A., Gomide F., A fast learning algorithm for evolving neo-fuzzy neuron, Applied Soft Computing, vol. 14, Part B, January 2014, 194 – 209 p.

21. Takagi T., Sugeno M., Fuzzy identification of systems and its application to modeling and control, IEEE Trans. On System, Man and Cybernetics, 15, 1985, 116 – 132 p.

22. Jang J. S., Sun C. T., Mizutani E., Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence. Upper Saddle River: Prentice Hall, 1997.

23. Hu Z., Bodyanskiy Ye., Kulishova N., Tyshchenko O., A Multidimensional Extended Neo-Fuzzy Neuron for Facial Expression Recognition, International Journal of Intelligent Systems and Applications (IJISA), vol.9, No.9, 2017, 29 – 36 p.

24. Goodwin G.C., Ramage P.J., Caines P.E., Discrete time stochastic adaptive control, SIAM J. Control and Optimisation, 19, 1981, 829 – 853 p.

25. Haykin S., Neural Networks. A Comprehensive Foundation. Upper Saddle River: Prentice Hall, 1999.

26. Cichocki A., Unbehauen R., Neural Networks for Optimization and Signal Processing. Stuttgart: Teubner, 1993.

27. Lucey P., Cohn J.F., Kanade T., Saragih J., Ambadar Z. and Matthews I., The Extended Cohn-Kanade Dataset (CK+): A complete dataset for action unit and emotion-specified expression, Proceedings of IEEE workshop on CVPR for Human Communicative Behavior Analysis, San Francisco, USA, 2010.

ДОДАТОК А

Код програми

App.py

```
import os
import cv2
import numpy as np
from keras.preprocessing import image
import warnings
warnings.filterwarnings("ignore")
from keras.preprocessing.image import load_img, img_to_array
from keras.models import load_model
import matplotlib.pyplot as plt

model = load_model("best_model.h5")

face_haar_cascade = cv2.cascadeclassifier(cv2.data.haarcascades
+ 'haarcascade_frontalface_default.xml')

cap = cv2.videocapture(0)

while true:
    ret, test_img = cap.read()
    if not ret:
        continue
    gray_img = cv2.cvtcolor(test_img, cv2.color_bgr2rgb)

    faces_detected                                          =
face_haar_cascade.detectmultiscale(gray_img, 1.32, 5)

    for (x, y, w, h) in faces_detected:
        cv2.rectangle(test_img, (x, y), (x + w, y + h), (255, 0,
0), thickness=7)
        roi_gray = gray_img[y:y + w, x:x + h]
        roi_gray = cv2.resize(roi_gray, (224, 224))
```

```
        img_pixels = image.img_to_array(roi_gray)
        img_pixels = np.expand_dims(img_pixels, axis=0)
        img_pixels /= 255

        predictions = model.predict(img_pixels)

        max_index = np.argmax(predictions[0])

        emotions = ('angry', 'disgust', 'fear', 'happy', 'sad',
'surprise', 'neutral')
        predicted_emotion = emotions[max_index]

        cv2.puttext(test_img,    predicted_emotion,    (int(x),
int(y)), cv2.font_hershey_simplex, 1, (0, 0, 255), 2)

    resized_img = cv2.resize(test_img, (1000, 700))
    cv2.imshow('facial emotion analysis ', resized_img)

    if cv2.waitkey(10) == ord('q'):
        break

cap.release()
cv2.destroyallwindows()
```

Model.ipynb

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

from keras.layers import flatten, dense
from keras.models import model
from keras.preprocessing.image import imagedatagenerator ,
img_to_array, load_img
from keras.applications.mobilenet import mobilenet,
```

```
preprocess_input
from keras.losses import categorical_crossentropy

base_model = mobilenet( input_shape=(224,224,3), include_top=
false )

for layer in base_model.layers:
  layer.trainable = false

x = flatten()(base_model.output)
x = dense(units=7 , activation='softmax' )(x)

model = model(base_model.input, x)

rain_datagen = imagedatagenerator(
     zoom_range = 0.2,
     shear_range = 0.2,
     horizontal_flip=true,
     rescale = 1./255
)

train_data = train_datagen.flow_from_directory(directory=
"/content/train",
target_size=(224,224),
batch_size=32,
)

train_data.class_indices

val_datagen = imagedatagenerator(rescale = 1./255 )

val_data = val_datagen.flow_from_directory(directory=
"/content/test",

target_size=(224,224),batch_size=32,)
```

```
from keras.callbacks import modelcheckpoint, earlystopping

es = earlystopping(monitor='val_accuracy', min_delta= 0.01 ,
patience= 5, verbose= 1, mode='auto')

mc = modelcheckpoint(filepath="best_model.h5", monitor=
'val_accuracy', verbose= 1, save_best_only= true, mode =
'auto')

call_back = [es, mc]

hist = model.fit_generator(train_data,
          steps_per_epoch= 10,
          epochs= 30,
          validation_data= val_data,
          validation_steps= 8,
          callbacks=[es,mc])

from keras.models import load_model
model = load_model("/content/best_model.h5")
```

ДОДАТОК Б

Відомість кваліфікаційної роботи

Позначення	Найменування	Дод. відомості
	Текстові документи	
1.	Пояснювальна записка	51 с.
	Інші документи	
2.	Презентаційні матеріали	__ плакатів

.	.	Прізвище та ініціали.	Підп.	Дата			
Розробив		Кравець А. В.			Дослідження нейромережевих технологій для розпізнавання емоцій людини у реальному часі	Шифр групи	Код напр./спец.
Перевірив		Кулішова Н. Є.				СШІм-21-1	122
Н.контр.		Малєєва І.А.				ХНУРЕ кафедра ШІ	
Затв.		Філатов В.О.					