НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»
ІНСТИТУТ ПРИКЛАДНОГО СИСТЕМНОГО АНАЛІЗУ
КАФЕДРА МАТЕМАТИЧНИХ МЕТОДІВ СИСТЕМНОГО АНАЛІЗУ

На правах рукопису
УДК 004.056.53

До захисту допущено
В. о. завідувача кафедри ММСА
О.Л.Тимощук
«___» _____ 2019 р.

Магістерська дисертація

на здобуття ступеня магістра за спеціальністю 124 Системний аналіз
на тему: «Підходи до розпізнавання емоцій за обличчям із застосуванням активної моделі форми»

Виконав:
студент ІІ курсу, групи КА-81 мп
Браславська Євгенія Вікторівна _____

Керівник:
доцент кафедри ММСА,
к.т.н., доц. Тимошенко Ю.А. _____

Рецензент:
доцент кафедри Прикладної матетатики,
КПІ ім. Ігоря Сікорського,
к.т.н., доц. Маслянко П.П. _____

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

Студент _____

Київ
2019

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ СІКОРСЬКОГО»
ІНСТИТУТ ПРИКЛАДНОГО СИСТЕМНОГО АНАЛІЗУ
КАФЕДРА МАТЕМАТИЧНИХ МЕТОДІВ СИСТЕМНОГО АНАЛІЗУ

Рівень вищої освіти — другий (магістерський)
Спеціальність — 124 «Системний аналіз»

ЗАТВЕРДЖУЮ
В. о. завідувача кафедри ММСА
О. Л. Тимощук
«___» _____ 2019 р.

ЗАВДАННЯ
на магістерську дисертацію студентці Браславській Євгенії Вікторівні

1. Тема дисертації: «Підходи до розпізнавання емоцій за обличчям із застосуванням активної моделі форми», науковий керівник дисертації Тимошенко Юрій Олександрович, к.т.н., доцент, затверджені наказом по університету від «___» _____ № _____

2. Термін подання студентом дисертації: 13 грудня 2019 р.
3. Об’єкт дослідження: розпізнавання емоцій за обличчям людини
4. Предмет дослідження: активна модель форми та методи, що використовуються для класифікації емоцій
5. Перелік завдань, які потрібно розробити:
1) дослідити сучасний стан та особливості застосування математичного моделювання та оптимізації у вирішенні проблеми розпізнавання емоцій за обличчям;
2) розробити математичну модель за допомогою активної моделі форми та класифікатора;
3) розв’язати розроблену математичну модель та на її основі створити програмний продукт;
4) пошук даних для навчання класифікатора;
5) реалізувати метод виділення ключових ознак обличчя людини за допомогою активної моделі форми та тріангуляції Делоне;
6) розробити стартап-проект виведення на ринок результатів дослідження;
7) розробити концептуальні висновки за результатами наукового дослідження.

6. Орієнтовний перелік графічного (ілюстративного) матеріалу:
1). Демонстративні фото баз даних (рис.);
2). Графічні розв'язки тріангуляції Делоне (рис.);
3). Схема роботи рекурентної нейронної мережі (рис.);
4). Таблиці порівняння отриманих результатів з іншими методами (рис.);
5). Таблиці у розділі стартап-проекту.

7. Дата видачі завдання: 05 вересня 2019 р.

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації
1.	Концептуальний вступ дисертації. Формулювання об'єкта, предмета, цілі, завдань, новизни, практичної значущості результатів	05.09.2019—13.09.2019
2.	Перший розділ. Огляд літературно-інформаційних джерел. Понятійно-категоріальний апарат. Характеристика об'єкта	16.09.2019—04.10.2019
3.	Другий розділ. Розробка математичної моделі для задачі розпізнавання емоцій за обличчям з використанням активної моделі форми та класифікатора, її розв'язок	07.10.2019—25.10.2019
4.	Третій розділ. Імплементація отриманих результатів у програмний продукт. Тестування програми	28.10.2019—08.11.2019
5.	П'ятий розділ. Стартап-проект	11.11.2019—22.11.2019
6.	Концептуальні висновки. Перспективи розвитку отриманих рішень	25.11.2019—29.11.2019

Студент Є.В.Браславська

Науковий керівник дисертації Ю.О.Тимошенко

РЕФЕРАТ

Магістерська дисертація: 94 с., 1 ч., 28 табл., 15 рис., 3 дод., 29 джерел.

ГЛИБОКЕ НАВЧАННЯ, РЕКУРЕНТНА НЕЙРОННА МЕРЕЖА, РОЗПІЗНАВАННЯ ЕМОЦІЙ, АКТИВНА МОДЕЛЬ ФОРМИ, ГІСТОГРАМИ НАПРЯМЛЕНИХ ГРАДІЄНТІВ

Об’єкт дослідження – розпізнавання емоцій за обличчям людини за допомогою активної моделі форми та класифікатора.

Мета роботи – покращення розпізнавання емоцій за обличчям при умові наявності обмеженої кількості даних та обмеженого обчислювального ресурсу.

Методи дослідження – моделювання системи знаходження обличчя на фото, системи виділення ключових ознак обличчя, системи класифікації з опорою на добуті ознаки обличчя.

Зроблено приклад системи, що проводить обробку фото та розпізнає базову емоцію обличчя. Визначена ефективність використання активної моделі форми у якості генератора ознак обличчя для подальшого навчання класифікатора.

Здійснено моделювання повної системи обробки та розпізнавання даних, на основі 68 ключових точок обличчя, тріангуляції обличчя та навчання рекурентної нейронної мережі. Досягнуто точності в 72% правильності розпізнавання емоцій.

Упровадження розробленого алгоритму в системах продажу товарів та послуг дозволяє отримувати інформацію щодо задоволення клієнта товаром чи послугою для подальшої оцінки попиту та якості послуг.

ABSTRACTS

The theme: “Approaches to facial emotions recognition using active shape model”.

Master's Thesis: 94 p., 1 part, 28 tabl., 15 fig., 3 appendixes., 29 bibliographic references.

DEEP LEARNING, RECURRENT NEURAL NETWORK, EMOTION RECOGNITION, ACTIVE SHAPE MODEL, HISTOGRAM OF ORIENTED GRADIENTS

Object of research – facial emotion recognition using active shape model.

Purpose of the work – improving facial emotion recognition with limited data and limited computing resources.

Methods of research – modeling of the system to find the face in the photo, also the system to detect the key facial features, and the classification system based on the obtained facial features.

The example of the system was done that performs photo processing and recognizes the basic emotion of the face. The effectiveness of using the active form model as a generator of facial features for further training of the classifier is determined.

A complete data processing and recognition system was modeled on the basis of 68 key face points, face triangulation and recurrent neural network training. 72% accuracy of emotion recognition is achieved.

The implementation of the current system for sales and services systems will help to obtain information about customers satisfaction with a product or service for further evaluation of demand and quality of services.

ЗМІСТ

ВСТУП

Емоції – важливий фактор в людській комунікації. Вони допомагають зрозуміти наміри співрозмовника. Люди розпізнають такі емоції як задоволення, сум, та злість судячи з виразу обличчя та тону голосу співрозмовника. Згідно з дослідженнями [1, 2], вербальна комунікація займає тільки третину людського спілкування, в той час як невербальна – дві третини. Серед способів невербальної комунікації, що несуть в собі емоційний контекст, вираз обличчя є одним з найбільш важливих інформаційних каналів у персональній комунікації. А тому міміка обличчя є важливою галуззю досліджень за останні 25 років, що стосуються не тільки психології, але й комп'ютерних обчислень та людиноподібних систем [2].

Інтерес до автоматичного розпізнавання емоцій підтримується стрімким зростом прийомів машинного навчання, включаючи комунікацію типу комп'ютер-людина [3,4], віртуальну реальність [5], доповнену реальність [6], вдосконалені асистенти водія [7] та розваги [8,9]. Також розпізнавання емоцій має велике практичне значення, особливо для збору відгуків користувачів про будь-який продукт або послугу у бізнесі. Завдяки створеній системі стає можливим проводити детальний аналіз які саме емоції у аудиторії споживачів викликає той чи інший товар та сервіс, що дозволяє будувати більш точні прогнози майбутніх продажів та зміну популярності товарів та послуг, через що дана робота є актуальною.

Наразі існують підходи для розпізнавання емоцій за виразом обличчя за допомогою камери, тобто за фото, але також є роботи з використанням різноманітних сенсорів, таких як інфрачервоні датчики та інше, навіть з використанням ритмів серця. Такі підходи загалом краще справляються зі своєю задачею, але не можна сказати, що вони перевершують методи з використанням фото та відео.

Ціль даної роботи – дослідити різні підходи до розпізнавання емоцій, застосовуючи широкий спектр різноманітних датчиків для виявлення характерних ознак емоцій, а також – різноманітні підходи до вирішення подальшої задачі класифікації. Також у роботі аналізується можливість застосування активної моделі форми для першого етапу роботи, тобто первинної обробки фото.

Алгоритми розпізнавання емоцій можуть бути впроваджені у різноманітні системи для збору статистики про емоційний відгук користувачів товарів чи послуг. Також розпізнавання емоцій може бути використане у сфері штучного інтелекту для покращення комунікації користувачів з штучним інтелектом по типу електронних консультантів та різноманітних ботів, також роботів, що мають взаємодіяти з людиною та проявляти емоції. Ще одне застосування – розпізнавання справжніх емоцій людини у сфері криміналістики та правопорядку, щоб визначати зловмисників на етапі планування злочину, наприклад, підривників у місцях великого скупчення людей.

На відміну від існуючих робіт, що досліджують можливості застосування активної моделі форми для розпізнавання емоцій за виразом обличчя, у цій роботі дослід проводиться в умовах сильно лімітованих ресурсів, як обчислювальних, так і даних для навчання. До того ж, обмежені ресурси для виконання розпізнавання згідно побудованому алгоритму, щоб використання алгоритму розпізнавання було можливе на багатьох користувацьких пристроях, що не є добре технічно оснащеними.

РОЗДІЛ 1. ОГЛЯД ОБЛАСТІ

Існує багато робіт, які вирішують задачу розпізнавання емоцій людини за виразом обличчя. Усі вони пропонують різноманітні підходи та методи для досягнення результату, але загалом можна виділити два основні етапи проведення розпізнавання: аналіз атрибутів обличчя та подальша класифікація здобутих результатів.

Першим етапом проведення розпізнавання емоцій за виразом обличчя є зняття даних про стан обличчя на момент розпізнавання. Для вирішення поставленої задачі за останні 25 років були створені різноманітні сенсори, такі як електроміограф, електрокардіограф, електроенцефалограф. Також у галузі активно використовується аналіз аудіофайлів, фото та відео контенту за допомогою різноманітних програмних засобів. Одним з перспективних джерел отримання такого роду інформації є поширені веб-камери, так як вони дозволяють отримати повну інформацію про емоційний стан людини, є доступними та дешевими, а також не потребують безпосереднього контакту з тілом. Саме тому надалі зосередимося на більш детальному аналізі методів обробки зображень та відео зі звичайних або інфрачервоних камер.

Розпізнавання емоцій обличчя за типом вхідних даних може бути поділене на дві групи – таке, що використовує для розпізнавання зображення, та таке, що використовує відео [13]. Перший тип, розпізнавання зі статичної картинки, опирається виключно на статичні ознаки обличчя, що добуті шляхом вилучення певних ознак з вибраних пікових кадрів з послідовності зображень. Другий тип, динамічне розпізнавання, використовує просторово-часові ознаки, щоб розпізнати динаміку емоції в послідовності виразів обличчя. Хоча динамічне розпізнавання має кращі показники розпізнавання ніж статичне розпізнавання, так як використовує додаткову інформацію, у нього є декілька проблем. Наприклад, виділені динамічні ознаки мають різну

тривалість переходу та різні характеристики ознак обличчя в залежності від певної людини. Більше того, часова нормалізація, що використовується для того, щоб отримати послідовність виразів обличчя з фіксованою кількістю кадрів, може призвести до втрат в часовому масштабі.

Усі дослідження за методологією можна умовно поділити на ті, що зроблені за допомогою нейронних мереж, та ті, що зроблені за допомогою інших методів.

Будь який процес розпізнавання обличчя містить у собі три головних компоненти, що зображені на рисунку 1.1:

1) знаходження обличчя та його компонентів на фото/відео;
2) виділення певних атрибутів, необхідних для розпізнавання;
3) класифікація отриманого результату.

Рисунок 1.1 – Процедура розпізнання емоцій людини з використанням нейронної мережі [3]

На відміну від традиційних класичних підходів, глибоке навчання – це базовий підхід для машинного навчання. Зазвичай таке навчання потребує великої кількості даних для того, щоб досягти конкуруючих результатів. Саме тому при малих об’ємах бази даних більш доцільно використовувати класифікатори, адже результати нейронних мереж будуть поступатись тим результатам, що демонструють на цьому рівні класифікатори [11].

Підходи до розпізнавання емоцій за виразом обличчя, що базуються на глибокому навчанні, мають набагато меншу залежність від побудови моделі обличчя, на відміну від підходів, що засновані на фізиці обличчя, або ті, що використовують інші техніки попередньої обробки фото, адже в якості входу використовую фото [12]. Серед усіх наявних моделей глибокого навчання, найбільш популярною є згорткова нейронна мережа (CNN). У згортковій нейронній мережі вхідне зображення згортається за допомогою набору фільтрів в згорткових шарах, щоб на виході отримати карту ознак. Усі карти ознак потім комбінуються у повністю з'єднану мережу, і тоді вираз обличчя можна трактувати як належний до певного класу базуючись на виході усього алгоритму.

1.1 Термінологія

Перед тим як перейти до розгляду результатів розпізнавання емоцій обличчя, наведемо спеціальну термінологію області:

— Система кодування дії обличчя (СКДО) – це система, що базується на русі м'язів обличчя і може описувати дії обличчя для вираження індивідуальних емоцій, яка описана Екманом і Фрізеном [3] в 1978 році. СКДО кодує рухи певних м'язів обличчя як одиниці дії, що відображають чіткі миттєві зміни зовнішнього вигляду обличчя [3];

— Орієнтири обличчя – це візуально помітні точки на обличчі, такі як кінець носа, кінці брів та рота. Парні позиції кожного з двох орієнтирів або локальна текстура орієнтира використовується як вектор ознак для розпізнавання емоцій обличчя. Загалом підходи для знаходження орієнтирів обличчя можна класифікувати на чотири типи відповідно до способу генерування таких моделей: модель на основі фігури (ASM), модель на основі зовнішнього вигляду (AAM), модель на основі регресії з комбінацією

локальних та глобальних моделей та моделі на основі згорткових нейронних мереж. Моделі орієнтирів обличчя – це моделі що навчаються за зовнішнім виглядом. Спочатку модель отримує приблизну грубу ініціалізацію. Потім початкову форму переміщують у кращу позицію крок за кроком до повного зближення з правильним варіантом [2];

— Основні емоції – це сім основних людських емоцій: щастя, здивування, гнів, смуток, страх, відраза та нейтральність, як показано на рисунку 1.2;

— Складні емоції – це поєднання двох основних емоцій. Існує 12 складних емоцій, що найчастіше виражають люди, та ще три додаткові емоції (приголомшення, ненависть і побоювання). Вони зображені на рисунку 1.2;

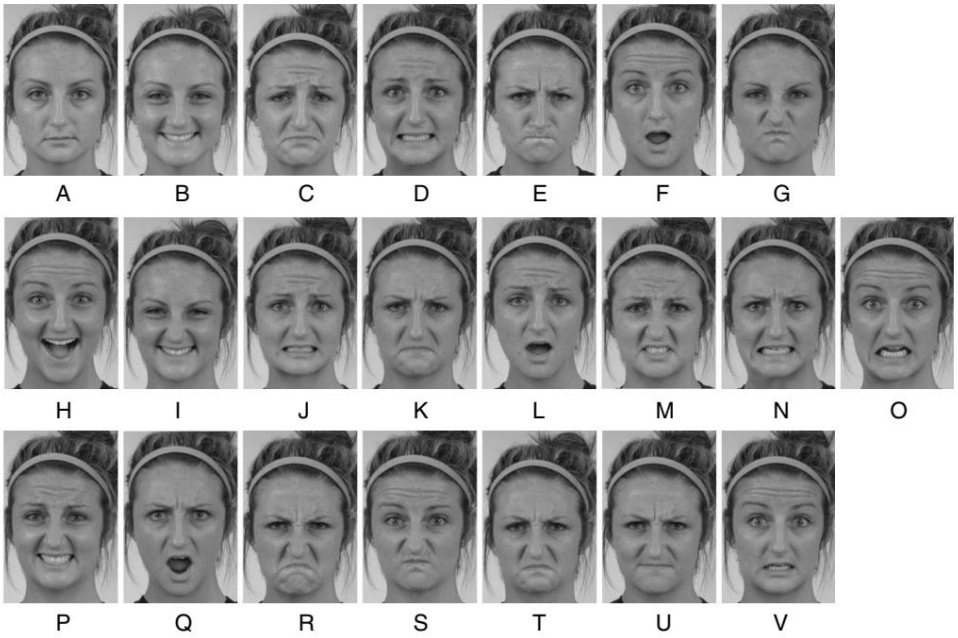

Рисунок 1.2 – Приклади базових емоцій: (A) – нейтральність, (B) – щастя, (C) – сум, (D) – страх, (E) – гнів, (F) – здивування, (G) – відраза; приклади 12 основних складних емоцій: (H) – радісне здивування, (I) – радісна огидність, (J) – сумна боязливість, (K) – сумна злість, (L) – сумна здивованість, (M) – сумна огидливість, (N) – злякана злість, (O) – злякане здивування, (P) – злякана огидність, (Q) – сердите здивування, (R) – сердита огидність, (S) – здивування з відразою, (T) – приголомшення, (U) – ненависть, (V) - побоювання [2].

— Мікроемоції вказують на більш спонтанні та тонкі рухи обличчя, які відбуваються мимоволі. Вони, як правило, видають справжні та основні емоції людини на короткому проміжку часу;

— Одиниці мімічної дії кодують 46 основних дій окремих індивідуальних або групи м'язів, які, як правило, видно під час створення певної емоції [5]. Щоб розпізнати емоції обличчя, виявляють окремі одиниці мімічної дії, і система класифікує категорію емоцій з виразу обличчя відповідно до комбінації одиниць мімічної дії. Наприклад, якщо зображення було позначене як таке, що містить 1-шу, 2-гу, 25-ту та 26-ту одиниці мімічних дій, то система класифікує його як вираження емоції категорії «здивований», як зазначено у таблиці 1.1.

Таблиця 1.1 - Типові одиниці мімічної дії, що спостерігаються у кожній базовій та складній емоції [5]

Категорія	Одиниці мімічної дії	Категорія	Одиниці мімічної дії
Радість	12, 25	Сумна огидливість	4, 10
Сум	4, 15	Злякана злість	4, 20, 25
Страх	1, 4, 20, 25	Злякане здивування	1, 2, 5, 20, 25
Гнів	4, 7, 24	Злякана огидність	1, 4, 10, 20, 25
Здивованість	1, 2, 25, 26	Сердите здивування	4, 25, 26
Відраза	9, 10, 17	Сердита огидність	1, 2, 5, 10
Радісний сум	4, 6, 12, 25	Здивування з відразою	1, 2, 12, 25, 26

Кінець таблиці 1.1

Категорія	Одиниці мімічної дії	Категорія	Одиниці мімічної дії
Радісне здивування	1, 2, 12, 25	Радісна зляканість	4, 10, 17
Радісна огидність	10, 12, 25	Побоювання	1, 2, 5, 25
Сумна боязливість	1, 4, 15, 25	Приголомшення	4, 9, 10
Сумна злість	4, 7, 15	Ненависть	4, 7, 10
Сумна здивованість	1, 4, 25, 26	-	-

1.2 Класичні підходи для розв’язання задачі розпізнавання емоцій за обличчям

У сфері автоматичних систем для розпізнавання емоцій за виразом обличчям було досліджено багато різних підходів. Спільним для цих підходів є проблема виявлення області обличчя та знаходження геометричних рис, особливостей зовнішнього вигляду або поєднання геометричних та зовнішніх рис заданого обличчя.

Для підходів, заснованих на виявленні геометричних особливостей, взаємозв’язок між компонентами обличчя використовується для побудови функціонального вектора, що використовується для подальшого навчання [7]. Наприклад, Гірім та Лі [7] використовували два типи геометричних рис, що базувались на положенні та куті нахилу 52 орієнтирів обличчя. Перша геометрична риса - кут нахилу та евклідова відстань між кожною парою ознак. Друга – відстань та кути віднімались від відповідних вимірів з першого кадру відеоряду. В подальшому для класифікатора пропонувались

два підходи: метод AdaBoost з динамічним виявленням у часі, або ж SVM на підсилених векторах функцій.

Особливості зовнішнього вигляду зазвичай беруться з усього обличчя одразу або з різних областей обличчя, що містять різні типи інформації. Як приклад використання глобальних ознак, Хеппі [8] використовував локальну бінарну схему (LBP) – гістограми у вигляді блоків різних розмірів для усього обличчя як вектори ознак – та класифікував різні вирази обличчя за допомогою аналізу основних компонентів (PCA). Хоча цей метод реалізовано у режимі реального часу, точність розпізнавання має тенденцію до погіршення, оскільки він не може відобразити локальні відмінності компонентів обличчя у векторних ознаках. На відміну від підходу, заснованого на рисах усього обличчя, у методах, що використовують тільки регіони, використовують пріоритет цих регіонів. Різні регіони обличчя мають різний рівень важливості. Наприклад, очі та рот містять більше інформації, ніж лоб та щоки. Гімір [7] визначав ознаки зовнішнього вигляду шляхом поділу всієї області обличчя на місцеві регіони, характерні для домену. Важливі місцеві регіони визначаються за допомогою додаткового методу пошуку, що призводить до зменшення розмірних характеристик та покращення точності розпізнавання.

Що стосується гібридних ознак, деякі підходи [9] поєднали геометричні та зовнішні ознаки, щоб доповнити слабкі сторони обох підходів та забезпечити ще кращі результати в певних випадках.

У випадку відеоряду, багато систем [9, 10] вимірюють геометричне переміщення орієнтирів обличчя між поточним та попереднім кадрами як часові особливості та визначають ознаки зовнішності як просторові особливості. Основна відмінність розпізнавання емоцій по обличчю для нерухомих зображень та відеоряду полягає в тому, що орієнтири в останніх відслідковуються кадр за кадром і система генерує нові динамічні ознаки шляхом переміщення між попереднім та поточним кадрами. Алгоритми

класифікації у двох типах розпізнавання подібні. Для розпізнавання мікроекспресії використовуються камери високої швидкості для зйомки великих відеорядів. Поліковський [10] бере за основу для досліджень мікроекспресій відеоряди, зняті на швидкості не менше ніж 200 кадрів в секунду. В його дослідженнях ділянки обличчя діляться на конкретні регіони, після чого з руху в кожній області формується гістограма тривимірних градієнтів для подальшого розпізнавання.

Крім розпізнавання емоцій за виразом обличчя з двовимірних зображень, тривимірні та чотиривимірні (динамічні тривимірні) моделі усе частіше використовуються в дослідженні аналізу експресії. Ці підходи виникли через проблеми, які присутні в двовимірних зображеннях, такі як варіації пози та освітленість. Тривимірне розпізнавання виразів обличчя складається із знаходження та класифікації ознак, як і у випадку з двовимірним розпізнаванням. Слід зауважити, що динамічна та статична системи дуже відрізняються між собою через характер наданих даних. У статичних системах особливості знаходять за допомогою статистичних моделей, таких як деформаційна модель, модель активної форми, аналіз двовимірних уявлень та функції на основі відстані. На відміну від статичних, динамічні системи використовують послідовності тривимірних зображень для аналізу міміки, використовуючи тривимірні ознаки, що базуються на русі. Для тривимірних систем такому застосовують аналогічні з двовимірними зображеннями алгоритми класифікації [11]. Хоча розпізнавання емоцій за обличчям на основі тривимірного зображення демонструвало більш високу продуктивність, ніж на основі двовимірного зображення, тривимірні та чотиривимірні концепції також мають власні проблеми, такі як висока обчислювальна вартість через високу роздільну здатність і частоту кадрів, а також кількість тривимірної інформації, що потрібна для розпізнавання.

Деякі дослідники [12, 13] намагалися розпізнати емоції по обличчю, використовуючи інфрачервоні зображення замість зображення спектра

видимого світла, оскільки зображення видимого світла змінюється залежно від стану освітленості. Чжао та інші [12] використовують відзняті відеопослідовності з близьким до інфрачервоного освітленням та дескриптори LBP-TOP (локальні бінарні патерни з трьох ортогональних площин). У цьому дослідженні використовуються ознаки обличчя на основі компонентів для поєднання геометричної та зовнішньої інформації з обличчя. Для розпізнавання емоцій використовуються класифікатори SVM та розрідженого представлення. Шен та інші [13] використовували інфрачервоні теплові відеозаписи, розраховуючи різницю горизонтальних та вертикальних температур з різних підрегіонів обличчя. Для розпізнавання використовується алгоритм Adaboost зі слабкими класифікаторами k-найближчих сусідів. Своч та Пеніазек [14] розпізнавали вирази обличчя та емоції, засновані лише на глибинному каналі від сенсора Microsoft Kinect без використання камери. У цьому дослідженні використовуються локальні рухи в області обличчя як ознаки, а розпізнавання міміки відбувається з використанням залежностей між певними емоціями. Суджоно та Гунаван [15] використовували датчик руху Kinect для виявлення області обличчя на основі інформації про глибину на фото та моделі активної зовнішності (AAM) для відстеження виявленого обличчя. Роль AAM – коригування форми та фактурної моделі на новому обличчі, використовуючи варіанти форми та фактури із результатів тренувань. Для розпізнавання емоцій за виразом обличчям використовується зміна основних ознак AAM та нечіткої логіки на основі попередніх знань, отриманих з системи кодування дій обличчя. Вей та інші [16] запропонували розпізнавання емоцій за виразом обличчя за допомогою датчика Kinect та інформації про колір одночасно. У цьому дослідженні знаходяться векторні точки ознак обличчя за алгоритмом стеження за обличчям, використовуючи надані датчиком дані, та за їх допомогою розпізнають шість емоцій за алгоритмом випадкового лісу.

Зазвичай у традиційних підходах ознаки та класифікатори визначаються з допомогою експертів. Для знаходження ознак використовують багато відомих функцій, такі як гістограма градієнтів, локальні бінарні патерни, відстань та кутове відношення між орієнтирами, та попередньо підготовлені класифікатори, такі як опорні вектори, AdaBoost та випадковий ліс, які використовуються для класифікації отриманих ознак. Класичні підходи вимагають порівняно меншої обчислювальної потужності та пам'яті, на відміну від підходів, заснованих на глибокому навчанні. Тому ці підходи досі вивчаються для використання у вбудованих системах у режимі реального часу через їх низьку обчислювальну складність та високу для даних умов ступінь точності. Однак програми для виявлення ознак та класифікатори повинні бути розроблені програмістом, і вони не можуть бути загалом оптимізовані для підвищення продуктивності.

1.3 Підходи на основі глибокого навчання для розпізнавання емоцій по обличчю

В останні десятиліття відбувся прорив алгоритмів глибокого навчання, застосованих у сфері комп’ютерного зору, включаючи згорткову нейронну мережу (CNN) та рекурентну нейронну мережу (RNN). Ці алгоритми на основі глибокого навчання використовуються для знаходження ознак, їх класифікації та розпізнавання. Основна перевага CNN полягає в тому, щоб повністю уникнути або сильно зменшити вплив залежності від фізичних особливостей моделей та/або інших методів попередньої обробки, застосувавши навчання «від кінця до кінця», виходячи безпосередньо з вхідних зображень [16]. З цієї причини CNN досягла найбільших результатів у різних областях, включаючи розпізнавання об'єктів, розпізнавання обличчя, розуміння оточення та розпізнавання емоцій за виразом обличчям.

CNN містить три типи гетерогенних шарів: згорткові шари, шари підвибірки та повністю з’єднані шари. Згорткові шари беруть зображення або карти ознак як вхідні дані і згортають вхідні дані за допомогою наборів фільтрів на манер ковзкого вікна для отримання карт ознак, які представляють просторове розташування обличчя на зображенні. Ваги згорткових фільтрів для карт ознак одного шару є спільними, а входи шару – локально пов'язані [12]. По-друге, шари підвибірки зменшують розмір входу шляхом усереднення або взяття максимального значення заданих карт вхідних даних, щоб зменшити їх розміри і тим самим ігнорувати зміни невеликих зрушень та геометричних спотворень [13]. Останні повністю з'єднані шари структури CNN обчислюють відсоток належності вхідного зображення до певного класу емоцій. Більшість методів на основі глибокого навчання [13] адаптували CNN безпосередньо для виявлення обличчя на зображенні і використовують результат роботи мережі як вектор ознак для класифікатора.

Брейер та Кіммель [17] використовували методи візуалізації CNN, щоб зрозуміти модель, навчену за допомогою різних наборів даних для розпізнавання емоцій за виразом обличчям, та продемонстрували можливості нейронних мереж, навчених розпізнаванню емоцій, як за набором даних, так і для різних завдань, пов'язаних з розпізнаванням емоцій. Юнг та ін. [18] використовували два різні типи CNN: перша отримує часові зовнішні ознаки із послідовностей зображень, тоді як друга отримує часові геометричні риси з часових орієнтирів обличчя. Ці дві моделі поєднуються за допомогою методу інтеграції для підвищення ефективності розпізнавання емоцій обличчя.

Чжао та ін. [19] запропонували метод регіонального та багаторівневого глибокого навчання (DRML), який є єдиною глибокою мережею. DRML - це регіональний шар, який використовує функції зворотного зв’язку для визначення важливих областей обличчя та змушує навчені ваги захоплювати структурну інформацію обличчя. Повна нейронна мережа навчається

самостійно та автоматично вивчає ключові ознаки у всіх варіаціях, що є властивими локальному регіону.

Як ми визначили в нашому огляді, багато підходів використовують CNN безпосередньо для розпізнавання емоцій за обличчям. Однак, оскільки методи на основі CNN не можуть відображати часові зміни в компонентах обличчя, був розроблений гібридний підхід, що поєднує CNN для просторових особливостей окремих кадрів, і довгострокову короткочасну пам'ять (LSTM) для тимчасових особливостей послідовності кадрів. LSTM - це особливий тип RNN, розроблений для вирішення проблеми довгострокової залежності з використанням короткочасної пам'яті. LSTM має ланцюгоподібну структуру, хоча модулі, що повторюються, мають відмінну структуру, як показано на рисунку 1.3.

Усі рекурентні нейронні мережі мають ланцюгову форму з чотирьох однакових модулів нейронної мережі. Один модуль має наступні компоненти [20]:

— Стан комірки - це горизонтальна лінія, що проходить через верхню частину діаграми, як показано на рисунку 1.3. LSTM має можливість видаляти або додавати інформацію до стану комірки;

— Шар забуття використовується для того, щоб визначити, яку нову інформацію зберігати в стані комірки;

— Шар вхідного сигналу використовується для визначення того, які значення будуть оновлені в комірці;

— Шар вихідного сигналу забезпечує виходи на основі стану комірки.

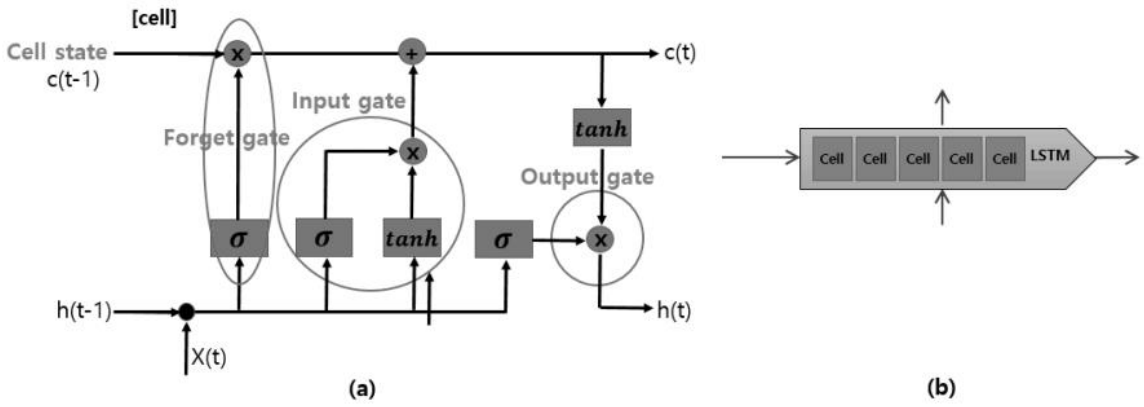

Рисунок 1.3 – Основна структура LSTM, адаптована з [20]. (a) Одна комірка LSTM містить чотири взаємодіючих шари: стан комірки, шар вхідного сигналу, шар забуття та шар вихідного сигналу, (b) Повторюваний модуль

Модель LSTM або RNN для моделювання послідовних зображень має дві переваги порівняно з автономними підходами. По-перше, моделі LSTM є простими в плані мінімальної настройки на всьому шляху роботи, коли вони інтегруються з іншими моделями, такими як CNN. По-друге, LSTM підтримує входи та виходи фіксованої або змінної довжини [20].

Показові дослідження, що використовують комбінацію CNN та LSTM (RNN), включають наступне:

Кахоу та ін. [20] запропонував гібридний фреймворк RNN-CNN для поширення інформації про послідовність, використовуючи інформацію на прихованому шарі, який постійно переоцінюється. У цій роботі автори представили закінчену систему для конкурсу «Розпізнавання емоцій наживо» (EmotiW) 2015 року та довели, що гібридна архітектура CNN-RNN для аналізу виразів обличчя може перевершити раніше застосований підхід CNN, якщо використовувати часове усереднення для агрегації.

Кім та ін. [13] використовували характерні стани емоцій (наприклад, початок, пікова точка та зміна виразів обличчя), які можуть бути визначені в послідовностях виразів обличчя незалежно від інтенсивності вираження

емоцій. Просторові ознаки показових кадрів стану виявлення емоцій отримуються за допомогою CNN. У другій частині тимчасові просторові ознак, що були знайдені у першій частині, класифікуються за допомогою LSTM.

Чу та ін. [14] запропонували багаторівневий алгоритм виявлення орієнтирів обличчя, що поєднує просторові та часові ознаки. Просторові ознаки знаходяться за допомогою CNN, що здатна зменшити вплив на дані специфічних особливостей кожної людини, що виділяють такі дескриптори, як HoG та Gabor. Для виявлення часових залежностей застосовують LSTM, вхід якої не залежить від довжини вхідних відеопослідовностей. Виходи CNN та LSTM додатково об'єднуються в єдину мережу для отримання прогнозування результату на кожному кадрі для 12 орієнтирів обличчя.

Хасані та Махур [15] запропонували архітектуру 3D Inception-ResNet з подальшим застосуванням LSTM, яка разом виявляє просторові та часові залежності ознак обличчя між різними кадрами у відеопослідовності. Орієнтири обличчя також використовуються як вхідні дані цієї мережі, підкреслюючи більшу важливість компонентів обличчя, а не його областей. Але це може погіршувати розпізнавання міміки обличчя.

Грейвс та ін. [16] використовували зворотну мережу для виявлення часових залежностей, наявних у послідовностях зображень, для подальшого проведення класифікації. В результатах експериментів із використанням двох типів LSTM (двонаправлена LSTM та однонаправлена LSTM) цим дослідженням було доведено, що двонаправлена мережа забезпечує значно кращі показники, ніж однонаправлена.

Джайн та ін. [17] запропонував метод глибокого навчання на основі шаблону (MAOP-DL), що оптимальний для різних кутів світла, щоб побороти проблему раптової зміни освітленості. Використовуючи оптимальні конфігурації, у цій роботі автори намагались знайти правильний набір ознак, які б враховували положення світла. Спочатку такий підхід видаляє фон і

відділяє передній план від усього зображенння, а потім визначає текстурні патерни та відповідні ключові ознаки обличчя. Потім відповідні ознаки вибірково вибираються, та застосовується LSTM-CNN для прогнозування відповідної емоції для заданого виразу обличчя.

На відміну від класичних підходів, для підходів, заснованих на глибокому навчанні, методи та класифікатори зазвичай підбирають експерти з глибоких нейронних мереж. Підходи на основі глибокого навчання отримують оптимальні значення з бажаними характеристиками безпосередньо з вхідних тестових даних за допомогою глибоких згорткових нейронних мереж. Однак зібрати велику кількість даних для тренування розпізнаванню емоцій за виразом обличчя, що мали б широку варіацію можливих умов розпізнавання, достатнього для того, щоб навчити глибокі нейронні мережі досить складно. Більше того, підходи, засновані на глибокому навчанні, вимагають більш потужної та складно побудованої обчислювальної техніки для проведення навчання та тестування, на відміну від класичних підходів. Тому при підборі моделі необхідно намагатись зменшити обчислювальне навантаження під час тренування алгоритму глибокого навчання.

Як визначено у розділі раніше, гібридні підходів CNN-LSTM та CNN-RNN для розпізнавання емоцій за обличчям мають подібні структури, як показано на рисунку 1.4. Підсумовуючи, основні принципи CNN-LSTM (RNN) – це поєднання LSTM з ієрархічним екстрактором візуальних ознак, побудованим на основі глибокого навчання, таким як модель CNN. Тому ця гібридна модель може навчитися розпізнавати та синтезувати часову динаміку для завдань, що включають послідовні зображення. Як показано на рисунку 1.4, кожна візуальна ознака, визначена за допомогою CNN, передається відповідній LSTM, щоб на виході утворити векторне подання зображення з фіксованою або змінною довжиною. Потім результати передаються в наступний модуль для рекурентного навчання. Нарешті,

прогнозована відповідь, тобто класифікація емоцій, обчислюється за допомогою застосування softmax [20].

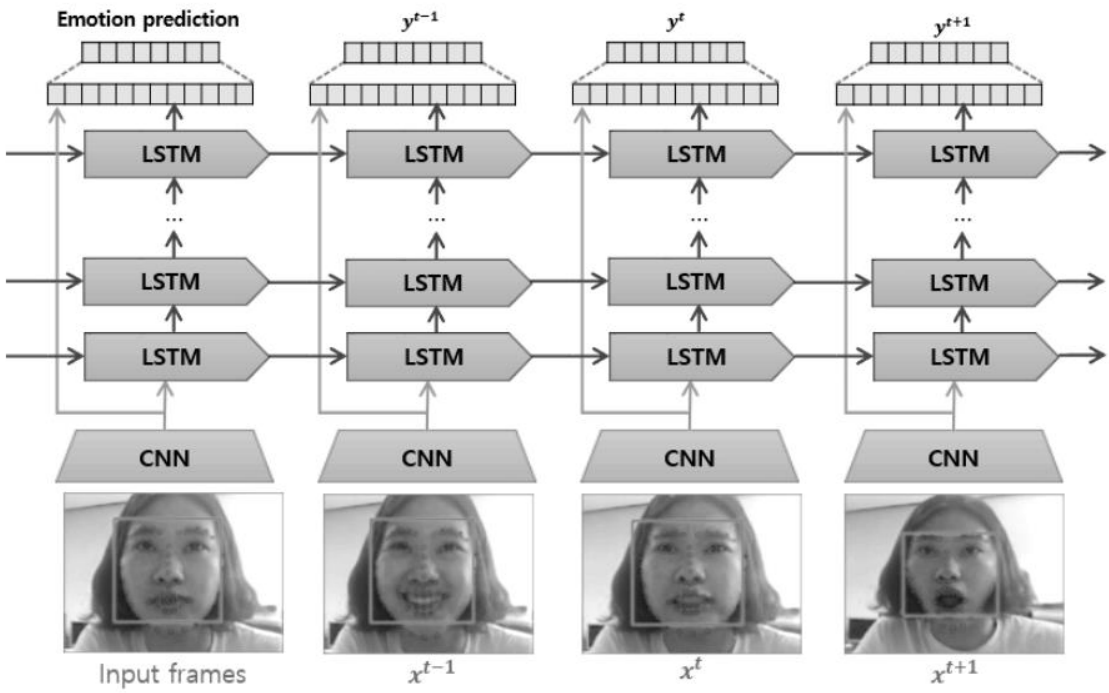

Рисунок 1.4 – Огляд загального гібридного підходу для глибокого навчання задачі розпізнавання емоцій за виразом обличчя. Виходи CNN та LSTM додатково об'єднуються в мережу, що об’єднує обидва методи, для отримання прогнозу для кожного кадру, адаптовано з [20].

1.4 Огляд баз даних для розпізнавання емоцій за виразом обличчям

У галузі розпізнавання емоцій за виразом обличчям для порівняння та численних експериментів використовуються численні бази даних. Традиційно емоції обличчя людини вивчалися за допомогою 2D-зображень або 2D-послідовностей відео. У разі 2D аналізу виникають труднощі з обробкою сильних варіацій пози обличчя та ледве помітної міміки обличчя.

В свою чергу аналіз тривимірних моделей обличчя сприяє вивченню найменших структурних змін, властивих спонтанному вираженню емоцій. Тому в цьому підрозділі коротко представлені деякі популярні бази даних, пов'язані з розпізнаванням емоцій, що складаються з 2D та 3D-послідовностей відео та нерухомих зображень:

— Розширена база даних Кон-Канаде (CK+) [10]: CK+ містить 593 набори відео послідовностей, що демонструють як спеціально поставлені, так і вільні (спонтанні) емоції, а також додаткові типи метаданих. Віковий діапазон 123 моделей становить від 18 до 30 років, більшість з яких жінки. Послідовності зображень можуть бути проаналізовані як на предмет одиниць дій, так і для базових емоцій. База надає протоколи та базові результати для відстеження ознак обличчя, вектори одиниць дії та розпізнавання емоцій. Зображення мають роздільну здатність 640 × 480 та 640 × 490 пікселів з 8-бітовою точністю для значень градації сірого;

— Складні емоції (CE) [17]: CE містить 5060 зображень, що відповідають 22 категоріям основних та складних емоцій для 230 різних людей (130 жінок та 100 чоловіків, середній вік 23 роки). База даних містить представників більшості етносів і рас, включаючи кавказців, азіатів, африканців та латиноамериканців. Оклюзії обличчя зведені до мінімуму, усі моделі без окулярів та без волосся на обличчі. Піддослідних чоловіків попросили поголити обличчя якомога чистіше, а всіх учасників – відкрити лоб, щоб були повністю видні брови. Кольорові фотографії зроблені за допомогою Canon IXUS з роздільною здатністю 3000 × 4000 пікселів;

— Денверська база даних спонтанних емоцій обличчя (DISFA) [22]: DISFA складається з 130000 стерео кадрів з відео із високою роздільною здатністю (1024 × 768), усього 27 дорослих людей (12 жінок та 15 чоловіків) з різних етносів. Наявність одиниць дії (шкала від 0 до 5) для всіх відеокадрів оцінювали вручну два експерти з FACS. База даних також включає 66 міток

на обличчі для кожного зображення з бази даних. Оригінальний розмір кожного зображення – 1024 × 768 пікселів;

— 3D вирази обличчя університету Бінгемтона (BU-3DFE) [23]: Оскільки 2D нерухомі зображення обличчя зазвичай використовуються у розпізнаванні, Yin та ін. [23] в Бінгемтонському університеті запропонували як альтернативу базу даних тривимірної міміки з коментарями, а саме BU-3DFE 3D. Вона була розроблена для дослідження 3D-облич і міміки людини, а також для розвитку загального розуміння поведінки людини. База даних містить загалом фото облич 100 осіб, 56 жінок і 44 чоловіків, що демонструють шість емоцій. У базі даних є 25 моделей 3D обличчя з певною емоцією для кожної людини, а також набір із 83 орієнтирів обличчя, зроблених вручну для кожної моделі в базі. Оригінальний розмір кожного зображення обличчя – 1040 × 1329 пікселів;

— Японська жіноча міміка (JAFFE) [24]: База даних JAFFE містить 213 зображень семи емоцій обличчя (шість основних емоцій обличчя та одна нейтральна), представлених десятьма різними японськими моделями. Оригінальний розмір кожного зображення обличчя - 256 × 256 пікселів;

— Розширене обличчя Yale B (B+) [25]: Ця база даних складається із 16128 зображень обличчя, зроблених під одним джерелом освітлення. База містить обличчя 28 різних людей, загалом 576 різних фото, включаючи дев’ять поз для кожної з 64 умов освітлення. Оригінальний розмір кожного зображення обличчя - 320 × 243 пікселів;

— MMI [26]: MMI складається з понад 2900 відеопослідовностей та нерухомих зображень із високою роздільною здатністю, загалом 75 людей. База містить коментарі щодо наявність одиниць дії у відеопослідовності (кодування подій) та частково коментується на рівні кожного кадру, вказуючи чи знаходиться одиниця дії у нейтральній, початковій, піковій фазі чи фазі зсуву. Він містить загалом 238 відеопослідовностей для 28 людей, як

чоловіків, так і жінок. Оригінальний розмір кожного зображення обличчя – 720 × 576 пікселів;

— Динамічна спонтанна 3D база Бінгемтон-Пітсбург (BP4D-Спонтанна) [27]: BP4D – це тривимірна база даних відео, що включає групу з 41 різних молодих людей (23 жінки та 18 чоловіків) зі спонтанною мімікою. Випробувані були у віці 18–29 років. Одинадцять з них – азіати, шестеро – афро-американці, четверо – іспанці та двадцять - американці. Риси обличчя відслідковувались у 2D та 3D вимірах, використовуючи як особистісний, так і загальний підходи. База даних сприяє вивченню 3D просторово-часових особливостей злегка помітної міміки для кращого розуміння зв'язку між емоцією та динамікою руху в одиниці дії обличчя, а також глибшого розуміння природних мімічних дій. Оригінальний розмір кожного зображення обличчя - 1040 × 1329 пікселів;

— Каролінська база даних емоційних облич (KDEF) [28]: Ця база даних містить 4900 зображень емоцій людини. База даних складається з 70 осіб, кожен із яких демонструє сім різних емоцій, сфотографованих з п’яти різних ракурсів. Оригінальний розмір кожного зображення обличчя – 562 × 762 пікселів.

Цікавою є також база даних MPI, що, на відміну від описаних вище баз даних, містить велику кількість різноманітних природних емоційних кадрів під час розмови. Припускається, що люди розуміють емоції, аналізуючи як розмову, так і емоції. Ця база даних складається з більш ніж 18800 зразків відеопослідовностей десяти жінок та дев'яти чоловіків, що демонструють різні вирази обличчя, записані з одної фронтальної та двох бічних сторін.

На рисунку 1.5 показані приклади дев'яти баз даних для FER з 2D та 3D зображеннями, а також відеопослідовностями.

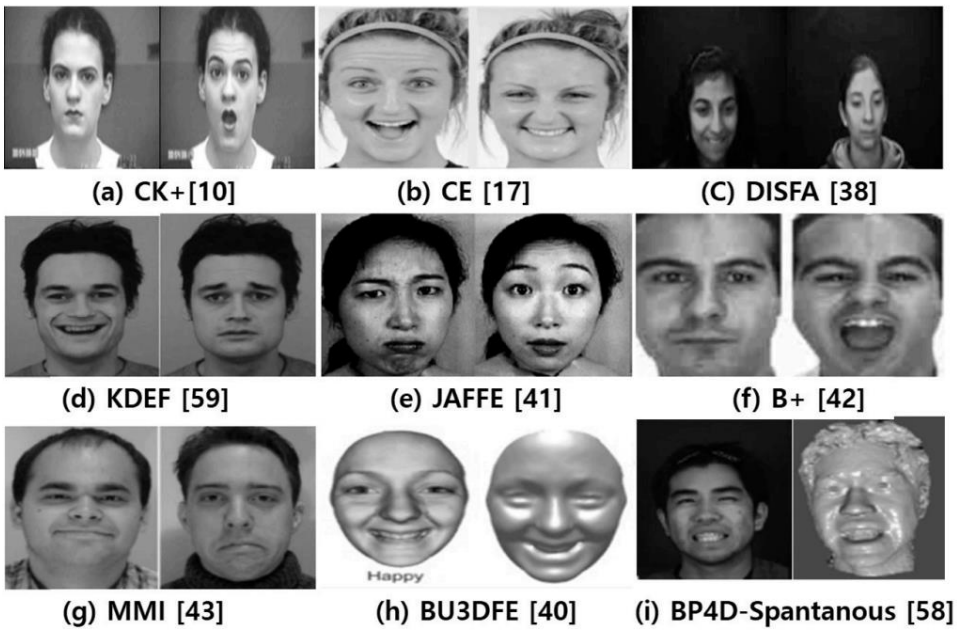

Рисунок 1.5 - Приклади дев'яти репрезентативних баз даних, пов'язаних з розпізнаванням емоцій за виразом обличчям. Бази даних (a) - (g) містять 2D статичні зображення та 2D відеопослідовності, а бази даних (h) - (i) містять послідовності у 3D та 3D-відео.

Останнім часом для досліджень емоцій людей використовують інфрачервоні датчики, такі як NIR-камера, теплова камера та датчики Kinect, оскільки зображення під видимим світлом легко змінюється, як тільки відбуваються зміни умови освітлення. База даних, що містить обличчя, отримані за допомогою камери NIR, Oulu-CASIA, NIR & VIS, складається з шести різних виразів обличчя 80 осіб віком від 23 до 58 років. 73,8% випробовуваних – чоловіки. Також база даних природного видимого та інфрачервоного зображення обличчя (USTC-NVIE) зібрала в собі одночасно спонтанні та поставлені кадри більш ніж 100 осіб, що були зроблені із використанням звичайної та інфрачервоної теплової камери. Ще одна база - база даних міміки та емоцій (FEEDB) - записана за допомогою сенсора Microsoft Kinect. Вона містить 1650 записів 50 осіб, що позують, демонструючи 33 різні емоції.

Як описано вище, для покращення якості розпізнавання емоцій за виразом обличчям окрім простої камери використовуються різноманітні датчики. Але існує певна межа покращення характеристик розпізнавання за умови використання лише якогось одного датчика. Спроби ще більше покращити результати розпізнавання, застосовуючи одразу декілька датчиків, тривають і сьогодні.

1.5 Оцінка ефективності розпізнавання емоцій за обличчям

Для оцінювання ефективності роботи та порівняння точності усіх можливих алгоритмів розпізнавання емоцій, метрики оцінювання підходів розпізнавання мають вирішальне значення, оскільки вони забезпечують стандарт кількісного порівняння. У цьому підрозділі подано короткий огляд основних метрик оцінювання та порівняння з результатами еталону.

1.5.1 Підходи до тестування моделей

Існують два різних типи проведення експерименту для підрахунку метрик: незалежно від особи та незалежно від бази даних [18]. У першому випадку кожна база даних розбивається на підвибірки для навчання та перевірки незалежно від людини, якій належить фото. Така метрика також називається K-кратною перехресною перевіркою. Мета K-кратної перехресної перевірки – уникнути таких проблем, як перенавчання, та зрозуміти наскільки добре модель зможе справлятися з завданням незалежно від вхідних даних [18]. За методом K-кратної перехресної перевірки кожен набір даних розбивається на K однакових сетів з унікальними особами. Потім

модель ітеративно навчається, використовуючи K-1 сет даних, і оцінюється на частині, що залишилася, до тих пір, поки всі особи не будуть перевірені. Перевірка проводиться з використанням менше 20% даних для самого тесту. Точність оцінюється шляхом усереднення коефіцієнта розпізнавання за K сетами. Наприклад, у десятикратній перехресній перевірці, прийнятій для оцінки, дев'ять сетів використовуються для навчання, а один – для тестування. Після того, як цей процес проводиться десять разів, точність десяти результатів усереднюється і визначається як ефективність класифікатора.

Другий підхід - це завдання, що не залежать від бази даних. У цьому завданні одна база даних використовується для тестування моделі, а решта баз даних, перелічених у таблиці 2, використовуються для навчання моделі. Модель ітеративно навчається з використанням K-1 баз даних по черзі і оцінюється на решті, поки не будуть протестовані всі бази. Точність оцінюється шляхом усереднення ефективності розпізнавання для K баз даних. Таким чином, метод аналогічний K-кратній перехресній перевірці.

1.5.2 Метрики оцінювання якості моделей

Метрики оцінювання розпізнавання емоцій за обличчям класифікуються на чотири методи, що використовують різні атрибути: точність, відклик, правильність та F1-оцінка.

Точність визначається як:

$$P = \frac{TP}{TP + FP} \tag{1}$$

де TP – кількість позитивних правильних відповідей, FP - кількість помилкових позитивних результатів.

Відклик визначається як:

$$R = \frac{TP}{TP + FN} \tag{2}$$

де FN - кількість негативних неправильних відповідей.

Точність - це частка автоматичного розпізнавання емоції i серед позитивних відповідей. Відклик - це кількість правильних розпізнавань емоції i над фактичною кількістю зображень із емоцією i [18]. Правильність - це відношення правильних результатів (істинних позитивних і справжніх негативних) до загальної кількості розглянутих випадків.

$$\text{Правильність} = \frac{TP + TN}{\text{Загальна кількіть розпізнавань}} \tag{3}$$

Інший показник, F1-оцінка, поділяється на дві метрики залежно від того, використовуються просторові або часові дані: F1-оцінка на основі кадру та F1-оцінка на основі події. Кожна із цих метрик фіксує різні характеристики результатів. Це означає, що F-оцінка на основі кадру показує можливості з точки зору просторової узгодженості, тоді як F-оцінка на основі подій демонструє можливості з точки зору узгодженості у часі [18]. F-оцінка на основі кадру визначається як

$$F1 - \text{кадр} = \frac{2RP}{R + P} \tag{4}$$

F1-оцінка на основі подій використовується для вимірювання ефективності розпізнавання емоцій на рівні сегменту, оскільки емоції розглядаються як часовий сигнал.

$$F1 - \text{подія} = \frac{2ER \times EP}{ER + EP} \quad (5)$$

де ER та EP - це відклик та точність на основі подій. ER - відношення правильно виявлених подій до усіх справжніх подій, тоді як EP - відношення правильно виявлених подій до усіх виявлених подій.

1.5.3 Порівняльний аналіз якості моделей та підходів до розпізнавання емоцій

Щоб показати порівняння між звичайними підходами, заснованими на знайдених різними методами ознаках, та підходами, що ґрунтуються на глибокому навчанні, у цьому розділі наведено загальнодоступні результати бази даних MMI. У таблиці 1.3 показано коефіцієнт розпізнавання шести звичайних підходів та шести підходів, заснованих на глибокому навчанні.

Таблиця 1.3 – Результати розпізнавання для бази даних MMI, адаптовані з [11].

Тип	Короткий опис основних алгоритмів	Вхідні дані	Точність (%)
Класичні підходи до розпізнавання емоцій за обличчям	Класифікатор розрідженого представлення з функціями LBP [11]	Статичний кадр	59.18
	Класифікатор розрідженого представлення з локальними фазами квантування [12]	Статичний кадр	62.72
	SVM з вейвлетами Габор [13]	Статичний кадр	61.89
	Класифікатор розрідженого представлення з LBP з трьох ортогональних площин [21]	Послідовність кадрів	61.19
	Класифікатор розрідженого представлення з функцією локальної фази квантування з трьох ортогональних площин [22]	Послідовність кадрів	64.11

Кінець таблиці 1.3

Тип	Короткий опис основних алгоритмів	Вхідні дані	Точність (%)
	Спільне представлення експресії CER [22]	Статичний кадр	70.12
	Середнє		63.20
Підходи до розпізнавання емоцій за обличчям на основі глибокого навчання	Глибоке навчання змінним частинам обличчя [23]	Послідовність кадрів	63.40
	Спільна тонка настройка в глибоких нейронних мережах [15]	Послідовність кадрів	70.24
	Глибокі мережі з застосуванням одиниць дії [16]	Статичний кадр	69.88
	Глибокі мережі на основі одиниць дії [17]	Статичний кадр	75.85
	Більш глибока CNN [27]	Статичний кадр	77.90
	CNN + LSTM з просторово-часовим представленням функцій [15]	Послідовність кадрів	78.61
	Середнє		72.65

Як показано в таблиці 1.3, підходи, засновані на глибокому навчанні, перевершують звичайні підходи в середньому 72,65% проти 63,2%. У класичних підходах до розпізнавання емоцій за обличчям спільне представлення експресії CER [22] має найбільшу продуктивність серед усіх

алгоритмів. У цьому дослідженні спробували обчислити різницю інформації між піковою експресією обличчя та її різними формами, що індивідуальні для кожного окремого обличчя, щоб зменшити ефект від індивідуальних особливостей обличчя на вилучення ознак. Оскільки виявлення ознак є стійким до пози обличчя та варіацій його форми, це дослідження має більш якісні показники розпізнавання, ніж інші класичні методи.

Серед підходів, що ґрунтуються на глибокому навчанні, два мають порівняно більшу ефективність порівняно з найсучаснішими методами. Це складна мережа CNN, запропонована в роботі [12], що складається з двох згорткових шарів, за кожним з яких йде максимальне об'єднання та чотири початкових шари. Ця мережа має однокомпонентну архітектуру, яка приймає зображення обличчя як вхідні дані та дає відповідь щодо належності міміки обличчя до однієї з семи базових емоцій. Підхід, що має найкращі показники точності [13] також складається з двох частин. У першій частині за допомогою CNN визначаються просторові ознаки ключових кадрів. У другій частині за допомогою LSTM визначаються також часові ознаки уже знайдених у першій частині просторових ознак. Виходячи з більшої точності комплексного гібридного підходу, що використовує просторово-часові ознаки обличчя, на ефективність розпізнавання емоцій за виразом обличчя значною мірою впливають не лише просторові зміни, але й часові.

Хоча підходи для розпізнавання емоцій за виразом обличчя, які засновані на глибокому навчанні, досягли значного успіху, що видно з експериментальних оцінок, усе ще залишається ряд питань, які заслуговують на подальше дослідження:

— Для навчання потрібні масштабний набір даних (у багатьох дослідженнях за оптимальну кількість приймають близько 30000 зображень) та величезна обчислювальна потужність (для мінімального результату потрібні професійні відео карти, наприклад такі, як лінійка Nvidia Tesla, що

має 12Gb оперативної пам’яті), оскільки для подальшого покращення точності розпізнавання структура моделей стає все більш глибокою;

— Необхідна велика кількість наборів даних, що зазвичай збираються вручну;

— Для навчання потрібні значні об’єми пам’яті, а також багато часу. Також вимоги до обчислювальної техніки та оперативної пам’яті при розпізнаванні роблять глибоке навчання непридатним для використання на мобільних платформах з обмеженими ресурсами [73];

— Для вибору відповідних гіперпараметрів, таких як швидкість навчання, розмір ядра конволюційних фільтрів та кількість шарів мережі, потрібні значні навички та досвід. Ці гіперпараметри також мають внутрішні залежності, що робить їх особливо «дорогими» для настройки.

Висновки

У цьому розділі було представлено короткий огляд підходів для розпізнавання емоцій за виразом обличчя. Як було описано вище, такі підходи можна розділити на два основних типи: класичні підходи та глибоке навчання. Класичні підходи складаються з трьох етапів: виявлення обличчя та лицьових компонентів, вилучення особливостей та класифікація виразів. Алгоритми класифікації, що застосовуються у класичному підході до розпізнавання емоцій, включають SVM, Adaboost та випадковий ліс. У свою чергу, підходи, що засновані на глибокому навчанні, значно зменшують залежність результату від фізичних особливостей обличчя та інших методів попередньої обробки, дозволяючи навчатися «від початку до кінця» безпосередньо з вхідних зображень. Особливим типом глибокого навчання є CNN, що демонструє високу спроможність нейронних мереж навчатись виявленню емоцій, використовуючи для цього різні типи даних. Однак,

оскільки методи розпізнавання емоцій за виразом обличчя на основі CNN не можуть відображати часові зміни в компонентах обличчя, були запропоновані гібридні підходи шляхом комбінування CNN для просторових ознак окремих кадрів та LSTM для часових ознак послідовних кадрів. В останніх дослідження гібридної архітектури CNN-LSTM було продемонстровано, що така зв’язка може перевершити результати раніше застосованих підходів CNN, якщо використовувати часове усереднення для агрегації. Однак підходи, засновані на глибокому навчанні, все ще мають ряд обмежень, включаючи необхідність великих наборів даних, великої обчислювальної потужності та великого обсягу пам’яті, як і багато часу як для етапів навчання, так і для тестування. Більше того, хоч гібридна архітектура демонструє високу ефективність, мікроемоції залишаються складною задачею для будь-якого підходу, оскільки вони є проявом більш спонтанної та тонкої міміки обличчя, яка відбувається мимоволі.

У цьому розділі також коротко представлені деякі популярні бази даних, пов'язані з розпізнаванням емоцій за виразом обличчя, що містять як відеопослідовності, так і з нерухомі зображення. Традиційно міміку людини вивчали, використовуючи або статичні 2D зображення, або 2D відеопослідовності. Однак, оскільки при аналізі 2D зображень виникають труднощі з обробкою варіацій в позі та тонких рухів обличчя, нові бази даних пропонують для вивчення 3D моделі обличчя для покращення обробки мікроелементів емоцій.

Крім того, були представлені основні методи оцінювання якості та точності підходів для забезпечення стандартних показників для порівняння. Методи оцінки в основному представлені у сфері якості розпізнавання саме емоцій, і в основному використовують точність та відгук. Однак слід запропонувати нові метод оцінювання, що оцінювали б розпізнавання послідовної міміки або розпізнавання мікроемоцій для рухомих зображень.

Незважаючи на те, що дослідження з розпізнавання емоцій за обличчям проводилися протягом останнього десятиліття, за останні роки ефективність розпізнавання значно покращилась за допомогою комбінації алгоритмів глибокого навчання. Оскільки розпізнавання емоцій є важливим способом навчити емоціям машини, вигідно проводити різні дослідження щодо їх майбутнього застосування. Якщо в майбутньому алгоритми глибокого навчання, орієнтовані на емоції, зможуть бути поєднані з додатковими датчиками таких систем, як «інтернет речей», можна допустити, що розпізнавання емоцій за виразом обличчя зможе досягти нових рівнів якості розпізнавання, включаючи навіть розпізнавання спонтанних мікроемоцій такого ж рівня, як і люди.

РОЗДІЛ 2. РОЗПІЗНАВАННЯ ЕМОЦІЙ ПО ОБЛИЧЧЮ ЗА ДОПОМОГОЮ АКТИВНОЇ МОДЕЛІ ФОРМИ

Запропонований підхід загалом містить два завдання: перше – знаходження ознак обличчя за допомогою активної моделі форми та підрахунок геометричних ознак, а друге – класифікація отриманих геометричних ознак за допомогою LSTM для визначення емоції обличчя на фото. Спочатку система знаходить 68 ключових точок обличчя. Потім, за допомогою тріангуляції Делоне для множини точок на площині, обличчя розбивається на трикутники. Згодом підраховуються довжини сторін усіх трикутників, дані нормуються. Наприкінці, розпізнавання емоції виконується на основі класифікації за допомогою рекурентної нейронної мережі LSTM.

2.1 Виявлення обличчя

Для виявлення зображень обличчя пропонується використовувати зв’язку гістограми напрямлених градієнтів та лінійного методу опорних векторів.

Ідея гістограм напрямлених градієнтів полягає у тому, що замість того, щоб використовувати напрямок градієнту кожного пікселя окремо, ми групуємо їх у комірки. Для кожної комірки обчислюються всі напрямки градієнтів, а потім усі напрямки у комірці підсумовуються. Чим більший вектор, тим більшу вагу він має для комірки, а випадкові маленькі напрямки ігноруються. Тобто ця гістограма дає домінуючу орієнтацію комірки. А усі комірки разом – уявлення про структуру зображення загалом. Гістограма зберігає загальну подачу об’єкта, але дає при цьому можливість варіативності.

Лінійний метод опорних векторів є неймовірнісним бінарним лінійним класифікатором. Модель є представленням зразків як точок у просторі, відображених таким чином, що зразки з окремих категорій розділені прогалиною, яка є якнайширшою. Нові зразки також відображаються у цьому просторі, і на основі цього робиться передбачення про їхню належність до категорії на основі того, у який бік прогалини вони потрапляють.

Відповідно, щоб знайти обличчя на фото, спочатку алгоритм знаходить гістограми напрямлених градієнтів. Потім, застосовуючи ковзке вікно, проходить по клітинам гістограм та перевіряє за допомогою лінійного методу опорних векторів чи наявне в цьому вікні обличчя. Наприкінці алгоритм вибирає з усіх знайдених позитивних вікон найкраще.

2.2 Попередня обробка зображення

Для попередньої обробки зображення пропонується використати два методи: модель активної форми та тріангуляцію Делоне.

Модель активної форми – це статична модель, що може як завгодно деформуватись щоб відповідати об’єкту на новому зображенні. Форма складається з певної кількості точок, кожна з яких відповідає за своє місце на об’єкті, у нашому випадку – на обличчі людини. На рисунку 2.1 зображена стандартна початкова модель активної форми обличчя людини, яка містить у собі 68 точок.

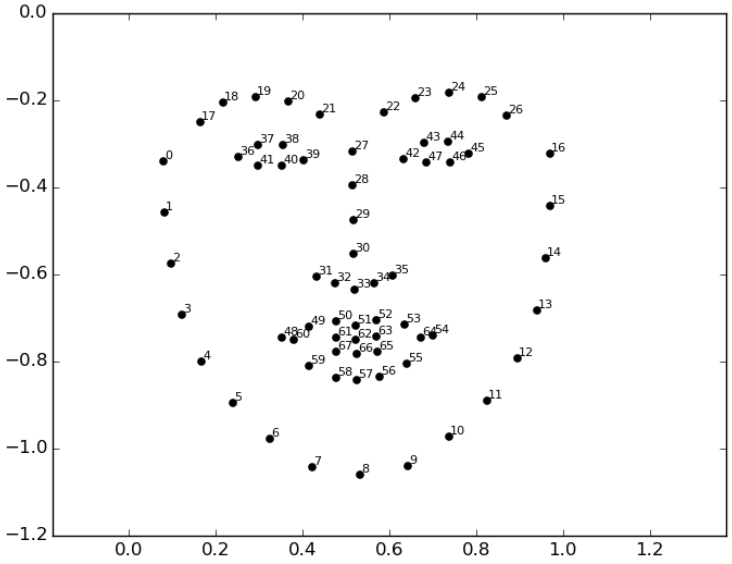

Рисунок 2.1 – Модель активної форми на 68 точок [9]

Для того, щоб знаходити точки моделі на новому обличчі, потрібно навчити модель пошуку найкращих позицій та співставлення з еталоном. Для пошуку точок застосовується відстань Махаланобіса, відстань у евклідовому просторі, що узагальнює поняття евклідової відстані. Визначається формулою:

$$d(X,Y;S) = \sqrt{(X-Y)^T S^{-1}(X-Y)} \tag{6}$$

де $X = (x_1 \dots x_N)^T$ – багатомірний вектор, від якого береться відстань;

$Y = (y_1 \dots y_N)^T$ – середні значення множини, до якої рахується відстань;

S – матриця коваріацій множини.

На основі моделі активної форми будується тріангуляція Делоне для множини точок на площині. Це така тріангуляція, що жодна точка множини не знаходиться всередині описаних довкола трикутників кіл в цій множині. Таке розміщення дозволяє максимально зменшити кількість маленьких кутів

трикутників. Тріангуляція Делоне для двовимірного простору зображена на рисунку 2.2.

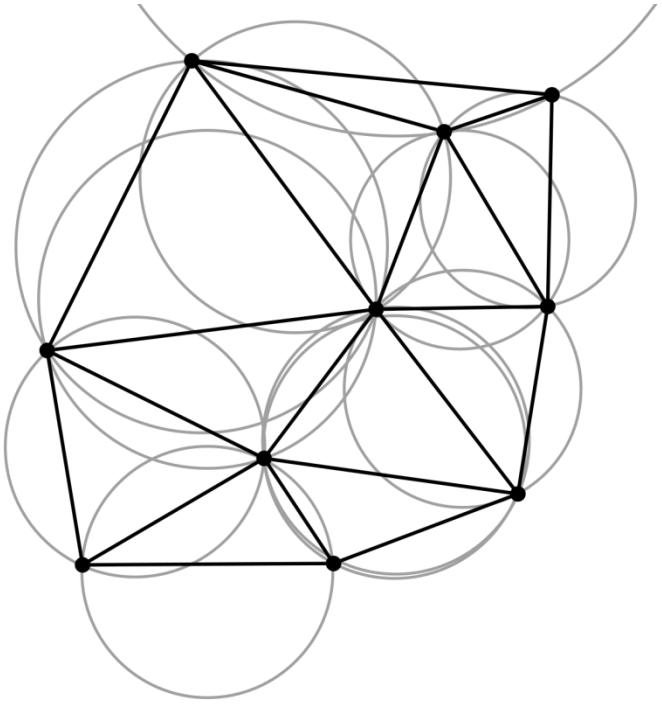

Рисунок 2.2 – Тріангуляція Делоне для двовимірного простору [9]

У загальному випадку розв’язання задачі тріангуляції Делоне відповідає дуальному графу розбиття Вороного. Діаграма Вороного – це особливий вид розбиття метричного простору, що визначається відстанями до заданої дискретної множини ізольованих точок цього простору. Простіше кажучи, у кожному секторі розбиття є одна і тільки одна ізольовано точка площини, а точки самої діаграми лежать до вершини всередині сектора ближче, ніж будь-яка інша ізольована точка. Діаграма Вороного для двовимірного простору зображена на рисунку 2.3.

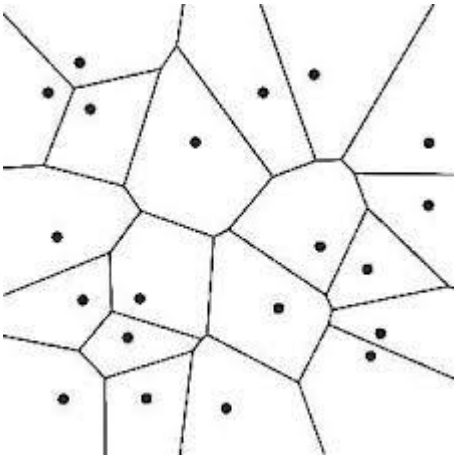

Рисунок 2.3 – Діаграма Вороного для двовимірного простору [10]

Для побудови такої діаграми застосовують алгоритм Форчуна. Алгоритм Форчуна заснований на застосуванні замітаючої прямої. Замітаюча пряма – це такий допоміжний об’єкт, що являє собою вертикальну пряму лінію. Лінія рухається зліва направо, залишаючи зліва точки, на основі яких будується діаграма Вороного. Межа між областю Вороного, прямою та областями точок складається з відрізків парабол, оскільки парабола – це геометричне місце точок, що рівновіддалене від заданої точки. Щоразу коли пряма проходить через чергову точку, ця точка додається до вже побудованої ділянки діаграми. Обчислювальна складність усього алгоритму дорівнює $O(n \log n)$. На рисунку 2.4 зображений алгоритм Форчуна для двовимірного простору.

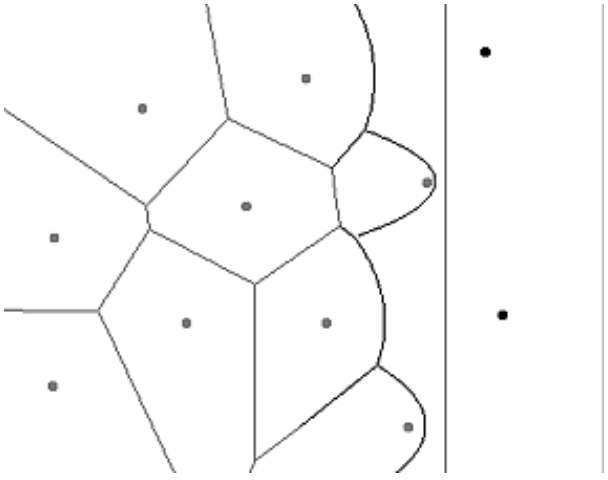

Рисунок 2.4 – Алгоритм Форчуна для двовимірного простору [10]

2.3 Підготовка даних для розпізнавання емоцій та навчання

Для подальшого використання даних, застосовуючи глибоке навчання, спочатку потрібно підготувати дані на вхід системи. Для цього пропонується наступний підхід: для кожної сторони визначених раніше трикутників на обличчі знаходиться довжина ребра. Потім усі довжини нормуються за формулою:

$$z = \frac{x - \min(x)}{\max(x) - \min(x)} \tag{7}$$

Для того, щоб мати однакові стандартизовані точки на кожному обличчі датасету, і як результат, дані, що можна порівнювати між собою, тріангуляція застосовувалась тільки до першого обличчя сету. На усіх інших обличчях трикутники визначались згідно заданому першому зразку.

В результаті було отримано вектори нормованих даних однакової довжини, що трактувались як агреговані ознаки обличчя на кожному фото.

2.4 Класифікація даних за допомогою LSTM

LSTM (long short-term memory) – це архітектура рекурентної нейронної мережі, що була запропонована Зеллом Хохрайтером та Юргеном Шмідгубером у 1997 році. Як і рекурентні нейронні мережі, вона є універсальною у тому сенсі, що за достатньої кількості вузлів мережі та правильних коефіцієнтів вона може обчислити будь-що, що може виконати комп’ютер. Мережа LSTM добре підходить для розв’язання задач

класифікації, обробки або передбачення часових рядів, якщо між важливими подіями існують часові затримки невідомої наперед довжини. Саме ця нечутливість до довжини прогалин дає перевагу LSTM над іншими нейронними мережами у ряді задач.

Головна особливість LSTM – наявність пам’яті про попередні етапи розпізнавання. Рекурентні нейронні мережі мають зворотній зв’язок. Наявність зворотного зв’язку дозволяє передавати інформацію від одного шару мережі до іншого. Рекурентну нейронну мережу можна розглядати як декілька копій однієї мережі, кожна з яких передає інформацію наступній копії. Розгортка рекурентної нейронної мережі зображена на рисунку 2.5.

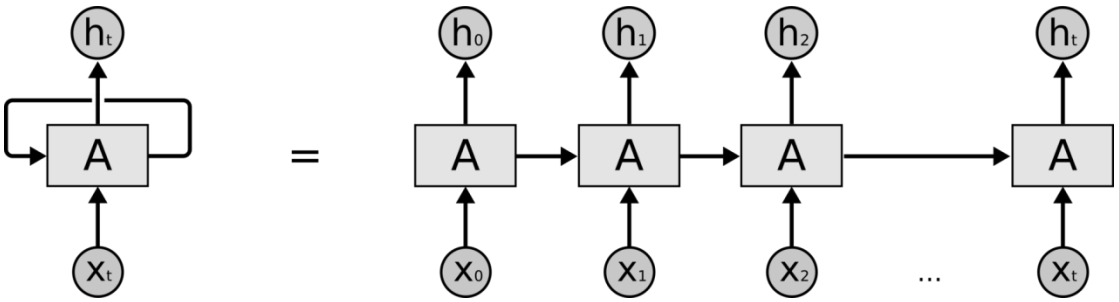

Рисунок 2.5 – Розгортка рекурентної нейронної мережі [15]

Особливість пам’яті звичайної рекурентної мережі є її коротка тривалість. Так, мережа може пов’язувати та ефективно використовувати інформацію, що знаходиться відносно недалеко одна від одної у часі. Якщо ж вимагається зв’язати ітерації, що досить віддалені кількістю кроків одна від одної, модель вже не зможе цього зробити. Така мережа має досить коротку пам’ять. Хоч в теорії проблем з довжиною затримки не має виникати, на практиці навчити рекурентну мережу такій кореляції не є можливим. LSTM була створена спеціально для того, щоб боротися з проблемою довготривалих залежностей.

Структура LSTM, як і структура звичайних рекурентних мереж, нагадує ланцюг з модулів мережі, що повторюються. У LSTM один модуль містить одразу чотири шари, що певним чином взаємодіють між собою.

Ключовий компонент – це стан комірки – горизонтальна лінія, що проходить у верхній частині, якщо дивитися на схему модуля LSTM (рисунок 2.6).

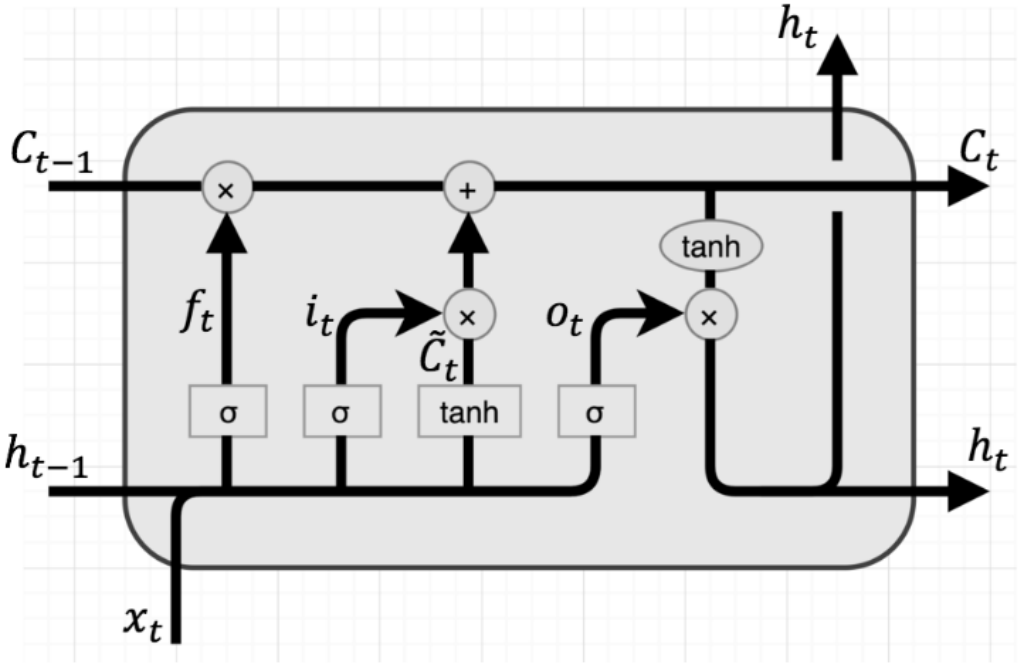

Рисунок 2.6 – Схема типового модуля LSTM [20]

Перший крок роботи LSTM – визначити, яку інформацію можна забути як неважливу. Це рішення приймає сігмоїдальний шар, що називається шаром фільтру забуття. Він дивиться на стан попереднього модуля h_{t-1} та початковий стан комірки x_t та повертає число від 0 (повністю забути) до 1 (повністю зберегти) для кожної комірки C_{t-1}. Обчислюється за загальною формулою, що має вигляд:

$$f_t = \sigma\left(W_f[h_{t-1}, x_t] + b_f\right) \tag{8}$$

Наступний крок – визначити, яка нова інформація має зберігатися в кожній комірці. Спочатку сигмоїдальний шар, що називається шаром вхідного фільтру, визначає, які значення потрібно перерахувати. Потім tanh-шар будує вектор нових значень-кандидатів $\widetilde{C_t}$, які можуть бути додані до стану комірки. Форму для цього виглядають наступним чином:

$$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i) \quad (9)$$

$$C_t = tahn(W_C[h_{t-1}, x_t] + b_C) \quad (10)$$

Старий стан комірки множиться на f_t, щоб забути непотрібне. Потім додається $i_t \times \widetilde{C_t}$. Тобто разом формула виглядає наступним чином:

$$C_t = f_t \times C_{t-1} + i_t \times \widetilde{C_t} \quad (11)$$

Наприкінці новий стан комірки приводиться до вигляду, який потрібен на виході з модуля. Спочатку застосовується сигмоїдальний шар, що вирішує яка інформація буде виведена як результат. Потім значення пропускається через tanh-шар, щоб на виході отримати значення в діапазоні від -1 до 1, та множаться на результат роботи сигмоїдального шару. Формули виглядають наступним чином:

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \quad (12)$$

$$h_t = o_t \times tahn(C_t) \quad (13)$$

Висновки

У цьому розділі було викладено алгоритм, що пропонується у роботі для розпізнавання базових емоцій за обличчям людини. Також були наведені описи методів для окремих складових алгоритму. Детально розібрана та проаналізована рекурентна нейронна мережа та її застосування. Було доведено доцільність її використання у роботі, оскільки вона суттєво легша, ніж звичайна нейронна мережа, а також має пам’ять, тобто може зв’язувати різні ознаки, навіть якщо вони знаходяться не поряд.

Також було викладено методи для обробки фото, точніше методи знаходження обличчя на фото та його подальшої тріангуляції. Знаходити обличчя на фото пропонується методом гістограм, а далі – знайти ознаки обличчя та з їх допомогою виконати тріангуляцію.

Запропоновано власний підхід до подальшої обробки інформації – знаходження довжин усіх трикутників та подальша нормалізація вектора довжин. Отриманий вектор можна трактувати як однозначне представлення обличчя та його головних ознак, на основі яких рекурентна нейронна мережа зможе класифікувати емоцію.

РОЗДІЛ 3. ПРАКТИЧНІ РЕЗУЛЬТАТИ

Застосовуючи методи, описані у розділі два, була розроблена програма, що за фото обличчя розпізнає емоцію із семи можливих: нейтральність, щастя, сум, здивування, злість, відраза та страх.

3.1 Бази даних, що використовувались у експерименті

Для навчання та перевірки розробленої моделі булло застосовано багато баз даних з постановочними фото. На усіх фото люди стояли прямо перед камерою та демонстрували пік усіх семи емоцій. Усі фото були розподілені на сім папок з відповідними емоціям назвами.

KDEF [28] – база даних з постановочними фото, що містить фото семи емоцій від 70 різних людей з п’яти ракурсів, у тому числі й фронтальні фото. Усього було використано 490 фото. Приклади фото з цієї бази даних на рисунку 3.1.

Рисунок 3.1 – Приклади фото з бази даних KDEF [28]

FacesDB [29] – також база даних з постановочними фото, що містить фронтальні фото облич для семи емоцій, а також гримас. Усього 38 різних людей. З цієї бази даних було використано 266 фото. Приклади фото з бази даних продемонстровано на рисунку 3.2.

Рисунок 3.2 – Приклади фото з бази даних FacesDB [29]

CK+ [10] – база даних з постановочними відео послідовностями, що відображають зміну емоції обличчя з нейтральної на одну з інших семи. Для цього експерименту були вибрані тільки останні пікові кадри. У базі даних наявні обличчя 123 людей, від 1 до 6 емоцій для кожної моделі. Приклади фото з бази даних продемонстровано на рисунку 3.3.

Рисунок 3.3 – Приклади фото з бази даних CK+ [10]

JAFFE [24] – база даних з постановочними фото японських жінок. Усього 10 моделей, по 7 емоцій для кожної жінки. Для експерименту були використані 70 фото. Приклади фото продемонстровані на рисунку 3.3.

Рисунок 3.3 – Приклади фото з бази даних JAFFE [24]

У сумі усі бази даних надали 1792 фото для обробки. З них 1434 були використані як дані для навчання моделі, а інші 358 фото – для перевірки правильності роботи програми.

3.2 Імплементація алгоритму

Для імплементації проекту було вибрано мову програмування python. Це мова об’єктно орієнтованого програмування, для якої написано багато бібліотек, що призначені для статистики, глибокого навчання та швидкого прототипування.

Так як для розпізнавання було вирішено використовувати LSTM, перший крок роботи – обробка даних та їх підготовка до використання нейронною мережею.

Для визначення обличчя на фото була використана бібліотека методів для обробки даних на dlib. Її метод знаходження обличчя та виділення його у квадрат використовує зв’язку гістограми напрямлених градієнтів та лінійного методу опорних векторів.

Також можливості бібліотеки dlib були використані для знаходження тріангуляції обличчя. Перше фото, що було у списку фото для обробки, використовувалось як прототип для усіх інших фото, щоб тріангуляція усіх облич співпадала та її можна було порівнювати.

Останнім пунктом були знаходження довжини сторін кожного трикутника та нормалізація усіх довжин, сформованих у вектор.

Ці три кроки обробки реалізовані у файлі «length.py», код програми викладений у додатку А.

У файлі «dataset.py» збирається список усіх файлів, що наявні для використання у папці _Dataset та проводиться обробка усіх фото одне за одним. Вектори для кожного фото зберігаються у файлі «geometry.txt». Також створюється файл «geometry_answers.txt», куди записуються відповіді до кожного відповідного фото – бінарні вектори розміром в 7 емоцій. Код цієї програми знаходиться у додатку Б.

Навчання моделі реалізовано у файлі «lstm.py». Скрипт може працювати у трьох режимах:

1) lstm.py -model=None

У цьому випадку програма прочитає файли даних, поділить їх на навчальні та тестові у співвідношенні 80:20 та навчить наперед визначену модель нейронної мережі за 50 епох. Навчена модель буде збережена у файл «model.hdf5»;

2) lstm.py -model=’model.hdf5’ -check_only=True

У цьому разі уже існуюча модель буде перевірена на даних з файлу. При такому виклику необхідно зазначити шлях до файлу моделі;

3) lstm.py -model=’model.hdf5’ -img=’img3.png’

Такий виклик просто розпізнає емоцію на фото за допомогою наданої моделі та виведе результат. Необхідно обов’язково вказати шлях до файлу моделі.

Код файлу «lstm.py» можна знайти у додатку В.

3.3 Результати навчання

В результаті на наданому датасеті вдалось досягнути 72% точності розпізнавання емоцій. Це стало можливим завдяки агрегації ознак обличчя з застосуванням тріангуляції. Адже якщо використовувати як вхідні дані виключно координати точок обличчя, максимальним результатом буде тільки 62% точності розпізнавання. Точність розпізнавання для кожної з емоцій вказана в таблиці 3.1.

Таблиця 3.1 – Точність розпізнавання для кожної з емоцій

Емоція	Кількість фото в вибірці	Кількість правильно розпізнаних фото	Кількість неправильно розпізнаних фото	Процент
Нейтральність	49	39	10	80%
Задоволення	49	47	2	96%
Сум	49	31	18	63%
Здивування	48	44	4	90%
Злість	49	16	33	33%
Відраза	49	40	9	82%
Страх	49	30	19	61%

З таблиці видно, що найкраще вдалося розпізнати задоволення та здивування, найгірші показники – у злості. Нижче, у таблиці 3.2, представлені детальні результати точності роботи моделі у вигляді матриці помилок.

Таблиця 3.2 – Матриця помилок програми з застосуванням LSTM

Задано / Результат	Нейтральність	Задоволення	Сум	Здивування	Злість	Відраза	Страх
Нейтральність	39	0	3	0	1	4	2
Задоволення	1	47	1	0	0	0	0
Сум	8	0	31	1	2	1	6
Здивування	0	0	0	45	0	0	4
Злість	6	0	3	0	15	24	0
Відраза	0	3	2	0	0	40	4
Страх	3	2	4	8	0	2	30

Для порівняння, на цих же даних були навчені класичні алгоритми класифікації, такі як AdaBoost та випадковий ліс. Загальна якість розпізнавання – 60,5% та 71% відповідно. У таблицях 3.3 та 3.4 наведені матриці помилок для цих методів.

Код програми, що навчає ці класифікатори, знаходиться у файлі «adaboost.py».

Таблиця 3.3 – Матриця помилок програми з застосуванням AdaBoost

Задано / Результат	Нейтральність	Задоволення	Сум	Здивування	Злість	Відраза	Страх
Нейтральність	32	0	7	0	5	0	5
Задоволення	0	34	0	0	0	6	9
Сум	16	0	24	0	3	0	6
Здивування	0	0	0	40	0	0	9
Злість	3	0	11	0	14	20	0
Відраза	5	1	2	0	6	31	4
Страх	2	1	5	5	2	2	32

Таблиця 3.4 – Матриця помилок програми з застосуванням випадкового лісу

Задано / Результат	Нейтральність	Задоволення	Сум	Здивування	Злість	Відраза	Страх
Нейтральність	38	0	5	1	1	2	2
Задоволення	1	45	0	0	0	3	0
Сум	6	0	34	2	3	1	3
Здивування	0	0	0	46	0	0	3
Злість	8	0	4	0	20	15	1
Відраза	2	3	1	0	1	38	4
Страх	4	2	1	12	1	3	26

Висновки

У цьому розділі було на практиці виконано описану у розділі 2 теорію щодо методу розпізнавання емоцій за обличчям людини. На мові програмування python було написано програму, що обробляє надані фото та навчає рекурентну нейронну мережу розпізнавати сім базових емоцій.

Був отриманий результат у 72% точності розпізнавання, що є середнім результатом для багатьох інших методів у різних наукових роботах, які було досліджено у розділі 1. Найкраще модель справляється з розпізнаванням добре виражених мімікою емоцій, такі як задоволення або здивування. Гірше – емоції, що люди демонструють більше очима, а за виглядом більше подібні до нейтральних, такі як злість та сум.

На відміну від багатьох результатів, що були представлені у розділі 1, у даному експерименті використовувалось до десяти разів менше даних для навчання, ніж у інших підходах. Проте завдяки підходу з тріангуляцією обличчя вдалося досягти таких самих результатів розпізнавання, при цьому для цього знадобилося набагато менше обчислювальних ресурсів як для навчання, так і для перевірки.

Недоліком моделі є її мала стійкість до зміни пози обличчя, оскільки зміна пози з прямої на поворот голови таким чином, щоб овал обличчя був не цілком на фото призводить до суттєвого падіння якості розпізнавання.

РОЗДІЛ 4. РОЗРОБЛЕННЯ СТАРТАП-ПРОЕКТУ

4.1 Опис ідеї проекту (товару, послуги, технології)

Опис ідеї стартап-проекту представлено в таблиці 4.1.

Таблиця 4.1 - Опис ідеї стартап-проекту

Зміст ідеї	Напрямки застосування	Вигоди для користувача
Розпізнавання емоцій дозволяє отримати більше інформації про внутрішній стан людини, що в свою чергу дає більше розуміння про відгук клієнта щодо продукту.	1. Робота з клієнтами магазину	Через камери в магазині адміністратор магазину зможе розуміти чи подобається клієнтам певний товар
	2. Оцінка роботи з клієнтами персоналу	Через камери в залах адміністратори зможуть оцінити наскільки клієнти задоволені роботою персоналу

Визначення сильних, слабких та нейтральних характеристик ідеї проекту представлено в таблиці 4.2.

Таблиця 4.2 - Сильні, слабкі та нейтральні характеристики ідеї проекту

№ п/п	Техніко-економічні характеристики ідеї	(Потенційні) товари/концепції конкурентів			W (слабка сторона)	N (нейтр. стор.)	S (сильна сторона)
		EmoDetect	Felena Soft	Microsoft Project Oxford			
1.	Можливість розпізнавання з відеоряду	+	-	-			+
2.	API	+	-	+	+		
3.	Відомість бренду	-	-	+	+		
4.	Офлайн розпізнавання	+	+	-		+	

4.2 Технологічний аудит ідеї проекту

Технологічна здійсненність ідеї проекту – таблиця 4.3.

Таблиця 4.3 - Технологічна здійсненність ідеї проекту

№ п/п	Ідея проекту	Технології її реалізації	Наявність технологій	Доступність технологій
1	Розпізнавання емоцій з відеоряду	OpenGL, DLib	Наявні. Необхідна розробка з використанням відповідних бібліотек.	Доступні
2	Отримання відеоряду з камери	FFmpeg	-/-	Доступні
Обрана технологія реалізації ідеї проекту: Проект буде реалізований на мові програмування python з використанням технології обробки фото OpenGl та бібліотеки для обробки обличчя DLib.				

4.2 Аналіз ринкових можливостей запуску стартап-проекту

Попередня характеристика потенційного ринку стартап-проекту представлена в таблиці 4.4.

Таблиця 4.4 - Попередня характеристика потенційного ринку стартап-проекту

№ п/п	Показники стану ринку (найменування)	Характеристика
1	Кількість головних гравців, од	3
2	Загальний обсяг продаж, грн/ум.од	1000
3	Динаміка ринку (якісна оцінка)	Зростає
№	Показники стану ринку (найменування)	Характеристика
4	Наявність обмежень для входу (вказати характер обмежень)	Немає
5	Специфічні вимоги до стандартизації та сертифікації	Немає
6	Середня норма рентабельності в галузі (або по ринку), %	15%

Характеристика потенційних клієнтів стартап-проекту представлена в таблиці 4.5.

Таблиця 4.5 - Характеристика потенційних клієнтів стартап-проекту

Потреба, що формує ринок	Цільова аудиторія (цільові сегменти ринку)	Відмінності у поведінці різних потенційних цільових груп клієнтів	Вимоги споживачів до товару
Отримання відгуку від клієнтів про товар	Торгові точки та заклади масового обслуговування	Поведінка користувачів програми фактично не буде відрізнятись	User-friendly інтерфейс; Швидкодія; Точність роботи; Можливість працювати без інтернет з’єднання
Отримання відгуку від клієнтів про якість обслуговування	Відділення роботи з клієнтами	Поведінка користувачів програми фактично не буде відрізнятись	User-friendly інтерфейс; Швидкодія; Точність роботи; Можливість працювати без інтернет з’єднання

Фактори загроз представлені в таблиці 4.6.

Таблиця 4.6 - Фактори загроз

№ п/п	Фактор	Зміст загрози	Можлива реакція компанії
1	Подібний функціонал з’явиться у більш відомих конкурентів	Люди будуть схильні довіряти більш відомим брендам	Активніша маркетингова кампанія

Фактори можливостей представлені в таблиці 4.7.

Таблиця 4.7 - Фактори можливостей

№ п/п	Фактор	Зміст можливості	Можлива реакція компанії
1	Гнучкі ціни	Збільшення доступності продукту	Введення підписки
2	Реалізація додатку на інших мобільних платформах	Збільшення можливої кількості споживачів	Введення полегшеної версії продукту для широких мас користувачів

Ступеневий аналіз конкуренції на ринку представлений в таблиці 4.8.

Таблиця 4.8 - Ступеневий аналіз конкуренції на ринку

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)
1. Вказати тип конкуренції - монополія/олігополія/ монополістична/чиста	Олігополія. Є невелика кількість популярних програм з подібним функціоналом	Необхідно, щоб у нас був особливий функціонал, який зможе зацікавити споживачів
2. За рівнем конкурентної боротьби - локальний/національний/…	Глобальна. На ринку присутні системи, розроблені за кордоном	Позитивно вплинути можна розширенням аудиторії за рахунок зниження цін та доступності технічного рішення
3. За галузевою ознакою - міжгалузева/ внутрішньогалузева	Міжгалузева	Робота з різними типами обслуговування, розширення кількості ситуацій, для яких система може бути застосована

Кінець таблиці 4.8

Особливості конкурентного середовища	В чому проявляється дана характеристика	Вплив на діяльність підприємства (можливі дії компанії, щоб бути конкурентоспроможною)
4. Конкуренція за видами товарів: - товарно-родова - товарно-видова - між бажаннями	Товарно - видова	Потрібно піднімати якість розпізнавання, вдаючись до різних алгоритмів, що будуть працювати точніше та швидше, ніж у конкурентів
5. За характером конкурентних переваг - цінова / нецінова	Нецінова. Перевагою в кожного виробника є особливості виконання розпізнавання	Розробка кращих алгоритмів.
6. За інтенсивністю - марочна/не марочна	Марочна	Торгова марка впливає на довіру клієнтів до якості продукту

Аналіз конкуренції в галузі за М. Портером – таблиця 4.9.

Таблиця 4.9 - Аналіз конкуренції в галузі за М. Портером

Складові аналізу	Прямі конкуренти в галузі	Потенційні конкуренти	Постачальники	Клієнти	Товари-замінники
	EmoDetect FelenaSoft Microsoft Project Oxford	Ціни, точність розпізнавання, розміри капіталовкладень	Змінні витрати постачальників, диференціація витрат	Контроль якості, система інформації	Ціна, лояльність споживачів
Висновки:	Інтенсивність боротьби з існуючими конкурентами достатньо висока. Але ніяких перешкод виходу на ринок немає	Виграти конкуренцію можна тільки за рахунок переваг у якості розпізнавання за рівних технічних умов	Постачальники не диктують умови роботи на ринку	Клієнти не диктують умови роботи на ринку	Обмеження для роботи на ринку через товари замінники

Обґрунтування факторів конкурентоспроможності наведено в таблиці 4.10.

Таблиця 4.10 - Обґрунтування факторів конкурентоспроможності

№ п/п	Фактор конкурентоспроможності	Обґрунтування (наведення чинників, що роблять фактор для порівняння конкурентних проектів значущим)
1	Інновації	Програмний продукт матиме функціонал, якого на даний момент немає в жодного з конкурентів
2	Якість	Висока якість розпізнавання, велика кількість допоміжних статичних даних
3	Підтримка і оновлення	Додаток буде постійно оновлюватись для додавання нового функціоналу, покращення якості розпізнавання та згідно з побажаннями користувачів.

Порівняльний аналіз сильних та слабких сторін наведено в таблиці 4.11.

Таблиця 4.11 - Порівняльний аналіз сильних та слабких сторін

№ п/п	Фактор конкурентоспроможності	Бали 1-20	Рейтинг товарів-конкурентів у порівнянні з Microsoft Project Oxford						
			–3	–2	–1	0	+1	+2	+3
1	Інновації	118					+		
2	Якість	115				+			
3	Підтримка і оновлення	18					+		

SWOT-аналіз стартап-проекту – таблиця 4.12.

Таблиця 4.12 - SWOT-аналіз стартап-проекту

Сильні сторони: висока якість прогнозу новий функціонал	Слабкі сторони: мала відомість немає клієнтської бази
Можливості: постійне покращення якості попит	Загрози: повільна робота продукту, конкуренція

Альтернативи ринкового впровадження стартап-проекту розглянуто в таблиці 4.13.

Таблиця 4.13 - Альтернативи ринкового впровадження стартап-проекту

№ п/п	Альтернатива (орієнтовний комплекс заходів) ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації
1	Швидкий вихід на ринок із «сирим» продуктом, можливі проблеми із точністю прогнозу та універсальністю	Низька	3-6 місяці
2	Поступовий вихід з готовим, відлагодженим продуктом. Висока якість та конкурентоспроможна ціна.	Висока	6-12 місяців

4.4 Розроблення ринкової стратегії проекту

Вибір цільових груп потенційних споживачів наведено в таблиці 4.14.

Таблиця 4.14 - Вибір цільових груп потенційних споживачів

№ п/п	Опис профілю цільової групи потенційних клієнтів	Готовність споживачі в сприйняти продукт	Орієнтовний попит в межах цільової групи (сегменту)	Інтенсивність конкуренції в сегменті	Простота входу в сегмент
1	Малі магазини та точки обслуговування клієнтів	Низька готовність (занадто дороге рішення)	Попит низький.	Висока, простіше взяти менш інноваційні рішення для визначення попиту на товари	Середня
2	Середні магазини та точки обслуговування клієнтів	Середня	Середній завдяки намаганням бути інноваційним, що дозволяють розміри компанії	Середня	Простий вхід, середні компанії не готові купувати дорогі продукти, тому погодяться на новинки

Кінець таблиці 4.14

№ п/п	Опис профілю цільової групи потенційних клієнтів	Готовність споживачів сприйняти продукт	Орієнтовний попит в межах цільової групи (сегменту)	Інтенсивність конкуренції в сегменті	Простота входу в сегмент
3	Великі магазини та точки обслуговування клієнтів	Висока	Високий (великі компанії намагаються привабити якомога більше клієнтів та досягти найбільших висот у пропозиціях, щоб конкурувати з такими ж великими компаніями)	Інтенсивність конкуренції помірна, є конкуренти з дорогими рішеннями та великими цінами	Вхід в сегмент складний
Які цільові групи обрано: 2, 3					

Визначення базової стратегії розвитку наведено в таблиці 4.15.

Таблиця 4.15 - Визначення базової стратегії розвитку

№ п/п	Обрана альтернатива розвитку проекту	Стратегія охоплення ринку	Ключові конкурентоспроможні позиції відповідно до обраної альтернативи	Базова стратегія розвитку*
1	2 та 3	Стратегія диференційованого маркетингу	Висока якість роботи, гнучкі ціни, ціна залежить від масштабу рішення	Стратегія диференціації

Визначення базової стратегії конкурентної поведінки наведено в таблиці 4.16.

Таблиця 4.17 - Визначення базової стратегії конкурентної поведінки

№ п/п	Чи є проект «першопрохідцем» на ринку?	Чи буде компанія шукати нових споживачів, або забирати існуючих у конкурентів?	Чи буде компанія копіювати основні характеристики товару конкурента, і які?	Стратегія конкурентної поведінки
1	Ні	Обидва варіанти	Буде. Інтерфейс та базові можливості програми	Стратегія виклику лідера

Визначення стратегії позиціонування наведено в таблиці 4.18.

Таблиця 4.18 - Визначення стратегії позиціонування

№ п/п	Вимоги до товару цільової аудиторії	Базова стратегія розвитку	Ключові конкурентоспроможні позиції власного стартап-проекту	Вибір асоціацій, які мають сформувати комплексну позицію власного проекту (три ключових)
1	Точність розпізнавання, відносна легкість рішення	Стратегія диференціації	Якість розпізнавання, універсальність	Позиціювання на якості та швидкості роботи алгоритму

4.5 Розроблення маркетингової програми стартап-проекту

Визначення ключових переваг концепції потенційного товару – таблиці 4.19.

Таблиця 4.19 - Визначення ключових переваг концепції потенційного товару

№ п/п	Потреба	Вигода, яку пропонує товар	Ключові переваги перед конкурентами (існуючі або такі, що потрібно створити
1	Реакція клієнта на товар перед ним	Забезпечує потребу	Переваг в даному контексті немає.
2	Робота офлайн	Забезпечує потребу	Перевага перед тими конкурентами, які не мають такого функціоналу
3	Збір статистики з відео у реальному часі	Забезпечує потребу	Перевага перед усіма конкурентами

Опис трьох рівнів моделі товару приведено в таблиці 4.20.

Таблиця 4.20 - Опис трьох рівнів моделі товару

<table>
<tr><td>Рівні товару</td><td colspan="3">Сутність та складові</td></tr>
<tr><td>I. Товар за задумом</td><td colspan="3">Система, що за допомогою камер розпізнає емоції людей під час розглядання товару або отримання певної послуги. Використовується для подальшої статистики про популярність нових товарів та якості роботи персоналу закладу.</td></tr>
<tr><td rowspan="6">II. Товар у реальному виконанні</td><td>Властивості/характеристики</td><td>М/Нм</td><td>Вр/Тх /Тл/Е/Ор</td></tr>
<tr><td>1. Швидкодія
2. Користувацький інтерфейс
3. Відмовостійкість
4. Якість роботи алгоритму</td><td>Нм
Нм
Нм
Нм</td><td>Тх/Тл/Е
Е
Тх/Тл
Тх/Е</td></tr>
<tr><td colspan="3">Якість: процент правильного розпізнавання алгоритмом</td></tr>
<tr><td colspan="3">Пакування: налаштування серверу консультантом компанії</td></tr>
<tr><td colspan="3">Марка: SpyMarket</td></tr>
<tr><td rowspan="2">III. Товар із підкріпленням</td><td colspan="3">До продажу: Система, що надає можливість збирати інформацію про задоволеність клієнтів товаром чи наданими послугами</td></tr>
<tr><td colspan="3">Після продажу: Швидкодія, зручний користувацький інтерфейс, підтримка, гарантія якості алгоритму</td></tr>
<tr><td colspan="4">Продукт буде з закритим вихідним кодом. Тобто готовий функціонал скопіювати не буде можливості, в тому числі й алгоритм, що застосовується для розпізнавання.</td></tr>
</table>

Визначення меж встановлення ціни – таблиця 4.21.

Таблиця 4.21 - Визначення меж встановлення ціни

№ п/п	Рівень цін на товари-замінники	Рівень цін на товари-аналоги	Рівень доходів цільової групи споживачів	Верхня та нижня межі встановлення ціни на товар/послугу
1	1000	1500	300000	1000-1200

Формування системи збуту наведено в таблиці 4.22.

Таблиця 4.22 - Формування системи збуту

№ п/п	Специфіка закупівельної поведінки цільових клієнтів	Функції збуту, які має виконувати постачальник товару	Глибина каналу збуту	Оптимальна система збуту
1	Пряма закупка	-	0	+

Концепція маркетингових комунікацій приведена в таблиці 4.23.

Таблиця 4.23 - Концепція маркетингових комунікацій

№ п/п	Специфіка поведінки цільових клієнтів	Канали комунікацій, якими користуються цільові клієнти	Ключові позиції, обрані для позиціонування	Завдання рекламного повідомлення	Концепція рекламного звернення
1	1 група – середні магазини та точки обслуговування клієнтів	Аналіз ринку, технічні виставки	Масштабовані пропозиції	Розповісти про переваги продукту для бізнесу	Масштабованість рішення
2	2 група – великі магазини та точки обслуговування клієнтів	Аналіз ринку, технічні виставки, презентації	Комплексна інтегрована система	Розповісти про переваги продукту для бізнесу	Комплексна інтегрована система

Висновки

Можемо зробити висновок про те, що проект цілком може бути комерційно успішним, оскільки ми знайшли досить великі потенційні групи клієнтів. Конкуренція на ринку існує, а, отже і існують додаткові складнощі при виході на ринок, але завдяки особливостям продукту їх можна подолати. Робимо висновок про те, що подальша імплементація проекту є доцільною. І

в майбутньому необхідно обрати стратегію по покращенню якості алгоритму розпізнавання, який відсутній на ринку для збільшення бази клієнтів та популяризації власного продукту.

ВИСНОВКИ

У роботі був виконаний порівняльний аналіз методів розпізнавання емоцій за обличчям, був детально розглянутий активна модель форми для виділення ознак обличчя та рекурентна нейронна мережа в якості класифікатора.

Після аналізу різних класичних методів для розпізнавання емоцій було доведено меншу ефективність в цілому таких підходів. Було прийнято рішення розглянути перспективи класифікаторів глибокого навчання.

Було детально розглянуто підхід із рекурентними нейронними мережами, зокрема LSTM. Було продемонстровано, що такий підхід дозволяє використовувати менше вхідних даних та будувати модель набагато менших розмірів у порівнянні з звичайними нейронними мережами. В свою чергу це призводить до суттєвого скорочення часу навчання моделі та кількості необхідних для цього ресурсів при незмінній ефективності розпізнавання.

Проаналізувавши метод, було проведено ряд практичних експериментів з використанням багатьох баз даних: KDEF, FacesDB, CK+ та JAFFE.

В результаті на вибраних оптимальних співвідношеннях тренувальних фото та фото для перевірки, що відповідно 1434 та 358 фото, отримали швидкість навчання близько 90 хвилин для чотири ядерного процесора. Середній час розпізнавання складає 5с. При цьому ефективність зв’язки активної моделі форми та LSTM складає 72%, що є середнім результатом для моделей глибокого навчання, побудованих з використанням великих моделей загорткових нейронних мереж.

Отже, у роботі було продемонстровано алгоритм для розпізнання емоцій за обличчям, який є значно легшим для виконання та невимогливим до обчислювальних ресурсів, але при цьому достатньо ефективним.

Для подальшого покращення результатів роботи побудованої моделі найбільш ефективним підходом було б розширити кількість фото у базі даних, щоб охопити якомога більше варіантів вираження емоцій, і в свою чергу, уникнути перенавчання та покращити агрегацію нейронної мережі.

Інше поле для покращення – застосування ознак обличчя, що складаються не з 68 характерних точок, а з 128. Це може значно покращити кількість ознак, що зможе містити в собі тріангуляція Делоне, і відповідно дасть можливість нейронній мережі краще класифікувати емоції за виразом обличчя навіть за наявного розміру бази даних.

Отримані результати можна впровадити у системах, що збирають статистику, та для покращення роботи ботів та роботів, щоб їх розмова з людиною виглядала більш природньою. Також для превентивних мір дану систему можна використовувати для розпізнавання емоцій людей в натовпі, для допомоги виявлення тих, хто веде себе підозріло.

Дане дослідження має велику цінність у сфері дослідження методів розпізнавання емоцій у сфері обмежених даних та обчислювальних ресурсів, де класифікатор не можна навчити, взявши просто дуже велику кількість прикладів.

ПЕРЕЛІК ПОСИЛАНЬ

1. Mehrabian, A. Communication without words. Psychol. Today 1968, Vol. 2, P. 53–56.

2. Kaulard K., Cunningham D.W., Bülthoff H.H., Wallraven C. The MPI facial expression database — A validated database of emotional and conversational facial expressions. PLoS ONE 2012, P. 7.

3. Dornaika F., Raducanu B. Efficient facial expression recognition for human robot interactionT*the 9th International Work-Conference on Artificial Neural Networks on Computational and Ambient Intelligence*, San Sebastián, Spain, 20–22 June 2007, P. 700–708.

4. Bartneck C., Lyons M.J. HCI and the fa3ce: Towards an art of the soluble. *In Proceedings of the International Conference on Human-Computer Interaction: Interaction Design and Usability*, Beijing, China, 22–27 July 2007, P. 20–29.

5. Hickson S., Dufour N., Sud A., Kwatra V., Essa I. Eyemotion: Classifying facial expressions in VR using eye-tracking cameras. *IEEE Winter Conference on Applications of Computer Vision 2019* (WACV) P. 1626-1635.

6. Chen C.H., Lee I.J., Lin L.Y. Augmented reality-based self-facial modeling to promote the emotional expression and social skills of adolescents with autism spectrum disorders. Res. Dev. Disabil. 2015, P. 396–403.

7. Assari M.A., Rahmati M. Driver drowsiness detection using face expression recognition. *The IEEE International Conference on Signal and Image Processing Applications*, Kuala Lumpur, Malaysia, 16–18 November 2011, P. 337–341.

8. Zhan C., Li W., Ogunbona P., Safaei F. A real-time facial expression recognition system for online games. *International Journal of Computer Games Technology*, 2008, P. 10-20.

9. Mourão A., Magalhães J. Competitive affective gaming: Winning with a smile. *The ACM International Conference on Multimedia*, Barcelona, Spain, 21–25 October 2013, P. 83–92.

10. Lucey P., Cohn J.F., Kanade T., Saragih J., Ambadar Z., Matthews, I. The extended Cohn-Kanade Dataset (CK+): A complete dataset for action unit and emotion-specified expression. *The IEEE Conference on Computer Vision and Pattern Recognition Workshops*, San Francisco, CA, USA, 13–18 June 2010, P. 94–101.

11. Kahou, S.E., Michalski V., Konda K. Recurrent neural networks for emotion recognition in video. *The ACM on International Conference on Multimodal Interaction*, Seattle, WA, USA, 9–13 November 2015, P. 467–474.

12. Walecki R., Rudovic O. Deep structured learning for facial expression intensity estimation. *Image Vis. Comput*. 2017, P. 143–154.

13. Kim D.H., Baddar W., Jang J., Ro Y.M. Multi-objective based Spatio-temporal feature representation learning robust to expression intensity variations for facial expression recognition. *IEEE Transactions on Affective Computing*, 2017, P. 223-236.

14. Ekman P., Friesen W.V. Facial Action Coding System: Investigator's Guide, 1st ed., Consulting Psychologists Press: Palo Alto, CA, USA, 1978, P. 1–15.

15. Hamm J., Kohler C.G., Gur R.C., Verma R. Automated facial action coding system for dynamic analysis of facial expressions in neuropsychiatric disorders. *Journal of neuroscience methods*, 2011, P. 237–256.

16. Jeong M., Kwak S.Y., Ko B.C., Nam J.Y. Driver facial landmark detection in real driving situation. *IEEE Transactions on Circuits and Systems for Video Technology*, 2017, P. 1–15.

17. Tao S.Y., Martinez A.M. Compound facial expressions of emotion. *Natl. Acad. Sci*. 2014, P. 1454–1462.

18. Benitez-Quiroz C.F., Srinivasan R., Martinez A.M. EmotioNet: An accurate, real-time algorithm for the automatic annotation of a million facial

expressions in the wild. *The IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, NV, USA, 26 June–1 July 2016, P. 5562–5570.

19. Kolakowaska A. A review of emotion recognition methods based on keystroke dynamics and mouse movements. *The 6th International Conference on Human System Interaction*, Gdansk, Poland, 6–8 June 2013, P. 548–555.

20. Kumar S. Facial expression recognition: A review. *The National Conference on Cloud Computing and Big Data*, Shanghai, China, 4–6 November 2015, P 159–162.

21. Ghayoumi M. A quick review of deep learning in facial expression. *Journal of Communication and Computer*, 2017, P. 34–38.

22. Mavadati S.M., Mahoor M.H., Bartlett K., Trinh P., Cohn J. DISFA: A spontaneous facial action intensity database. *IEEE Transactions on Affective Computing*, 2013, P. 151–160.

23. Yin L. et. al. A 3D facial Expression database for facial behavior research. *The International Conference on Automatic Face and Gesture Recognition*, Southampton, UK, 10–12 April 2006, P. 211–216.

24. Lyons M.J. et. al. Coding facial expressions with Gabor wave. *The IEEE International Conference on Automatic Face and Gesture Recognition*, Nara, Japan, 14–16 April 1998, P. 200–205.

25. B+. URL: https://computervisiononline.com/dataset/1105138686 (дата звернення 19.07.2019).

26. MMI. URL: https://mmifacedb.eu/ (дата звернення 19.07.2019).

27. Zhang X. et. al. BP4D-Spontaneous: A high resolution spontaneous 3D dynamic facial expression database. ImageVis. Comput. 2014, 32, P. 692–706.

28. KDEF. URL: http://www.emotionlab.se/resources/kdef (дата звернення 19.07.2019).

29. FacesDb. URL: http://app.visgraf.impa.br/database/faces (дата звернення 19.07.2019).

ДОДАТОК А. Лістинг файлу length.py

```
import cv2
import numpy as np
import dlib
import math
from sklearn.preprocessing import normalize

def extract_index_nparray(nparray):
    index = None
    for num in nparray[0]:
        index = num
        break
    return index

def get_geom(path_to_img):
    # get face landmarks
    img = cv2.imread(path_to_img)
    img_gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
    mask = np.zeros_like(img_gray)

    detector = dlib.get_frontal_face_detector()
    predictor = dlib.shape_predictor("shape_predictor_68_face_landmarks.dat")
    faces = detector(img_gray)
    for face in faces:
        landmarks = predictor(img_gray, face)
        landmarks_points = []
        for n in range(0, 68):
            x = landmarks.part(n).x
            y = landmarks.part(n).y
            landmarks_points.append((x, y))
        points = np.array(landmarks_points, np.int32)
        convexhull = cv2.convexHull(points)
        cv2.fillConvexPoly(mask, convexhull, 255)
```

```
# get delaunay triangulation
rect = cv2.boundingRect(convexhull)
subdiv = cv2.Subdiv2D(rect)
subdiv.insert(landmarks_points)
triangles = subdiv.getTriangleList()
triangles = np.array(triangles, dtype=np.int32)
lines = []
indexes_triangles = []
for t in triangles:
    pt1 = (t[0], t[1])
    pt2 = (t[2], t[3])
    pt3 = (t[4], t[5])

    lines.append([pt1,pt2])
    lines.append([pt2,pt3])
    lines.append([pt1,pt3])

    index_pt1 = np.where((points == pt1).all(axis=1))
    index_pt1 = extract_index_nparray(index_pt1)

    index_pt2 = np.where((points == pt2).all(axis=1))
    index_pt2 = extract_index_nparray(index_pt2)

    index_pt3 = np.where((points == pt3).all(axis=1))
    index_pt3 = extract_index_nparray(index_pt3)

    if index_pt1 is not None and index_pt2 is not None and index_pt3 is not None:
        triangle = [index_pt1, index_pt2, index_pt3]
        indexes_triangles.append(triangle)
length = []
for triangle_index in indexes_triangles:
    pt1 = landmarks_points[triangle_index[0]]
    pt2 = landmarks_points[triangle_index[1]]
    pt3 = landmarks_points[triangle_index[2]]
    length.append(math.hypot(pt2[0]-pt1[0],pt2[1]-pt1[1]))
    length.append(math.hypot(pt2[0]-pt3[0],pt2[1]-pt3[1]))
    length.append(math.hypot(pt3[0]-pt1[0],pt3[1]-pt1[1]))
```

```
    length = normalize([length])
    return length[0], indexes_triangles

def get_next_geom(path_to_img,indexes_triangles):
    # get face landmarks
    img = cv2.imread(path_to_img)
    img_gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
    mask = np.zeros_like(img_gray)
    detector = dlib.get_frontal_face_detector()
    predictor = dlib.shape_predictor("shape_predictor_68_face_landmarks.dat")
    faces = detector(img_gray)
    for face in faces:
        landmarks = predictor(img_gray, face)
        landmarks_points = []
        for n in range(0, 68):
            x = landmarks.part(n).x
            y = landmarks.part(n).y
            landmarks_points.append((x, y))

    # Triangulation of the second face, from the first face delaunay triangulation
    length = []
    for triangle_index in indexes_triangles:
        pt1 = landmarks_points[triangle_index[0]]
        pt2 = landmarks_points[triangle_index[1]]
        pt3 = landmarks_points[triangle_index[2]]

        length.append(math.hypot(pt2[0]-pt1[0],pt2[1]-pt1[1]))
        length.append(math.hypot(pt2[0]-pt3[0],pt2[1]-pt3[1]))
        length.append(math.hypot(pt3[0]-pt1[0],pt3[1]-pt1[1]))

    length = normalize([length])
    return length[0]
```

ДОДАТОК Б. Лістинг файлу dataset.py

```
from length import get_geom, get_next_geom
import emotions as emo
from collections import defaultdict
import pickle
import os

def main():
    print('Starting')
    path_to_dataset = os.path.join('..','_Dataset','facesdb','OriginalImg')
    geom_approach(path_to_dataset)
    print('Ending')

def geom_approach(path_to_dataset):
    shapes, classes = make_geometry(path_to_dataset)
    print(classes[0:15])
    save_landmarks('geometry',shapes)
    labels = emo.classes_to_labels(classes)
    save_landmarks('geometry_answers',labels)

'''
Function to obtain a list of files from a directory
dir - path to dataset
'''
def list_files(dir):
videos_by_class = defaultdict(list)

subdirs = [x[0] for x in os.walk(dir) if len(x[0].split('/')) >= 1]

for subdir in sorted(subdirs):
        files = next(os.walk(subdir))[2]
        for file in sorted(files):
                if file.endswith('.bmp') or file.endswith('.png') or file.endswith('.jpg') or file.endswith('.JPG') or file.endswith('.tiff'):
```

```
                    file_name = os.path.join(subdir, file)
                    videos_by_class[os.path.basename(subdir)].append(file_name)

    return videos_by_class

def make_geometry(path_to_dataset):
    files = list_files(path_to_dataset)
    shapes = list()
    classes = list()

    shapes_anger = list()
    shapes_disgust = list()
    shapes_fear = list()
    shapes_joy = list()
    shapes_neutral = list()
    shapes_sadness = list()
    shapes_surprise = list()

    indexes = []
    for temp_class, files in files.items():
        for file in files:
            print('Process file: ',file)
            if len(indexes) == 0:
                shape, indexes = get_geom(file)
                save_landmarks('indexes',indexes)
            else:
                shape = get_next_geom(file, indexes)
            print(len(shape))
            if len(shape) != 0:
                if temp_class == 'anger':
                    shapes_anger.append(shape)
                elif temp_class == 'disgust':
                    shapes_disgust.append(shape)
                elif temp_class == 'fear':
                    shapes_fear.append(shape)
                elif temp_class == 'joy':
                    shapes_joy.append(shape)
```

```
                elif temp_class == 'neutral':
                    shapes_neutral.append(shape)
                elif temp_class == 'sadness':
                    shapes_sadness.append(shape)
                elif temp_class == 'surprise':
                    shapes_surprise.append(shape)

        while len(shapes_anger) != 0 and len(shapes_disgust) != 0 and len(shapes_fear) != 0 and len(shapes_joy) != 0 and len(shapes_neutral) != 0 and len(shapes_sadness) != 0 and len(shapes_surprise) != 0:
            if len(shapes_anger) != 0:
                shapes.append(shapes_anger.pop(0))
                classes.append('anger')
            if len(shapes_disgust) != 0:
                shapes.append(shapes_disgust.pop(0))
                classes.append('disgust')
            if len(shapes_fear) != 0:
                shapes.append(shapes_fear.pop(0))
                classes.append('fear')
            if len(shapes_joy) != 0:
                shapes.append(shapes_joy.pop(0))
                classes.append('joy')
            if len(shapes_neutral) != 0:
                shapes.append(shapes_neutral.pop(0))
                classes.append('neutral')
            if len(shapes_sadness) != 0:
                shapes.append(shapes_sadness.pop(0))
                classes.append('sadness')
            if len(shapes_surprise) != 0:
                shapes.append(shapes_surprise.pop(0))
                classes.append('surprise')

        return shapes, classes

    def save_landmarks(name,landmarks):
        file_name = name + '.txt'
        with open(file_name,'wb') as fp:
```

```
        pickle.dump(landmarks, fp)

def read_landmarks(file):
    file_name = file + '.txt'
    with open(file_name,'rb') as fp:
        test = pickle.load(fp)
    return test

if __name__ == "__main__":
    # execute only if run as a script
    main()
```

ДОДАТОК В. Лістинг файлу lstm.py

```
# lstm model
from numpy import mean
from numpy import std
from numpy import dstack
import numpy as np
from face_landmarks import detect_landmarks
from length import get_next_geom
from tensorflow.keras.models import Sequential
from tensorflow.keras.models import load_model
from tensorflow.keras.layers import Dense
from tensorflow.keras.layers import Flatten
from tensorflow.keras.layers import Dropout
from tensorflow.keras.layers import LSTM
from tensorflow.keras.utils import to_categorical
from tensorflow.keras import regularizers
from keras import initializers
from matplotlib import pyplot
import pickle
import os
import emotions as emo
import math
import sys

def read_landmarks(file):
    file_name = file + '.txt'
    with open(file_name,'rb') as fp:
        test = pickle.load(fp)
    return test

# load a dataset group, such as train or test
def load_dataset_group():
    X = read_landmarks('geometry')
    y = read_landmarks('geometry_answers')
    return X, y
```

```
# load the dataset, returns train and test X and y elements
# fit and evaluate a model
def evaluate_model(trainX, trainy, testX, testy):
verbose, epochs, batch_size = 1, 25, 64
model = Sequential([
        LSTM(200, activation='relu', recurrent_activation='sigmoid', return_sequences=True, input_shape=(1,342)),
        LSTM(200, activation='relu', recurrent_activation='sigmoid', return_sequences=True),
        LSTM(100, activation='relu', recurrent_activation='sigmoid'),
        Dropout(0.5),
        Dense(100, activation='relu'),
        Dense(7, activation='softmax')
        ])
model.compile(loss='mean_squared_error', optimizer='adam', metrics=['accuracy'])
# fit network
model.fit(trainX, trainy, epochs=epochs, batch_size=batch_size, verbose=verbose)
# evaluate model
_, accuracy = model.evaluate(testX, testy, batch_size=batch_size, verbose=0)
return accuracy, model

# summarize scores
def summarize_results(scores):
print(scores)
m, s = mean(scores), std(scores)
print('Accuracy: %.3f%% (+/-%.3f)' % (m, s))

def make_prediction(path_to_file, model):
   #1. Open image
   #2. Detect landmarks
   indexes = read_landmarks('indexes')
   shape = get_next_geom(path_to_file,indexes)
   #3. Prediction through module
   data_input = list()
   data_input.append(shape)
   data_input = np.array(data_input).reshape(1,1,342)
   answer = model.predict(data_input)
```

```
    answer = np.around(answer, decimals=5)
    print(answer)
    answer = np.argmax(answer)
    #4. Shift back result
    print(emo.Emotions(answer+1).name)

# run an experiment
def run_experiment(model,repeats=10):
X,y = load_dataset_group()
X = np.array(X).reshape(len(X),1,342)
y = np.array(y).reshape(len(y),7,1)
split_num = math.ceil(len(X)*0.8)
X_train = X[:split_num]
y_train = y[:split_num]
X_test = X[split_num+1:]
y_test = y[split_num+1:]
# repeat experiment
scores = list()
for r in range(repeats):
        if model == None:
                score, model = evaluate_model(X_train, y_train, X_test, y_test)
        else:
                score, model = model_train(model,X_train, y_train, X_test, y_test)
        score = score * 100.0
        print('>#%d: %.3f' % (r+1, score))
        scores.append(score)
# summarize results
summarize_results(scores)
return model

def model_train(model,trainX, trainy, testX, testy):
    verbose, epochs, batch_size = 1, 25, 64
    model.fit(trainX, trainy, epochs=epochs, batch_size=batch_size, verbose=verbose)
    _, accuracy = model.evaluate(testX, testy, batch_size=batch_size, verbose=0)
    return accuracy, model

def check(model):
```

```
    X,y = load_dataset_group()
    X = np.array(X).reshape(len(X),1,342)
    y = np.array(y).reshape(len(y),7,1)
    split_num = math.ceil(len(X)*0.8)
    true_ans = 0
    wrong = []
    right = []
    for i in range(len(X[split_num+1:])):
        data_input = list()
        data_input.append(X[split_num+i])
        data_input = np.array(data_input).reshape(1,1,342)
        answer = model.predict(data_input)
        answer = np.argmax(answer)
        chk = np.argmax(y[split_num+i])
        if answer == chk:
            true_ans = true_ans + 1
            right.append(emo.Emotions(chk+1).name)
        else:
            wrong.append(emo.Emotions(chk+1).name)
    print(true_ans/len(X[split_num+1:]))
    unique, counts = np.unique(right, return_counts=True)
    print(dict(zip(unique, counts)))
    unique, counts = np.unique(wrong, return_counts=True)
    print(dict(zip(unique, counts)))

if __name__ == '__main__':
print('Starting:', '\n')

import argparse

parser = argparse.ArgumentParser(description='States of running')
parser.add_argument('-model',   type=str, default=None,
help='Specify model, if you want to use it for training, or pass None to train a new one')
parser.add_argument('-check_only',       type=str, default=False,
help='True if skip training and only make check. Model required')
parser.add_argument('-img',              type=str, default=None,
help='Path to image to check. Will only check image, if specified. Model required')
```

```
args = parser.parse_args()

model = args.model

if args.check_only:
        if model == None:
                print('Model require to perform a check')
                sys.exit()
        elif args.img == None:
                check(model)
                sys.exit()
else:
        if args.img == None:
                model = run_experiment(model,repeats=1)
                model.save('model.hdf5')

if args.img != None:
        if model == None:
                print('Model require to perform a check')
                sys.exit()
        img = os.path.join(args.img)
        ans = make_prediction(img, model)
        print(ans)

print('\nEnding:')
```