

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ТАВРІЙСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ В. І. ВЕРНАДСЬКОГО
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ МУНІЦИПАЛЬНОГО
УПРАВЛІННЯ ТА МІСЬКОГО ГОСПОДАРСТВА
КАФЕДРА КОМП'ЮТЕРНИХ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ**

До захисту допущено

Завідувач кафедри

_____ Олександр ГУЙДА

“ ____ ” _____ 2023 р.

ПОЯСНЮВАЛЬНА ЗАПИСКА

до магістерської кваліфікаційної роботи освітнього ступеня **“магістр”**

з галузі знань 12 «Інформаційні технології»

спеціальності 122 «Комп'ютерні науки»

на тему: Розробка та дослідження алгоритмів для автоматичного
розпізнавання емоцій на зображеннях та відео.

Студент групи КНМ(з)-21 Зайченко Владислава Романівна _____
(шифр групи) (прізвище, ім'я, по батькові) (підпис)

Керівник роботи д.т.н., професор Дорошенко Ю. О. _____
(вчені ступінь та звання, прізвище, ініціали) (підпис)

Консультанти:

охорона праці та навко-
лишнього середовища

доцент Гуйда О.Г. _____
(вчені ступінь та звання, прізвище, ініціали) (підпис)

Київ - 2023

**ТАВРІЙСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ В. І. ВЕРНАДСЬКОГО
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ МУНІЦИПАЛЬНОГО
УПРАВЛІННЯ ТА МІСЬКОГО ГОСПОДАРСТВА**

Другий освітній рівень

Галузь знань 12 «ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ»

(шифр і назва)

Спеціальність 122 «КОМП'ЮТЕРНІ НАУКИ»

(шифр і назва)

освітньо-професійна програма «КОМП'ЮТЕРНІ НАУКИ»

(шифр і назва)

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ О. Г. Гуйда

“ ___ ” _____ 20__ р.

З А В Д А Н Н Я
НА МАГІСТЕРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ
студента Зайченко Владислави Романівни
(прізвище, ім'я, по батькові)

1 Тема роботи Розробка та дослідження алгоритмів для автоматичного розпізнавання емоцій на зображеннях та відео.

Керівник роботи д.т.н., професор Дорошенко Ю. О.,
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені ректором Університету від “25” вересня 2023 року

2 Строк подання студентом роботи “ 5 ” грудня 2023 р.

3 Вихідні дані до роботи Нормативна та довідкова література за темою магістерської роботи, алгоритми та технології розпізнавання зображень, комп'ютерний зір, типи емоцій людини, нейромережеві технології.

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

4.1 Провести дослідження алгоритмів розпізнавання емоцій.

4.2 Розробити оптимальний алгоритм розпізнавання емоцій.

4.3 Розробити програмний прототип для автоматичного розпізнавання емоцій.

4.4 розкрити питання охорони праці та навколишнього середовища при розробці технології.

4.5 Провести розрахунок витрат на розробку технології та дослідити ефективність роботи розробленого алгоритму.

5 Перелік графічного матеріалу:

Графічна робота виконана у вигляді мультимедійної презентації

6 Консультанти розділів роботи:

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Загальна частина	професор Дорошенко Ю. О.		
Технологічна частина	професор Дорошенко Ю. О.		
Спеціальна частина	професор Дорошенко Ю. О.		
Охорона праці та навколишнього середовища	доцент Гуйда О. Г.		
Економічна частина	професор Дорошенко Ю. О.		
Графічна частина	ст. викладач Фуртат О. В.		

7 Дата видачі завдання « 26 » вересня 2023 року

КАЛЕНДАРНИЙ ПЛАН

Назва етапів магістерської кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
<i>Загальна частина</i>	<i>жовтень</i>	
<i>Технологічна частина</i>	<i>жовтень</i>	
<i>Спеціальна частина</i>	<i>листопад</i>	
<i>Охорона праці та навколишнього середовища</i>	<i>жовтень</i>	
<i>Економічна частина</i>	<i>листопад</i>	
<i>Графічна частина</i>	<i>грудень</i>	

Студент _____

(підпис)

Владислава ЗАЙЧЕНКО
(прізвище та ініціали)

Керівник роботи _____

(підпис)

Юрій ДОРОШЕНКО
(прізвище та ініціали)

РЕФЕРАТ

Структура та обсяг роботи. Пояснювальна записка магістерської дисертації складається з п'яти розділів, містить 106 сторінок, 30 рисунків та 39 джерел.

Актуальність теми. Актуальність роботи визначається потребою вдосконалення систем обробки та аналізу великих обсягів візуальної інформації в сучасному інформаційному суспільстві, де величезний обсяг даних швидко набуває обличчя. Один із сучасних напрямків досліджень полягає у розробці алгоритмів для автоматичного розпізнавання емоцій людей на зображеннях та відео. Здатність системи розпізнавання емоцій адекватно взаємодіяти з візуальним контентом має велике значення для різних сфер, таких як розваги, медицина, маркетинг та інші.

Мета і завдання дослідження. Метою роботи є розробка, програмна реалізація та експериментальне дослідження алгоритмів для автоматичного розпізнавання емоцій людей на зображеннях та відео обличч людей.

Для досягнення поставленої мети, необхідно виконати такі завдання:

- 1) провести аналіз технологій розробки та дослідження алгоритмів для автоматичного розпізнавання емоцій людей на зображеннях та відео;
- 2) охарактеризувати створення та опис алгоритмів для автоматичного розпізнавання емоцій на зображеннях та відео;
- 3) скласти комплексний алгоритм автоматичного розпізнавання емоцій людини за зображенням її обличчя та подати його у вигляді структурно-функціональної схеми;
- 4) розробити за складеним алгоритмом програмний прототип для автоматичного розпізнавання емоцій людини.
- 5) експериментально апробувати і дослідити роботу програмного прототипу для автоматичного розпізнавання емоцій людини.

Об'єкт дослідження. Процес автоматичного розпізнавання емоцій людини за зображенням чи відео обличчя.

Предмет дослідження. Розробка та вивчення алгоритмів, спрямованих на автоматичне розпізнавання емоцій людини за зображенням чи відео обличчя.

Методи дослідження. В роботі використовуються алгоритми машинного навчання та нейронні мережі.

Наукова новизна. З використанням машинного навчання та нейронної мережі розроблено оригінальний комплексний алгоритм автоматичного розпізнавання емоцій людини за зображенням чи відео обличчя.

Практичне значення отриманих результатів. Розроблена система цілком підходить для будь-якого комерційного чи некомерційного проєкту, який має потребу в аналізі емоцій людини які виявляються на зображеннях та відеоматеріалах. Наразі розроблений програмний прототип, який може розпізнавати 7 основних емоційних станів людини: радість, подив, страх, відраза, сум, гнів, інтерес.

Ключові слова: емоції, машинне навчання, алгоритм, зображення, відео, розпізнавання, нейронні мережі.

ЗМІСТ

ОСНОВНІ УМОВНІ ПОЗНАЧЕННЯ	8
ВСТУП	9
1 АНАЛІЗ ТЕХНОЛОГІЙ РОЗРОБКИ ТА ДОСЛІДЖЕННЯ АЛГОРИТМІВ ДЛЯ АВТОМАТИЧНОГО РОЗПІЗНАВАННЯ ЕМОЦІЙ НА ЗОБРАЖЕННЯХ ТА ВІДЕО.....	11
1.1 Загальна концепція розпізнавання емоцій.....	11
1.2 Алгоритм як невід’ємна складова попередньої обробки зображень	19
1.3 Алгоритм комп’ютерного зору для виокремлення на зображенні обличчя та визначення його основних характеристик.....	21
1.4 Висновки до першої частини	28
2 СТВОРЕННЯ ТА ОПИС АЛГОРИТМІВ ДЛЯ АВТОМАТИЧНОГО РОЗПІЗНАВАННЯ ЕМОЦІЙ НА ЗОБРАЖЕННЯХ ТА ВІДЕО	29
2.1 Використання рекурентних нейронних мереж для автоматичного розпізнавання емоцій людини на зображеннях та відео її обличчя.....	29
2.2 Алгоритм AdaBoost.....	36
2.3 Гістограма напрямлених градієнтів	41
2.4 Застосування методу опорних векторів для розпізнавання емоцій.....	55
2.5 Висновки до другої частини.....	66

					МКР.122.001.ПЗ						
					<i>Розробка та дослідження алгоритмів для автоматичного розпізнавання емоцій на зображеннях та відео Пояснювальна записка</i>						
Змн.	Лист	№ докум.	Підпис	Дата					Літ.	Арк.	Аркушів
Розроб.		Зайченко В. Р.								6	103
Перевір.		Дорошенко О.В.							ННІМУМГ гр.КНм-21		
Н. Контр.		Фуртат О.В.									
Затверд.		Гуйда О. Г.									

3 РОЗРОБКА ПРОГРАМНОГО ПРОТОТИПУ ДЛЯ АВТОМАТИЧНОГО РОЗПІЗНАВАННЯ ЕМОЦІЙ НА ЗОБРАЖЕННЯХ ТА ВІДЕО	67
3.1 Розробка та налаштування програмного прототипу	67
3.2 Тестування програмного прототипу для автоматичного розпізнавання емоцій на зображеннях та відео	74
3.3 Висновки до третьої частини	76
4 ОХОРОНА ПРАЦІ ТА НАВКОЛИШНЬОГО СЕРЕДОВИЩА.....	78
4.1 Вимоги ергономіки до організації робочого місця оператора ПК.....	78
4.2 Заходи захисту від випромінювань оптичного діапазону.....	80
4.3 Охорона навколишнього природного середовища.....	82
4.4 Висновки до четвертої частини	85
5 ЕКОНОМІЧНА ЧАСТИНА	87
5.1 Розрахунок витрат на розробку технології.....	87
5.2 Висновки до п'ятої частини	90
ЗАГАЛЬНІ ВИСНОВКИ	92
ПЕРЕЛІК ПОСИЛАНЬ	96
Додаток А.....	100
Додаток Б	102

ОСНОВНІ УМОВНІ ПОЗНАЧЕННЯ

грн. - гривень

див. – дивитися

ДСТУ – Державний стандарт України

рис. – рисунок

AdaBoost – Adaptive Boosting

CNN – Convolutional Neural Networks

GRU – Gated Recurrent Unit

HOG – Histogram of Oriented Gradients

LSTM – Long Short-Term Memory

OpenCV – Open Source Computer Vision Library

RFE – Recursive Feature Elimination

RNN – Recurrent Neural Networks

SFS – Sequential Feature Selection

SSD – Single Shot MultiBox Detector

SVM – Support Vector Machine

ВСТУП

Актуальність роботи. В сучасному інформаційному суспільстві, де величезний обсяг даних швидко набуває обличчя, виникає необхідність вдосконалення систем обробки та аналізу великих обсягів візуальної інформації. Одним із актуальних напрямків досліджень в цьому контексті є розробка алгоритмів для автоматичного розпізнавання емоцій на зображеннях та відео. Здатність системи розпізнавання емоцій адекватно взаємодіяти з візуальним контентом має велике значення для ряду сфер, включаючи розваги, медицину, маркетинг, та інші.

Метою роботи є розробка, програмна реалізація та експериментальне дослідження алгоритмів для автоматичного розпізнавання емоцій людей на зображеннях та відео обличч людей.

Для досягнення поставленої мети, необхідно виконати такі завдання:

- 1) провести аналіз технологій розробки та дослідження алгоритмів для автоматичного розпізнавання емоцій людей на зображеннях та відео;
- 2) охарактеризувати створення та опис алгоритмів для автоматичного розпізнавання емоцій на зображеннях та відео;
- 3) скласти комплексний алгоритм автоматичного розпізнавання емоцій людини за зображенням її обличчя та подати його у вигляді структурно-функціональної схеми;
- 4) розробити за складеним алгоритмом програмний прототип для автоматичного розпізнавання емоцій людини.
- 5) експериментально апробувати і дослідити роботу програмного прототипу для автоматичного розпізнавання емоцій людини.

Об'єкт дослідження. Процес автоматичного розпізнавання емоцій людини за зображенням чи відео обличчя.

Предмет дослідження. Розробка та вивчення алгоритмів, спрямованих на автоматичне розпізнавання емоцій людини за зображенням чи відео обличчя.

Методи дослідження. В роботі використовуються алгоритми машинного навчання та нейронні мережі.

Наукова новизна. З використанням машинного навчання та нейронної мережі розроблено оригінальний комплексний алгоритм автоматичного розпізнавання емоцій людини за зображенням чи відео обличчя.

Практичне значення отриманих результатів. Розроблена система цілком підходить для будь-якого комерційного чи некомерційного проекту, який має потребу в аналізі емоцій людини які виявляються на зображеннях та відеоматеріалах. Наразі розроблений програмний прототип, який може розпізнавати 8 основних емоційних станів людини: злість, презирство, огида, страх, щастя, нейтральний стан, сум та здивування.

Результати дослідження мають практичне застосування та можуть знайти використання в різних сферах індустрії та науки, роблячи важливий внесок у вдосконалення сучасних технологій обробки візуальної інформації.

Апробація. Нами була проведена апробація (тестування) програмного комплексу яке представляло собою запровадження певного алгоритму для автоматичного розпізнавання емоцій на зображеннях та відео. У разі якщо людина виражала візуально емоцію, то програма з'єднана з відеокамерою визачала емоцію людини яку вона зображала.

Структура та обсяг роботи. Пояснювальна записка магістерської дисертації складається з п'яти розділів, містить 106 сторінок, 30 рисунків та 39 джерел.

1 АНАЛІЗ ТЕХНОЛОГІЙ РОЗРОБКИ ТА ДОСЛІДЖЕННЯ АЛГОРИТМІВ ДЛЯ АВТОМАТИЧНОГО РОЗПІЗНАВАННЯ ЕМОЦІЙ НА ЗОБРАЖЕННЯХ ТА ВІДЕО

1.1 Загальна концепція розпізнавання емоцій

Загальна концепція розпізнавання емоцій полягає в застосуванні комп'ютерних технологій для виявлення та класифікації емоцій, виражених людьми на зображеннях, відео або інших візуальних даних [2].

Основні етапи цього процесу включають:

- *Збір вхідних даних.*

Процес збору вхідних даних включає в себе отримання та накопичення зображень або відеоматеріалів, які містять інформацію про емоційний вираз обличчя. Це може включати в себе наступні кроки:

1. Обрання джерела, з якого будуть отримані дані. Це може бути набір зображень або відеозаписів, створених спеціально для дослідження або взяті з публічних джерел, таких як бази даних фотографій або відео.
2. Визначення критеріїв вибору зображень або відео, таких як різноманітність емоцій, розподіл за віковими групами чи етнічними особливостями. Створення методології для вибору та збору даних.
3. Створення колекції зображень або відео, де кожен запис включає інформацію про емоційний вираз обличчя. Якщо використовується існуючий датасет, слід перевірити його достовірність та репрезентативність.
4. Врахування питань конфіденційності та етичних аспектів під час збору даних, зокрема, якщо вони включають в себе особисті зображення.
5. Якщо датасет не включає в себе мітки емоцій, важливо визначити процедуру анотації, де експерти чи самі об'єкти анотують кожне зображення чи відео з врахуванням емоційного виразу.

Цей етап є ключовим для успішного розпізнавання емоцій, оскільки якість та репрезентативність вхідних даних безпосередньо впливають на ефективність навчання та випробування алгоритмів.

- *Обробка вхідних даних.*

Перед тим як приступити до обробки даних для розпізнавання емоцій, застосовують різноманітні методи для покращення якості даних та виділення ключових особливостей.

Це включає такі етапи:

1. Пре-процесинг зображень

Нормалізація освітлення. Забезпечення однорівневого освітлення на зображенні для уникнення відхилень у виявленні емоцій через нерівномірне освітлення.

Контрастність та яскравість. Регулювання контрастності та яскравості для поліпшення видимості деталей обличчя та емоційних виразів.

2. Обробка фону.

Видалення або розмиття фону. Це дозволяє зосередитися на обличчі та його емоційних виразах, виключаючи вплив фонових об'єктів.

3. Фільтрація шуму.

Зменшення шуму. Використання фільтрів для зменшення артефактів або шумів, що можуть виникнути при зйомці.

4. Детекція ключових точок обличчя.

Оптичне розпізнавання обличчя (Facial Landmark Detection) для точного визначення положення основних елементів обличчя, таких як очі, ніс та рот.

5. Класифікація та розпізнавання обличчя.

Використання навчених моделей для визначення емоцій на основі виявлених особливостей обличчя.



Рисунок 1.1 – Розпізнавання орієнтирів обличчя [3]

6. Аугментація даних.

Додавання штучних змін до даних. Застосування технік аугментації, таких як обертання, зміщення та зміни розміру, для розширення обсягу тренувальних даних та поліпшення роботи моделі на різних умовах.

Ці методи дозволяють підготувати дані перед їх обробкою алгоритмами розпізнавання емоцій, максимізуючи точність та надійність результатів.

- *Виявлення обличчя.* Використання алгоритмів комп'ютерного зору для локалізації та виділення обличчя на зображенні чи відео.

Використання каскадів класифікаторів, наприклад, класифікатора Хаара, який швидко та ефективно розпізнає обличчя за основними особливостями, такими як границі очей та форма рота.

Використання глибоких нейронних мереж, таких як Single Shot MultiBox Detector (SSD) або Faster R-CNN, які навчаються на великому обсязі даних та можуть впевнено визначати положення обличчя на зображеннях.

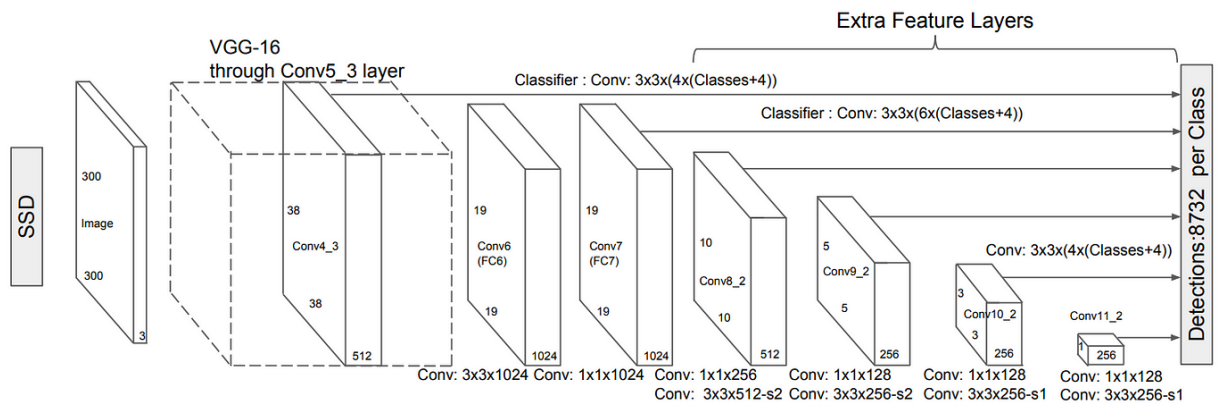


Рисунок 1.2 – Одиночний детектор MultiBox (SSD) [6]

Застосування алгоритмів для відзначення контурів обличчя, щоб підкреслити його форму та розмір.

Використання методів маскування для виділення та ізоляції області, яка відповідає обличчю, роблячи його більш помітним на зображенні чи відео.

Застосування алгоритмів, що адаптуються до руху та змін розміру обличчя в кадрі відео.

Ці алгоритми дозволяють ефективно та точно локалізувати обличчя на вхідних візуальних даних, що є ключовим етапом для подальшого аналізу та розпізнавання емоцій.

- *Витягування ознак обличчя.* Аналіз особливостей, таких як розташування очей, форма рота, вирази обличчя, які можуть свідчити про конкретні емоційні стани.
- *Розпізнавання емоцій.* Для визначення конкретних емоцій або груп емоцій, виявлених на обличчі, використовуються алгоритми машинного навчання, зокрема, моделі класифікації.

Нижче розглянемо загальний підхід до використання цих алгоритмів:

1. Навчання моделі.

- *Створення навчального датасету.* Збір даних, де для кожного обличчя маються мітки з визначенням емоційного стану (сум, радість, гнів, тощо).

- *Виділення ознак.* Вибір ключових ознак, таких як положення очей, рота, експресії, які будуть використовуватися для класифікації.
- *Обробка та аугментація даних.* Перед поданням даних на вхід моделі проводять їх обробку та, можливо, аугментацію для підвищення робастності моделі.

2. Вибір моделі.

- *Глибокі нейронні мережі.* Використання глибоких архітектур, таких як Convolutional Neural Networks (CNN) або Recurrent Neural Networks (RNN), які можуть ефективно вчити складні залежності між ознаками.
Convolutional Neural Networks (CNN).

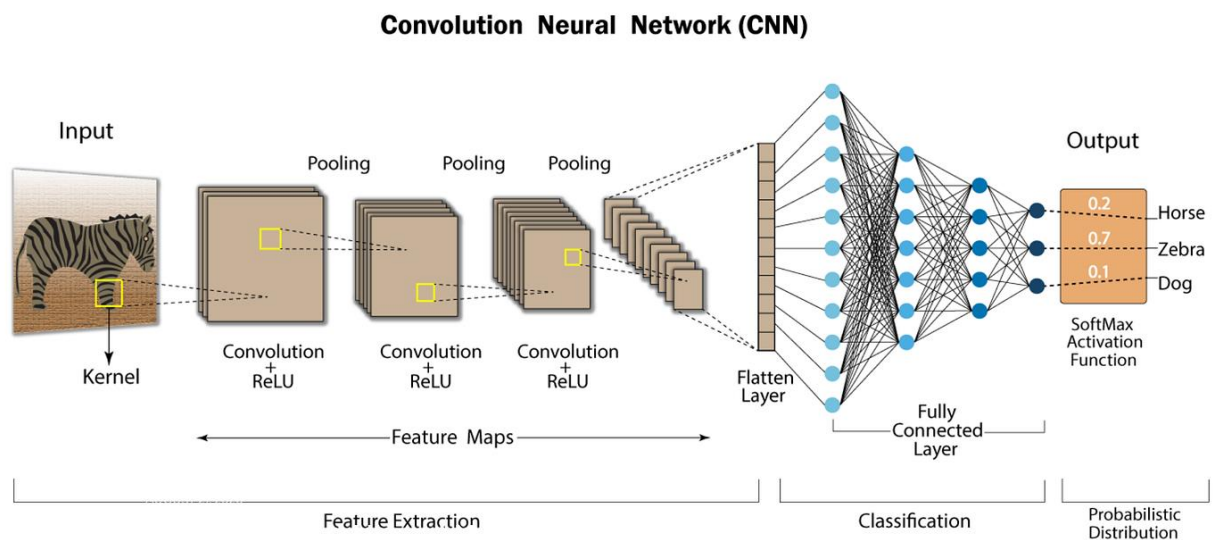


Рисунок 1.3 – Згорточні нейронні мережі (CNN) [13]

CNN широко використовуються для обробки зображень і відео, де просторові властивості грають важливу роль. Вони ефективно впораються з розпізнаванням об'єктів, класифікацією зображень, виявленням облич, сегментацією зображень та багатьма іншими завданнями в області комп'ютерного зору [13].

CNN використовують шари згортки для виявлення локальних функцій в зображеннях. Це дозволяє їм автоматично вивчати корисні фільтри та шаблони. Також вони можуть включати після згортки шари пулінгу для зменшення просторових розмірів.

Recurrent Neural Networks (RNN).

RNN використовуються для обробки послідовностей даних, таких як мовний аналіз, машинний переклад, розпізнавання мови та часовий ряд. Вони можуть ефективно враховувати контекст і залежності між послідовними елементами.

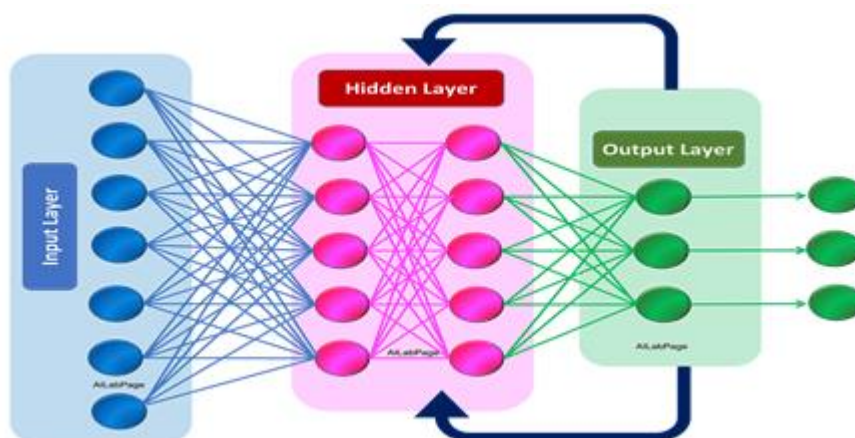


Рисунок 1.4 – Повторювані нейронні мережі (RNN) [34]

RNN мають здатність зберігати попередні стани і використовувати їх для обробки нових вхідних даних [34]. Однак вони можуть стикатися з проблемою зниклого градієнта, що ускладнює навчання на довгих послідовностях. У зв'язку з цим було розроблено вдосконалені архітектури, такі як Long Short-Term Memory (LSTM) і Gated Recurrent Unit (GRU), які допомагають вирішити цю проблему.

Обидві архітектури можуть використовуватися в різноманітних задачах і зазвичай використовуються у поєднанні з іншими техніками для досягнення кращих результатів, наприклад, з Transfer Learning, Batch Normalization, та іншими оптимізаційними методами.

- *Transfer learning.*

Transfer learning – це метод в машинному навчанні, який використовує попередньо навчені моделі для розв'язання нових задач або на нових наборах даних [21].

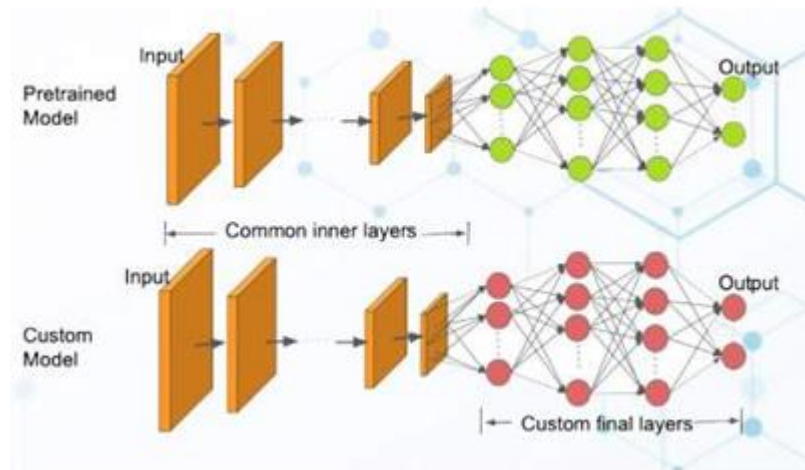


Рисунок 1.5 – Передача навчання [21]

Ідея полягає в тому, щоб використовувати знання, набуте під час навчання на одному завданні, для поліпшення виконання іншого, пов'язаного завдання.

Основні переваги використання попередньо навчених моделей у Transfer Learning включають:

- Попередньо навчені моделі часто навчаються на великих наборах даних, що дозволяє їм вивчати загальні риси та характеристики.
- Моделі, навчені на великих даних, можуть добре узагальнювати свої знання на нові дані, що робить їх корисними для використання в різних задачах.
- Використання попередньо навчених ваг дозволяє значно зменшити час та обсяг даних, необхідний для навчання нової моделі.

Існують два основних підходи до Transfer Learning:

Feature Extraction (вилучення ознак). Використовуючи попередньо навчену модель як «функцію вилучення ознак», ви можете використовувати ваги моделі, щоб отримати представлення вхідних даних, а потім тренувати новий класифікатор на цих представленнях.

Fine-tuning (доналаштування). Замороження частини попередньо навченої моделі та тренування тільки окремих шарів для адаптації до нового завдання. Це дозволяє вагам моделі адаптуватися до нових даних.

Transfer learning використовують під конкретну задачу розпізнавання емоцій.

3. Навчання та оцінка моделі.

- *Тренування моделі.* Подача даних на вхід моделі та її тренування з використанням зваженого втрати (weighted loss) для вирішення нерівноваги у розподілі класів.
- *Оцінка результатів.* Використання набору тестових даних для оцінки точності, відновлення та F1-мери моделі.

4. Використання моделі в реальному часі.

- *Інтеграція з системою розпізнавання обличчя.* З'єднання моделі з алгоритмами розпізнавання обличчя для аналізу в реальному часі.
- *Вдосконалення та налаштування:*
- *Гіперпараметри та оптимізація.* Вдосконалення результатів через налаштування гіперпараметрів, оптимізацію та виправлення помилок.

В результаті цього процесу отримуємо модель, яка здатна класифікувати емоційний стан обличчя на вхідних даних. Ця модель може бути застосована у великому спектрі застосувань, включаючи системи відстеження емоцій у реальному часі, розваги, медицину та інші галузі.

- *Оцінка точності та надійності.* Аналіз результатів та визначення ефективності алгоритму в реальних умовах.

Ця концепція дозволяє використовувати технології розпізнавання емоцій у різноманітних областях, таких як розваги, медицина, психологія, маркетинг та інші, що відкриває нові можливості для вдосконалення взаємодії між технологіями та користувачами.

1.2 Алгоритм як невід’ємна складова попередньої обробки зображень

Початковий етап має важливе значення для отримання зображення та внесення необхідних змін у нього. Ці зміни спрямовані на покращення якості зображення та його максимальну адаптацію для автоматичного аналізу.

На сьогоднішній день існує безліч алгоритмів попередньої обробки зображень, але в даній роботі я бажаю висвітлити три конкретні алгоритми:

а) *Лінійне усереднення сусідніх точок* - це простий тип алгоритму попередньої обробки, який використовується для видалення шуму з зображення.

Лінійне усереднення сусідніх точок – це ефективний метод попередньої обробки зображень, який застосовується для зменшення впливу шумів та вирівнювання текстури. Цей алгоритм базується на простій, але потужній ідеї про те, що кожна точка на зображенні може бути виражена як середнє значення її сусідніх точок.

Основна мета лінійного усереднення - зменшити випадкові коливання яскравості в околі кожної точки та створити більш згладжений, менш шумний образ. Цей метод особливо ефективний в контексті зображень, де шум може бути спричинений різними факторами, такими як електронний шум у цифрових зображеннях чи артефакти при зйомці.

Переваги використання цього методу:

1. *Зменшення шуму*. Лінійне усереднення сусідніх точок може допомогти зменшити шум в зображенні. Вагове середнє значення дозволяє враховувати сусідні пікселі з різною вагою, що корисно при обробці зашумлених зображень.
2. *Зменшення ефекту «мозаїки»*. При зменшенні роздільної здатності (декілька пікселів об’єднуються в один) може виникнути ефект «мозаїки».

Лінійне усереднення дозволяє згладити переходи між сусідніми областями та поліпшити вигляд зображення.

3. *Підвищення роздільної здатності*. З іншого боку, при підвищенні роздільної здатності (зображення збільшується) лінійне усереднення може допомогти

згладити зображення та зменшити артефакти, які можуть виникнути через розтягування пікселів.

4. *Простота реалізації*. Метод лінійного усереднення є відносно простим у реалізації та обчисленнях. Він використовує базові операції, такі як сумування та множення, і не вимагає великої обчислювальної потужності.

Застосування в різних галузях: Лінійне усереднення може застосовуватися в різних галузях, таких як обробка зображень, обробка сигналів та графічний дизайн.

Алгоритм лінійного усереднення може бути описаний наступним чином. Для кожної точки на зображенні береться середнє значення його сусідів, враховуючи певний локальний контекст. Цей локальний контекст може бути визначений за допомогою квадратної або круглої області навколо кожної точки, де проводиться усереднення. Такий підхід дозволяє адаптувати процес усереднення до особливостей структури зображення [16].

Один із ключових плюсів лінійного усереднення – його простота та ефективність. Цей алгоритм може бути застосований до зображень різної природи, від фотографій до медичних знімків. Важливою перевагою є також те, що лінійне усереднення не вимагає складних обчислень та забезпечує відмінні результати в умовах, де присутні локальні аномалії чи шуми.

Однак слід відзначити, що лінійне усереднення може втратити деяку деталізацію на зображенні, особливо в областях з великою зміною яскравості або контрасту. Також важливо обирати оптимальний розмір локального вікна для усереднення, оскільки надто великі вікна можуть впливати на роздільну здатність зображення.

Отже, лінійне усереднення сусідніх точок є простим та ефективним методом попередньої обробки, який знаходить широке застосування в обробці зображень. Його успішна реалізація вимагає уважного вибору параметрів та врахування особливостей конкретного типу зображень.

б) *Лінійне розтягування кольорової гистограми зображення* є методом, призначеним для вдосконалення контрастності зображення. Цей підхід

дозволяє присвоїти нові значення рівням яскравості на вхідному зображенні, що лежать в інтервалі від мінімального до максимального значення. Це робиться з метою охоплення всього можливого діапазону змін яскравості, а саме від значення 0 до 255.

в) *Алгоритм нарощування областей є методом для сегментації зображення.*

Його основна ідея полягає в виборі початкової точки для певної області та поступовому нарощуванні цієї області на зображенні, перевіряючи однорідність сусідніх пікселів.

Таким чином, якщо різниця у яскравості між даною точкою та її сусідом невелика, вони вважаються частинами однієї спільної області. Основна мета цього алгоритму полягає в розділенні зображення на індивідуальні частини для подальшого використання, наприклад, для визначення категорії, до якої належать об'єкти.

1.3 Алгоритм комп'ютерного зору для виокремлення на зображенні обличчя та визначення його основних характеристик

На другому етапі алгоритму розпізнавання емоцій відбувається пошук обличчя користувача, використовуючи характерні ознаки Хаара. Такий же підхід використовується і на третьому етапі для визначення окремих рис обличчя. Алгоритм можна розглядати як послідовність чотирьох етапів:

а) Розрахунок ознак Хаара

Основою цього розрахунку є метод Віоли-Джонса, який вводить так звані примітиви Хаара. Ці примітиви представляють собою розбиття певної прямокутної області на різні прямокутні підобласті. Розрахунок включає в себе сумування інтенсивності пікселів в кожній підобласті та обчислення різниці між сумами. Різниця між значеннями підобластей визначає значення цього примітиву. Цей підхід дозволяє замінити звичайну піксельну систему зображення на систему, засновану на ознаках Хаара. Під час наступних

обчислень, ми будемо використовувати саме цю систему для виділення обличчя та його характеристик.

Приклади примітивів Хаара можемо побачити на рисунку 1.6.

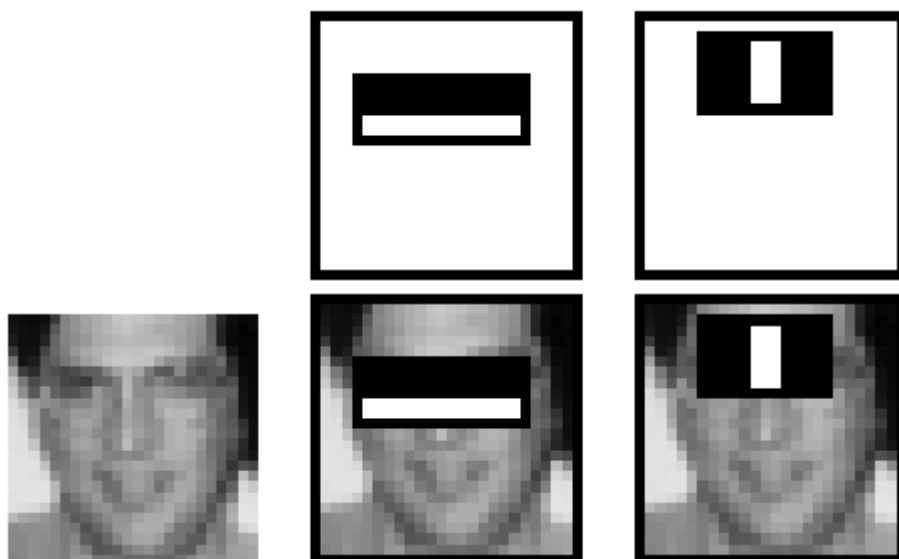


Рисунок 1.6 – Приклади ознак Хаара [6, с.90]

б) Створення інтегральних образів є ключовим кроком, що розширює можливості визначення класифікаторів обличчя. Хоча попередній етап надає нам ефективну систему для визначення обличчя, важливо вдосконалити обчислення, особливо при обробці цілого зображення, де попіксельне вирахування значень може виявитися витратним за часом.

Запропоновано використовувати інтегральне зображення з метою оптимізації обчислень. Інтегральне зображення виступає як проміжне представлення, де для кожного пікселя вхідного зображення встановлюється значення, що відображає суму верхніх лівих пікселів відносно початкового пікселя. Це підходить для швидкого і ефективного визначення областей обличчя та їх характеристик.

Інтегральні образи дозволяють нам значно скоротити час обчислень, враховуючи суми піксельних значень перед визначенням ключових ознак.

Цей метод не лише поліпшує продуктивність, але також допомагає у врахуванні ширшого контексту зображення, що є важливим аспектом в розпізнаванні обличчя та емоцій.

в) Важливо відзначити, що не всі ознаки Хаара є необхідними для ефективного виявлення об'єктів. Суттєві тільки ті ознаки, які належать самому об'єкту. Отже, для визначення оптимальних ознак, що належать обличчю та його рисам, із значною кількістю доступних ознак Хаара використовується алгоритм AdaBoost. Цей алгоритм фактично вибирає найкращі ознаки та навчає класифікатори для їхнього подальшого використання.

AdaBoost застосовує комбінацію слабких класифікаторів з метою створення одного потужного класифікатора, який може ефективно використовуватися для виявлення обличчя та його рис.

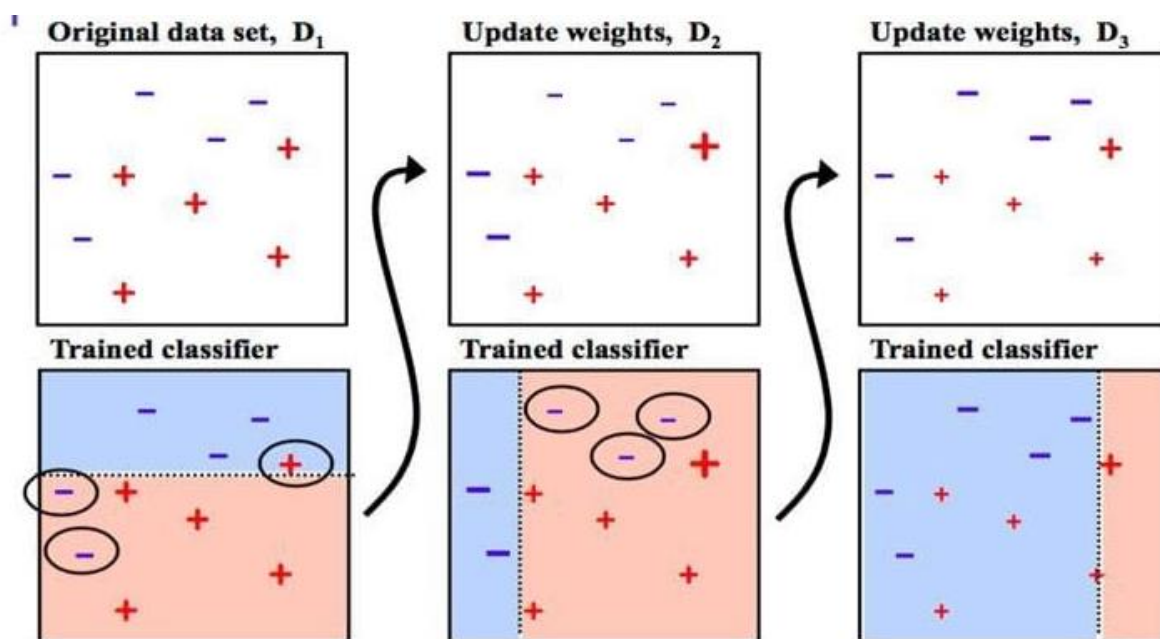


Рисунок 1.7 – Адаптивне посилення [17, с. 280]

Цей підхід дозволяє підкреслити ключові ознаки, що характеризують об'єкт, та оптимізує процес визначення обличчя на зображенні.

AdaBoost виявляється інструментом, який не лише враховує значущість ознак, але і створює потужний класифікатор, здатний до впевненого та точного визначення об'єктів, таких як обличчя та їх риси.

с) Впровадження каскадних класифікаторів представляє собою стратегію, яка допомагає оптимізувати та прискорити процес визначення об'єктів, зокрема обличчя та його риси. Цей метод базується на ідеї послідовного використання класифікаторів різної складності, де кожен наступний класифікатор фільтрує області, які пройшли попередній етап.

Основною перевагою каскадних класифікаторів є ефективність в роботі з великою кількістю ознак та швидке визначення невідповідних областей на ранніх етапах. Це досягається завдяки тому, що більшість нецільових областей відкидаються на ранніх етапах, а лише об'єкти, що виглядають подібно до цільового, піддаються докладнішому аналізу на подальших етапах.

Такий підхід впровадження каскадних класифікаторів сприяє покращенню продуктивності та швидкості процесу розпізнавання об'єктів у великих наборах даних чи в реальному часі.

1.4. Розпізнавання емоцій людини знаходження ключових точок на зображенні її обличчя

Розпізнавання емоцій за допомогою знаходження ключових точок на обличчі є важливим кроком у вдосконаленні точності та надійності систем автоматичного аналізу виразів обличчя. Цей процес включає в себе визначення стратегічних точок, що характеризують основні частини обличчя, такі як очі, ніс, та рот.

Виявлення цих ключових точок здійснюється за допомогою різних технік комп'ютерного зору, таких як аналіз форми та розташування об'єктів на зображенні. Деякі з методів використовують алгоритми машинного навчання для автоматизованого визначення цих точок на різних типах обличчя та в різних емоційних станах.

Знаходження ключових точок на обличчі дозволяє системі отримати детальну інформацію про форму та положення обличчя, що, в свою чергу, допомагає в точному визначенні виразів обличчя та виведенні висновків про емоційний стан особи. Цей підхід стає важливим елементом розвитку систем

розпізнавання емоцій, забезпечуючи більшу точність та адаптивність до різних контекстів та умов.

Отже, відповідно до методу розпізнавання емоцій, наступним кроком є визначення ключових точок рис обличчя.

Процес отримання цих точок включає наступні етапи: перехід від кольорового до напівтонового (grayscale) зображення, подальший перехід до бінарного зображення, визначення границь рис обличчя і виокремлення ключових точок.

Зазвичай перехід від кольорового до напівтонового зображення виконується на етапі захоплення та відстежування обличчя [9].

Для визначення границь об'єктів на зображенні використовується алгоритм Канні. Після цього застосовується гістограма напрямлених градієнтів для виявлення ключових точок. Весь алгоритм розділений на чотири обов'язкові етапи, включаючи видалення шуму за допомогою функції Гауса, виділення градієнту інтенсивності границь за методом Соболя, вилучення не максимумів зображення, подвійна порогова фільтрація та трасування областей неоднозначності. Алгоритм видалення шуму за допомогою функції Гауса був розглянутий нами раніше, а всі інші етапи будуть розглянуті більш детально нижче.

Маски для методу Соболя використовуються для визначення градієнту інтенсивності границь на зображенні. Ці маски представляють собою матриці, які застосовуються до кожного пікселя зображення для визначення напрямку та інтенсивності змін яскравості в околиці цього пікселя. Маски Соболя виглядають наступним чином:

Маска Соболя для визначення горизонтального градієнту (G_x):

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad (1.1)$$

Маска Соболя для визначення вертикального градієнту (G_y):

$$G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad (1.2)$$

Ці маски застосовуються окремо до кожного пікселя зображення для обчислення градієнтів по горизонталі та вертикалі. Зміна яскравості вздовж кожного напрямку враховується, і результати цих операцій використовуються для подальшого визначення напрямку та інтенсивності градієнтів на зображенні.

Використовуючи маски для методу Соболя, можна розрахувати складові градієнта G_x та G_y . Це виконується шляхом множення значень пікселів на відповідні значення масок та сумування усіх отриманих значень.

Після цього за допомогою формул можна визначити значення кожного пікселя для виділення контурів на зображенні.

Насамперед, складові градієнта розраховуються за наступними формулами:

$$\begin{aligned} G_x &= \sum(I * G_{x_mask}) \\ G_y &= \sum(I * G_{y_mask}) \end{aligned} \quad (1.3)$$

де I - значення пікселя на зображенні, G_{x_mask} та G_{y_mask} - відповідні маски для горизонтального та вертикального градієнтів.

Після отримання складових градієнта можна визначити величину градієнта та його напрямок для кожного пікселя за допомогою наступних формул:

Величина градієнта

$$(G) = \sqrt{G_x^2 + G_y^2} \quad (1.4)$$

Напрямок градієнта

$$(\theta) = \arctan \frac{G_y}{G_x} \quad (1.5)$$

Отримані значення можуть бути використані для подальшого виділення контурів та розпізнавання об'єктів на зображенні.

Після визначення напрямів градієнту та значень пікселів за методом Соболя, виконується вилучення (встановлення чорного кольору) пікселів, які мають значення менше, ніж значення двох сусідніх пікселів по напрямку градієнта.

Це виконується за наступними правилами:

1. Якщо кут градієнта пікселя дорівнює 0° , порівнюються пікселі у східному та західному напрямках.
2. Якщо округлений кут градієнта дорівнює 45° , порівнюються пікселі у напрямках північного заходу та південного сходу.
3. Якщо кут градієнта пікселя дорівнює 90° , порівнюються пікселі у північному та південному напрямках.
4. Якщо кут градієнта пікселя дорівнює 135° , порівнюються пікселі у напрямках південного заходу та північного сходу.

Далі виконується порогова фільтрація зображення для виділення сильних та слабких пікселів. За визначеними значеннями слабого та сильного порогів порівнюють значення кожного пікселя. Якщо значення перевищує сильний поріг, пікселю привласнюється максимальне значення. Якщо значення більше за слабкий поріг, але менше за сильний, залишається поточне значення пікселя. Якщо значення менше за слабкий поріг, встановлюється мінімальне значення для цього пікселя.

Останнім етапом алгоритму Канні є трасування області неоднозначності. Для кожного пікселя, який не отримав максимального чи мінімального значення на попередньому кроці, перевіряється наявність сусідніх пікселів з максимальними значеннями. Якщо це твердження вірне, пікселю привласнюється максимальне значення, в іншому випадку – мінімальне значення.

Отже, фінальним етапом після визначення границь рис обличчя є використання глибинного навчання для аналізу емоцій особи на зображенні.

1.4 Висновки до першої частини

У результаті проведеного аналізу методів розробки та вивчення алгоритмів для автоматичного визначення емоцій на зображеннях та відео було виявлено, що використані алгоритми та методи мають важливе значення для точного визначення та аналізу емоцій на вхідних візуальних даних.

Локалізація обличчя є ключовим етапом у цьому процесі, і використані алгоритми, включаючи визначення контурів, маскування та адаптацію до руху та змін розміру обличчя, виявилися ефективними та точними. Ці методи дозволяють ефективно визначати та ізолювати обличчя на вхідних візуальних даних, що створює підґрунтя для подальшого аналізу та розпізнавання емоцій.

Застосування нейронних мереж, зокрема Convolutional Neural Networks (CNN), розширює можливості обробки зображень та відео, забезпечуючи успішне розпізнавання об'єктів, класифікацію зображень та сегментацію. Рекурентні нейронні мережі, такі як Long Short-Term Memory (LSTM) і Gated Recurrent Unit (GRU), вирішують проблему зниклого градієнта та допомагають зберігати попередні стани для обробки нових вхідних даних, що важливо для аналізу динаміки емоцій.

Алгоритм лінійного усереднення виявився ефективним для обробки зображень, проте важливо враховувати оптимальний розмір локального вікна для усереднення та уникнення втрати деталізації в областях з великою зміною яскравості чи контрасту.

У визначенні емоцій використовуються різні методи, включаючи пошук обличчя за допомогою ознак Хаара та глибинне навчання. Ці підходи допомагають виявити та аналізувати емоції особи на зображенні.

Отже, зазначені алгоритми та методи відіграють ключову роль у розробці систем для автоматичного визначення емоцій, надаючи надійні та точні інструменти для аналізу візуальних даних у реальному часі.

2 СТВОРЕННЯ ТА ОПИС АЛГОРИТМІВ ДЛЯ АВТОМАТИЧНОГО РОЗПІЗНАВАННЯ ЕМОЦІЙ НА ЗОБРАЖЕННЯХ ТА ВІДЕО

2.1 Використання рекурентних нейронних мереж для автоматичного розпізнавання емоцій людини на зображеннях та відео її обличчя

Використання рекурентних нейронних мереж для автоматичного визначення емоцій на зображеннях та відео - це підхід, який базується на застосуванні архітектур нейронних мереж із здатністю враховувати контекстуальні залежності в часі. Рекурентні нейронні мережі (RNN) є особливим типом нейронних мереж, що дозволяє моделювати послідовність даних та враховувати попередні інформаційні стани [8].

У визначенні емоцій на зображеннях та відео, де послідовність кадрів може мати важливий вплив на розпізнавання емоційного стану, RNN виявляється корисним інструментом. Ця архітектура дозволяє зберігати та використовувати попередні інформаційні стани при обробці нових кадрів, що може покращити точність розпізнавання емоцій на послідовних вхідних даних [8].

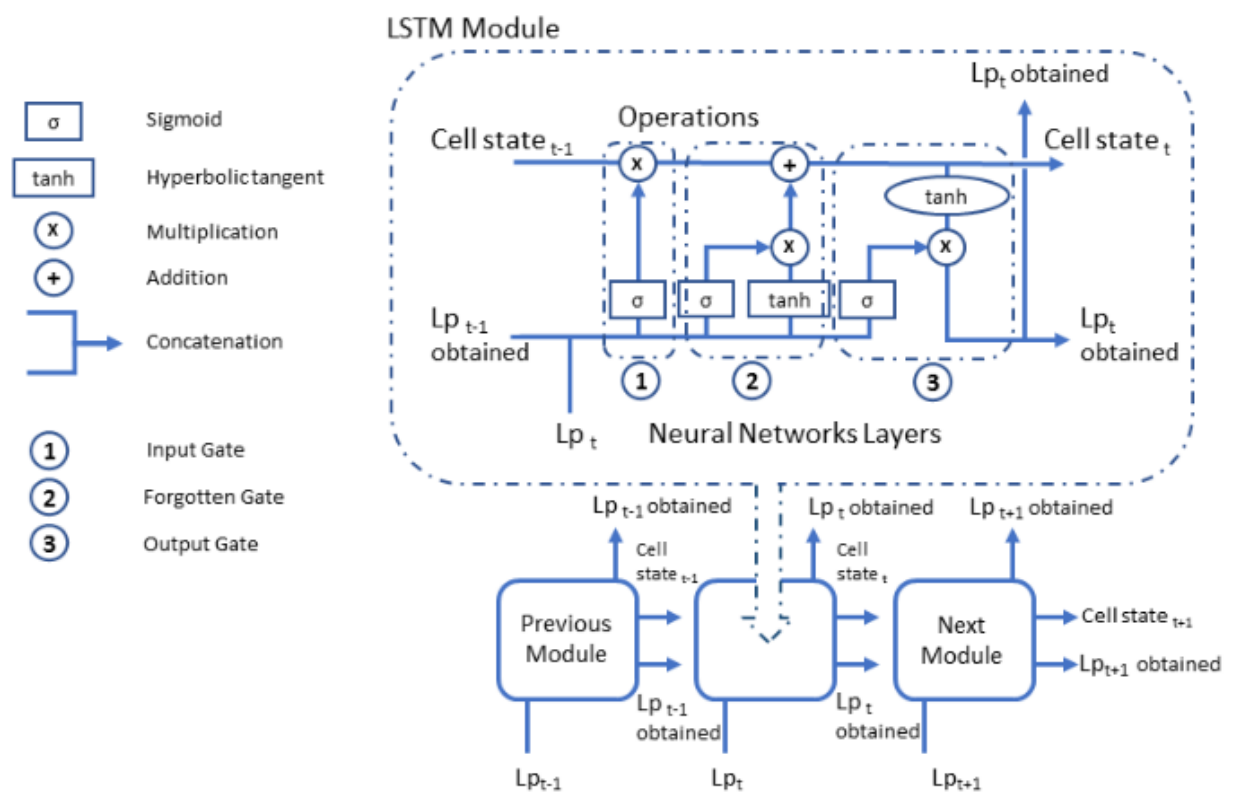
Основні характеристики RNN включають:

1. *Рекурентні зв'язки.* RNN має внутрішні стани, які можуть передавати інформацію з попередніх моментів часу в поточний. Це дозволяє моделі враховувати контекст і динаміку в часі.
2. *Параметризація ваги за часом.* У ваги RNN вводяться параметри за часом, що дозволяє їм змінюватися в залежності від певного моменту часу. Це робить RNN придатною для обробки послідовностей.
3. *Шари повторюваності.* Одна і та ж сама мережа використовується для обробки кожного елемента в послідовності. Це забезпечує об'єднання знань, отриманих з різних моментів часу.

Проте, класичні RNN мають певні обмеження, такі як проблема зниклих градієнтів, яка може призводити до труднощів у тренуванні на довгих послідовностях.

У зв'язку з цим були розроблені модифікації RNN, такі як Long Short-Term Memory Networks (LSTM) та Gated Recurrent Units (GRU).

Long Short-Term Memory Networks (LSTM). Вони мають додаткові структури пам'яті, які дозволяють моделі зберігати та використовувати інформацію на довший термін.



Риснок 2.1 – Загальна схема довгострокової мережі пам'яті [17]

Це допомагає уникнути проблеми зниклих градієнтів.

Основні особливості LSTMs включають:

1. *Комірка пам'яті (Cell State)*. Одна з основних ідей LSTMs - це введення комірки пам'яті, яка може зберігати інформацію та взаємодіяти з іншими частинами мережі. Це дозволяє LSTM зберігати та вибирати, яку інформацію важливо запам'ятовувати або забувати.

2. *Ворота (Gates)*. LSTMs використовують три типи воріт для контролю потоку інформації: ворота забування (Forget Gate), ворота вводу (Input Gate) і ворота виводу (Output Gate). Кожне ворота має свою вагу, яка визначає, яка частина інформації повинна передаватися чи забуватися.

Розглянемо три типи воріт більш детально.

«Ворота забування» або «*Forget Gate*» - це поняття, що використовується в контексті рекурентних нейронних мереж (RNN) і, зокрема, в рекурентних мережах довготривалої пам'яті (LSTM).

У контексті LSTM, що є спеціальним типом RNN, «ворота забування» відповідає за те, яку частину інформації з попереднього кроку часу слід забути чи проігнорувати. Це дозволяє мережі взаємодіяти з довгостроковою пам'яттю, визначаючи, яку інформацію важливо зберігати для майбутніх прогнозів чи обчислень, а яку можна вважати неактуальною або несуттєвою.

Ворота забування в LSTM мережі вирішують проблему зникнення та вибухання градієнтів, що можуть виникнути при тренуванні глибоких рекурентних мереж. Вони регулюють, яка інформація повинна передаватися через часові кроки, і це дозволяє моделі краще пристосовуватися до завдань, пов'язаних з послідовністю даних [18].

Узагальнюючи, ворота забування допомагають LSTM-мережам ефективніше взаємодіяти з послідовністю даних, забезпечуючи баланс між зберіганням та видаленням інформації з довгострокової пам'яті.

В «рекурентних мережах довготривалої пам'яті» (LSTM), «ворота вводу» або «*Input Gate*» - це інша важлива компонента, яка регулює, яка інформація повинна бути додана до довгострокової пам'яті на кожному кроці часу.

Input Gate контролює, яка частина нових даних повинна бути додана до поточного стану довготривалої пам'яті. Це важливо для того, щоб модель могла враховувати нові вхідні дані та оновлювати свій стан. Input Gate працює як механізм фільтрації, який визначає, яка інформація є важливою для збереження.

Робота Input Gate включає в себе використання сигмоїдальної функції для визначення того, яка частина нових даних повинна бути додана до поточного стану. Сигмоїдальна функція генерує значення між 0 і 1 для кожного елемента нових даних, і це значення визначає, наскільки цей елемент важливий для додавання до довгострокової пам'яті.

Input Gate, спільно з Forget Gate і Output Gate, ускладнює структуру LSTM мережі, дозволяючи їй ефективно взаємодіяти з послідовністю даних та уникати проблем, таких як зникнення та вибухання градієнтів при тренуванні.

«Ворота виводу» або «Output Gate» - це інша ключова складова рекурентних мереж довготривалої пам'яті (LSTM). Ця частина мережі визначає, яка інформація з поточного стану довготривалої пам'яті повинна бути виведена як вихід моделі.

Output Gate використовує сигмоїдальну функцію та тангенс гіперболічний (\tanh) для визначення вмісту поточного стану, який буде виведений. Вони працюють як механізми вагової корекції, дозволяючи моделі фільтрувати та контролювати, яка інформація повинна передаватися на вихід.

Процес виведення може бути описаний наступним чином:

1. *Сигмоїдальна функція*. Генерує значення між 0 і 1 для кожного елемента поточного стану. Ці значення визначають, яка частина інформації повинна бути виведена.
2. *Тангенс гіперболічний (\tanh)*. Генерує значення між -1 і 1 для кожного елемента поточного стану. Ці значення представляють саму інформацію, яка готується до виведення.
3. *Поелементне множення*. Результати сигмоїдальної функції і тангенсу гіперболічного множаться, щоб отримати вивід, який буде переданий наступному шару мережі або використовується для обчислення остаточного виводу моделі.

Ворота виводу дозволяють моделі визначати, яка частина довгострокової пам'яті є важливою для виведення, і це допомагає моделі

навчатися робити змірні та збагачені висновки на основі введених даних та їх історії.

4. *Функції активації*. LSTMs використовують функції активації для регулювання виходу. Зазвичай використовуються сигмоїдальна функція для воріт і гіперболічний тангенс для комірки пам'яті.

5. *Повторювачі (Recurrent Connections)*. LSTMs використовують повторювачі для передачі інформації з попереднього часового кроку в поточний. Однак, завдяки використанню воріт, вони можуть контролювати, яка частина інформації буде передана.

Поняття повторювачів (recurrent connections) в мережах нейронів представляє собою важливий аспект, який сприяє ефективності та гнучкості в розв'язанні різноманітних завдань. Ці повторювачі дозволяють передавати інформацію в часі, надаючи нейронним мережам здатність аналізувати та враховувати контекст у динамічних середовищах.

Однією з ключових переваг повторювачів є їхня здатність зберігати попередні стани внутрішніх представлень. У звичайних фіксованих архітектурах, таких як звичайні шари нейронів, інформація про вхідні дані обробляється без будь-якого врахування попередніх станів. Повторювачі дозволяють мережі зберігати та використовувати інформацію з минулих моментів, що особливо корисно в аналізі послідовних даних, таких як мовлення, часові ряди або текст.

Ще однією важливою характеристикою повторювачів є їхня здатність вирішувати завдання з динамічною довжиною вхідних послідовностей. У той час як звичайні нейронні мережі мають фіксований розмір вхідних даних, повторювачі можуть ефективно працювати з послідовностями різної довжини. Це зроблено за рахунок того, що кожен елемент послідовності оброблюється по черзі, а не одразу, надаючи гнучкість та універсальність в розв'язанні різних завдань.

Однак, разом із зазначеними перевагами, повторювачі також мають свої виклики. Проблема зникнення та вибуху градієнта – це один з основних

недоліків повторювачів, який може виникнути при тренуванні глибоких мереж. Розв'язання цих проблем включає в себе використання різних архітектур повторювачів, таких як LSTM (Long Short-Term Memory) або GRU (Gated Recurrent Unit), які враховують ці аспекти та зменшують ймовірність виникнення проблем з градієнтом.

Отже, повторювачі – це важливий елемент в арсеналі інструментів у галузі глибокого навчання. Вони відкривають нові можливості для розв'язання завдань, пов'язаних з послідовностями, та дозволяють моделям адаптуватися до змін в часі. Незважаючи на труднощі, пов'язані з тренуванням глибоких мереж із повторювачами, їхні переваги роблять їх важливим інструментом у сфері штучного інтелекту та машинного навчання.

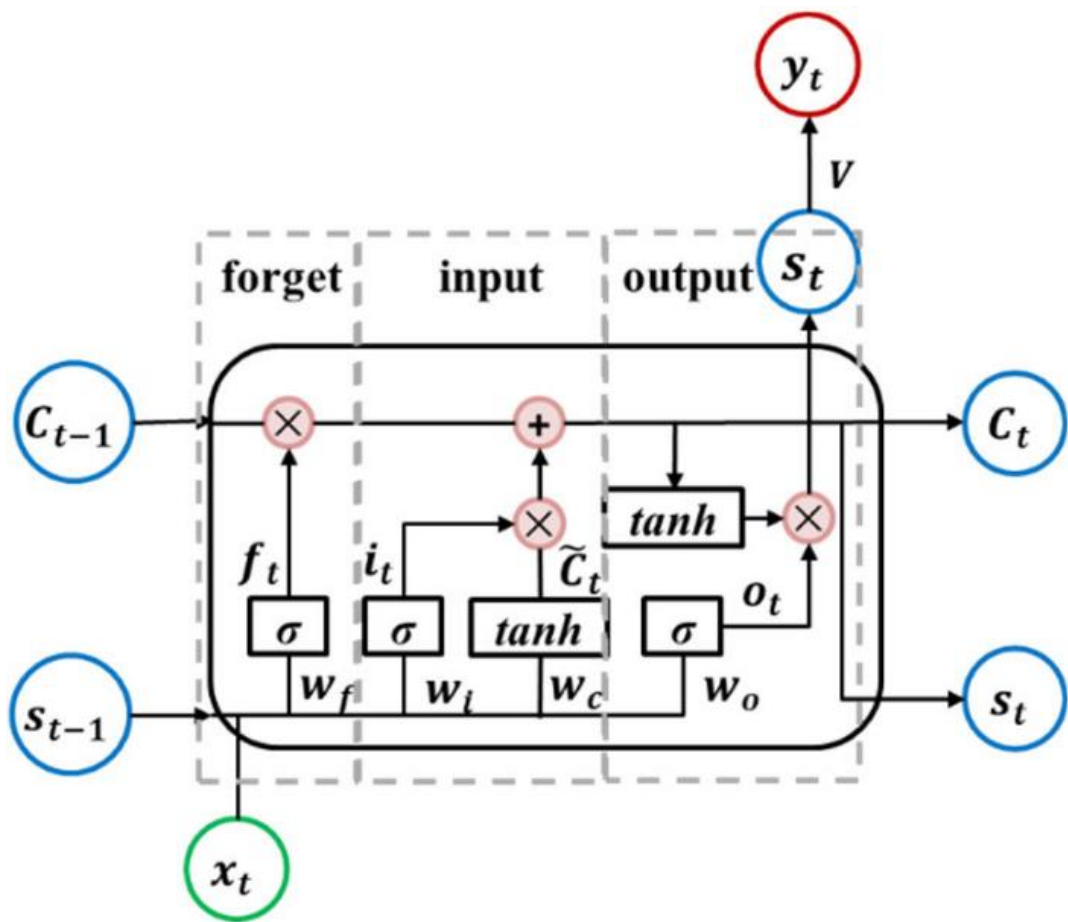
5. *Вхідний та вихідний шар*. LSTMs мають вхідний та вихідний шар, які взаємодіють з коміркою пам'яті та воротами [18].

Процес роботи LSTM можна розділити на кілька кроків:

1. *Забування (Forget)*. Ворота забування вирішує, яку інформацію слід зберегти чи відкинути з комірки пам'яті.
2. *Введення (Input)*. Ворота вводу визначають, яку нову інформацію слід додати до комірки пам'яті.
3. *Оновлення (Update)*. Комірка пам'яті оновлюється, враховуючи забуту інформацію та нові дані.
4. *Вихід (Output)*. Ворота виводу визначають, яку інформацію вибрати з комірки пам'яті для передачі на вихід.

LSTMs дозволяють ефективно моделювати послідовності з тривалими залежностями, що робить їх особливо корисними для задач, таких як розпізнавання мови, переклад та аналіз емоцій на зображеннях та відео.

Gated Recurrent Units (GRU). Це інша модифікація RNN, яка вирішує деякі недоліки оригінальної архітектури. GRU має менше параметрів, але зберігає схожі можливості враховувати контекст в часі.



Риснок 2.2 – Закриті рекурентні одиниці (GRU) [10]

Основні характеристики GRU включають:

1. *Вхідний та вихідний стани.* GRU також має внутрішній стан (hidden state), але він не розділяється на комірку пам'яті та вихідний стан, як у LSTM. Замість цього, в GRU використовується один вектор стану.
2. *Ворота оновлення та ворота скидання.* GRU використовує два ворота - ворота оновлення (Update Gate) і ворота скидання (Reset Gate). Ворота оновлення регулює, як ми можемо оновлювати стан, а ворота скидання визначає, яку частину попереднього стану слід ігнорувати.
3. *Функція активації.* GRU використовує гіперболічний тангенс як функцію активації для оновлення стану та скидання.
4. *Мінімальна кількість параметрів.* Один з відзначених плюсів GRU в порівнянні з LSTM - це менша кількість параметрів. GRU має менше ваг та операцій, що робить його більш обчислювально ефективним.

Процес роботи GRU може бути розділений на наступні кроки:

1. *Ворота скидання (Reset Gate)*. Визначає, яку частину попереднього стану слід ігнорувати.
2. *Ворота оновлення (Update Gate)*. Регулює, як ми можемо оновлювати стан.
3. *Оновлення стану*. Визначається новий стан за допомогою воріт та попереднього стану.

GRU дозволяє моделювати довгострокові залежності в послідовностях та зазвичай використовується в тих випадках, коли важлива обчислювальна ефективність, або коли обсяг даних обмежений.

RNN і його модифікації використовуються у різних задачах, включаючи розпізнавання мови, машинний переклад, аналіз часових рядів, в тому числі і для розпізнавання емоцій на зображеннях та відео. Вони дозволяють моделі розуміти контекст і залежності в часі, що робить їх корисними у великому спектрі завдань, де важлива динаміка даних.

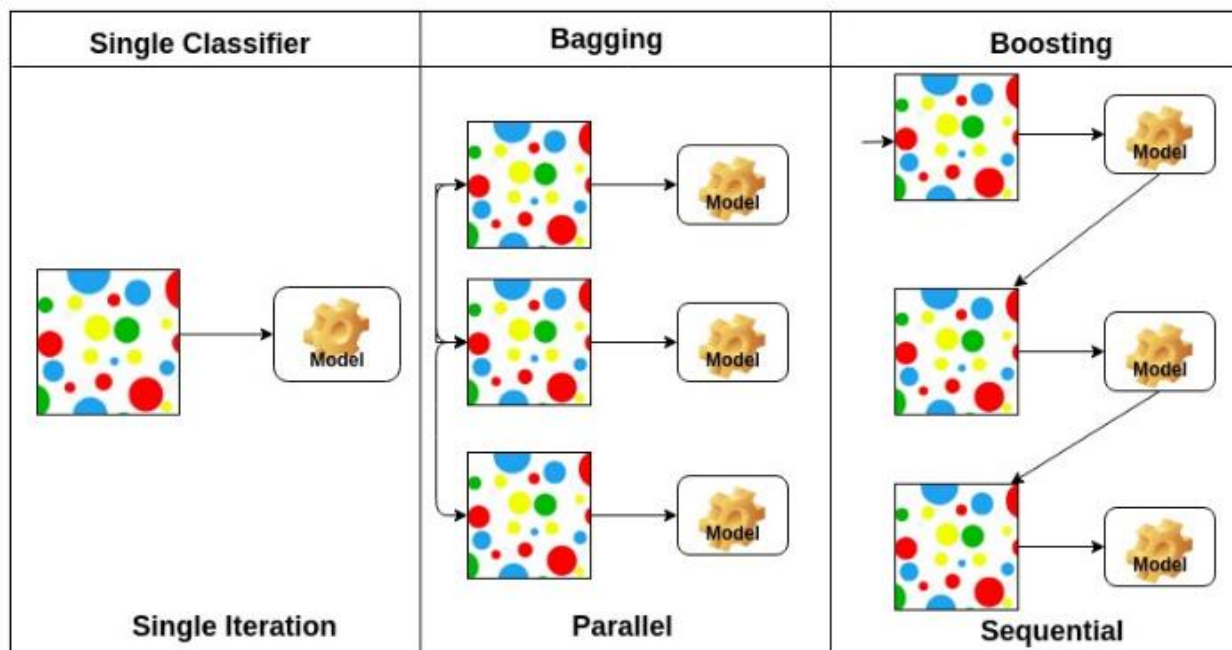
Використання рекурентних нейронних мереж для цієї задачі може враховувати динаміку зміни емоцій в часі на відео та у зображеннях, дозволяючи системі ефективно виявляти та класифікувати різні емоційні стани.

2.2 Алгоритм AdaBoost

Алгоритм AdaBoost (Adaptive Boosting) є алгоритмом машинного навчання, який використовується для підвищення точності моделей класифікації. Він був запропонований Йоафом Фрейдманом, Робертом Шапіро та Пітером Бустінгом в 1996 році. AdaBoost є адаптивним алгоритмом, що об'єднує декілька слабких класифікаторів для створення сильного класифікатора.

Алгоритм AdaBoost (Adaptive Boosting) – це ансамблевий метод машинного навчання, який використовується для покращення результатів

базових моделей (званих також «слабкими учасниками») шляхом надання їм ваг в залежності від їхньої точності.



Риснок 2.3 – Адаптивне посилення [11, с. 78]

Ось деякі переваги алгоритму AdaBoost:

1. *Висока точність.* AdaBoost відомий своєю здатністю досягати високої точності класифікації. Це досягається завдяки важливості, яку алгоритм приділяє правильно класифікованим прикладам під час тренування.

2. *Адаптивність.* AdaBoost є адаптивним методом, що дозволяє підвищувати вагу помилково класифікованих прикладів, надаючи більше уваги складним елементам даних.

3. *Використання слабких учасників.* AdaBoost може використовувати слабкі учасники, тобто моделі, які можуть бути трохи кращими, ніж випадковий вибір. Це робить його ефективним для поєднання простих моделей в один сильний класифікатор.

4. *Зменшення ризику перенавчання.* AdaBoost має тенденцію менше схильний до перенавчання, порівняно з деякими іншими методами, завдяки використанню ваг та фокусу на найбільш складних прикладах.

5. *Широкий спектр застосувань.* AdaBoost може бути використаний для різноманітних задач, таких як класифікація та регресія. Він успішно використовується в багатьох галузях, включаючи комп'ютерний зір, обробку природної мови та інші.

6. *Відсутність важливих гіперпараметрів.* AdaBoost не має багато гіперпараметрів, які потрібно налаштовувати, що полегшує його використання та конфігурацію.

Хоча AdaBoost має свої переваги, важливо також враховувати, що він може бути чутливим до шуму в даних та викидів, і в окремих випадках йому може бути важко адаптуватися до дуже складних залежностей в даних.

Основні принципи роботи алгоритму AdaBoost:

1. *Слабкий класифікатор.* AdaBoost використовує так звані «слабкі» класифікатори, які мають невелику точність навіть на випадковому виборі. Зазвичай це може бути дерево рішень (наприклад, дерево глибини 1, також відоме як «пень» - stump) або інші прості моделі.
2. *Навчання за допомогою ваг.* AdaBoost приділяє вагу кожному екземпляру даних у навчальному наборі. Спочатку всі ваги встановлюються рівними, а під час навчання ваги змінюються так, щоб акцентувати увагу на помилково класифікованих екземплярах.
3. *Послідовне навчання.* AdaBoost навчає класифікатори послідовно. Кожен новий класифікатор створюється з урахуванням результатів попередніх. Екземпляри, які були помилково класифіковані попередніми класифікаторами, отримують більшу вагу, щоб їх важливість зростала.
4. *Голосування з вагою.* Після навчання всі класифікатори об'єднуються в сильний класифікатор за допомогою голосування з вагою. Класифікатори, які мають менше помилок, мають більший вплив на фінальне рішення.

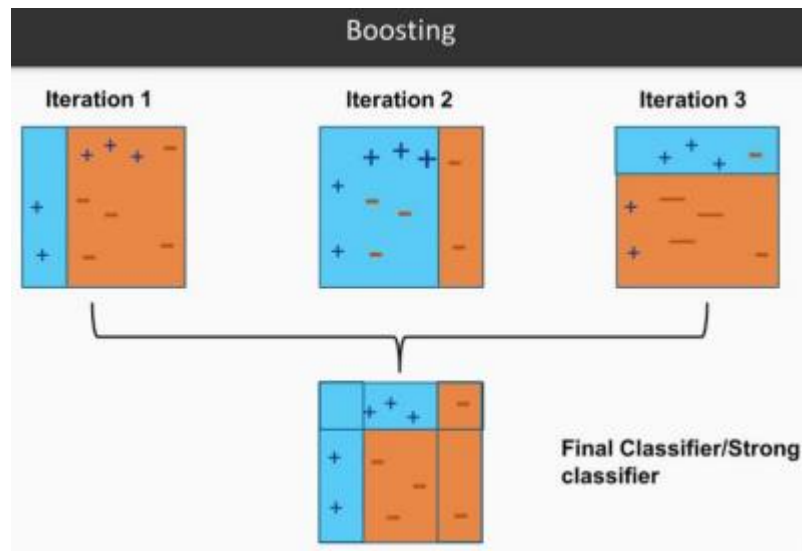
Алгоритм AdaBoost може бути описаний наступним чином:

1. *Ініціалізація ваг.* Кожен екземпляр у навчальному наборі починається з однаковою вагою.

2. *Навчання класифікатора.* Додається новий слабкий класифікатор, який спрямований на зменшення помилок на попередніх ітераціях.
3. *Оцінка точності.* Обчислюється точність класифікатора на навчальному наборі, з урахуванням ваги кожного екземпляру.
4. *Обчислення ваги класифікатора.* Визначається вага класифікатора з урахуванням його точності.
5. *Оновлення ваг екземплярів.* Збільшуються ваги помилково класифікованих екземплярів для надання їм більшого впливу на наступну ітерацію.
6. *Повторення.* Процес повторюється досягнення певної кількості класифікаторів або до досягнення задовільної точності.

Після навчання, класифікатор AdaBoost може використовуватися для класифікації нових прикладів за допомогою голосування всіх слабких класифікаторів з вагою.

Об'єднання слабких класифікаторів в сильний виглядає наступним чином (див. рис. 2.4) .



Риснок 2.4 – Об'єднання слабких класифікаторів в сильний [11]

Сильний класифікатор AdaBoost виражається наступною формулою:

$$F(x) = \text{sign}\left(\sum_{t=1}^T a_t h_t(x)\right)$$

де:

$F(x)$ - фінальне рішення сильного класифікатора для вхідного прикладу x ,

T - кількість слабких класифікаторів,

a_t - вага слабкого класифікатора $h_t(x)$ в голосуванні,

sign - функція знаку, яка повертає 1 для додатних значень та -1 для від'ємних.

Кожен слабкий класифікатор $h_t(x)$ вирішує конкретну задачу класифікації та повертає -1 або 1.

Вага a_t визначається на основі точності класифікатора, де вага збільшується для точніших класифікаторів і зменшується для менш точних.

Сильний класифікатор об'єднує рішення всіх слабких класифікаторів за допомогою лінійної комбінації з вагами a_t . Це дозволяє вагомо враховувати внесок кожного слабкого класифікатора в фінальне рішення.

Мета методу AdaBoost – акцентувати увагу на певних ознаках шляхом активного оновлення їх ваг, з метою максимізації ймовірності правильної класифікації шуканого об'єкта. Тим часом непотрібні або малоефективні ознаки Хаара вилучаються, сприяючи ефективнішому процесу виявлення об'єкта. Ознаки, які успішно призводять до істинних-позитивних та істинно-негативних класифікацій, отримують більші ваги, оскільки є ймовірніше виявити шуканий об'єкт. З іншого боку, ознаки, що часто призводять до помилково-позитивних і помилково-негативних класифікацій, отримують менші ваги, існує навіть можливість присвоєння їм ваги нуль.

Після визначення сильних класифікаторів використовується техніка каскадних класифікаторів. Ця стратегія включає етапове застосування наборів класифікаторів для виявлення певних рис обличчя. На ранніх етапах використовуються простіші ознаки Хаара та відповідні класифікатори. На пізніших етапах, які включають складніші ознаки, аналізуються менші деталі обличчя чи його рис. Якщо на початковому етапі виявлено відсутність об'єкта у певній області, алгоритм пропускає цю область і переходить до наступної,

що значно економить час, оскільки зайві області не обробляються на більшості етапів.

Обробка другого етапу розпочнеться лише тоді, коли об'єкти на першому етапі будуть виявлені на зображенні. Процес розпізнавання триватиме до завершення усіх етапів, після чого об'єкт буде класифіковано як обличчя. У цьому розділі був описаний алгоритм AdaBoost, який використовується для покращення класифікації рис обличчя. Цей алгоритм є необхідною частиною методу Віоли-Джонса, використовуючи ознаки Хаара, та дозволяє ефективно розпізнавати об'єкти з мінімальною похибкою. Саме тому він є одним із найпоширеніших методів розпізнавання обличчя, а, відповідно, і емоцій.

2.3 Гістограма напрямлених градієнтів

Визначення емоцій на зображенні є останнім кроком який включає в себе виявлення ключових точок обличчя та визначення конкретної емоції.

Для досягнення цієї мети використовується дескриптор, який може ідентифікувати точки рис обличчя та аналізувати емоційний стан.

Зокрема, для виявлення ключових точок застосовується метод гістограм напрямлених градієнтів.

Гістограма напрямлених градієнтів (Histogram of Oriented Gradients, HOG) є методом в комп'ютерному зорі та обробці зображень для виявлення об'єктів або структур на зображенні, зокрема, ключових точок обличчя.

Ось деякі переваги використання HOG:

1. Детектування країв та текстур.

В області комп'ютерного зору та обробки зображень однією з ключових задач є виявлення об'єктів та їх характеристик на зображеннях.

Детектування країв та текстур є важливим етапом у багатьох візуальних задачах, таких як розпізнавання облич, аналіз медичних зображень, або автономне керування транспортними засобами. Одним із ефективних методів

виявлення об'єктів є використання історично важливого підходу - Histogram of Oriented Gradients (HOG).

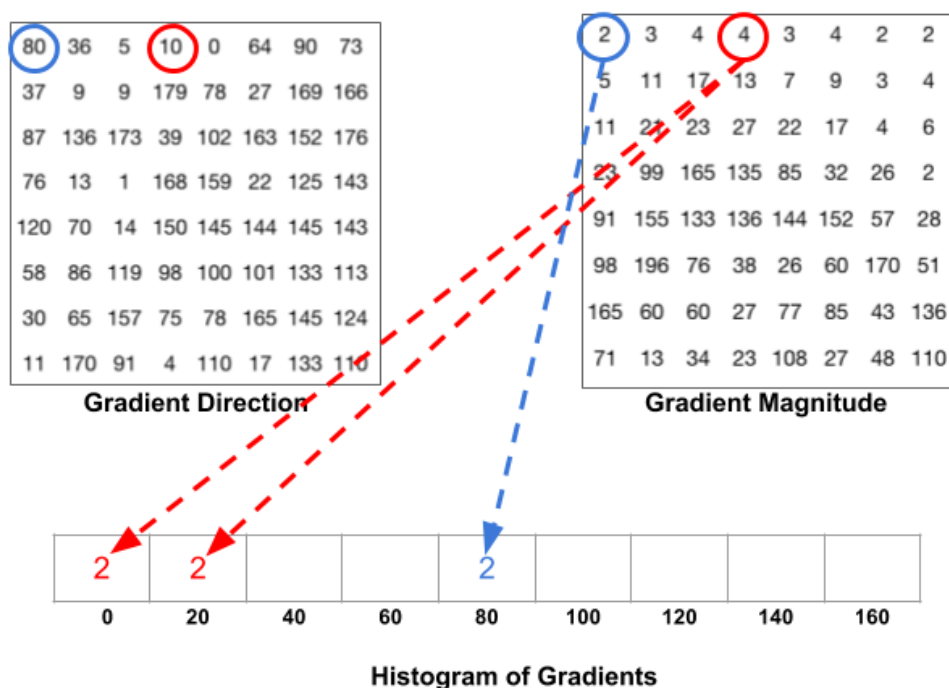


Рисунок 2.5 – Гістограма орієнтованих градієнтів [28]

HOG - це метод, який став популярним у 2005 році завдяки роботі Навіджа та Далалі у контексті розпізнавання людських облич. Однак відтоді його успішно використовують у різних областях. Цей метод базується на тому, що структура об'єктів може бути визначена за допомогою градієнтів яскравості на зображенні.

Суть методу полягає в наступних кроках:

1. *Градiєнтні зображення.* Спочатку обчислюються градієнти яскравості на зображенні. Це дозволяє визначити, як швидко змінюється яскравість в кожній точці зображення та в якому напрямку.
2. *Кутова гістограма орієнтацій (Histogram of Oriented Gradients).* За допомогою градієнтів формується гістограма орієнтацій, яка показує, як часто градієнти мають певний кут. Це дозволяє визначити основні напрямки текстур та країв.

3. *Створення вектора ознак.* Інформацію з гістограм орієнтацій об'єднують у вектор ознак для кожного блоку чи клітини зображення.
4. *Нормалізація вектора ознак.* Щоб зробити метод менш чутливим до змін яскравості, використовують нормалізацію вектора ознак.
5. *Сканування зображення.* Проводиться сканування зображення з використанням створеного вектора ознак для виявлення об'єктів з певними характеристиками.

Метод HOG дозволяє ефективно виявляти краї та текстури, забезпечуючи високу точність та стійкість до змін в умовах освітлення. Він також широко використовується у задачах визначення облич, виявленні об'єктів на великих відстанях та в багатьох інших візуальних завданнях.

Однак, важливо зазначити, що використання HOG може бути обмеженим у деяких випадках, наприклад, коли об'єкти мають складні текстурні або колірні характеристики. Також, при роботі з великою кількістю даних, обробка може вимагати значних обчислювальних ресурсів [17, с.266].

У підсумку, детектування країв та текстур за допомогою HOG - це потужний підхід, який знаходить своє застосування у багатьох візуальних задачах, забезпечуючи високу ефективність та стійкість до різних умов.

2. Інваріантність до змін масштабу та освітлення

Однією з ключових викликів в обробці зображень є забезпечення інваріантності до змін масштабу та освітлення для ефективного виявлення об'єктів. У цьому контексті Histogram of Oriented Gradients (HOG) виявляється потужним методом, який дозволяє досягти цієї інваріантності та забезпечує надійне виявлення країв та текстур.

Масштаб та освітлення є ключовими факторами, які можуть впливати на якість обробки зображень та виявлення об'єктів. Зміна масштабу може призвести до того, що об'єкти будуть виглядати іншими відносно їх реального розміру, а різниця в освітленні може змінювати яскравість та контрастність на зображенні. HOG-підхід вирішує ці проблеми, забезпечуючи інваріантність та стабільність в процесі виявлення об'єктів.

Основна ідея HOG полягає у використанні локальних градієнтів яскравості для визначення структури об'єктів на зображенні. Процес обчислення градієнтів та створення гістограми орієнтацій дозволяє визначити напрямки текстур та країв у різних частинах зображення. Це важливо не тільки для точного виявлення об'єктів, але і для того, щоб зробити метод стійким до змін масштабу та освітлення.

Інваріантність до змін масштабу досягається завдяки тому, що HOG визначає структуру об'єктів не в абсолютних розмірах, а відносно їхнього локального контексту. Таким чином, об'єкти розпізнаються на основі їхньої локальної текстурної інформації, а не абсолютних розмірів, що дозволяє зберігати інваріантність до масштабу.

Щодо інваріантності до освітлення, HOG виявляється менш чутливим завдяки використанню градієнтів яскравості, які описують структуру об'єкта, а не його абсолютний рівень освітлення. Таким чином, коливання в яскравості на зображенні не суттєво впливають на виявлення об'єктів.

Завдяки цим особливостям, HOG-підхід виявляється ефективним і стійким методом для виявлення країв та текстур, незалежно від змін масштабу та освітлення. Він широко використовується в різних областях, таких як розпізнавання облич, виявлення об'єктів на зображеннях в реальному часі, та в інших візуальних завданнях, де інваріантність до змін умов грає важливу роль.

Отже, HOG може бути відносно інваріантним до змін масштабу та освітлення. Це означає, що він може ефективно працювати з об'єктами різного розміру та освітлення.

3. Використання в об'єктному визначенні.

Об'єктне визначення, або об'єктне виявлення, є ключовою складовою великої кількості візуальних та комп'ютерно-зорових систем, зокрема у сферах розпізнавання облич, виявлення об'єктів та безпілотних систем. Одним

із ефективних методів об'єктного визначення є використання Histogram of Oriented Gradients (HOG).

HOG виявляється особливо потужним у виявленні об'єктів на зображеннях завдяки його здатності розпізнавати текстури та краї на основі градієнтів яскравості. У контексті об'єктного визначення, HOG використовується для створення ознакового вектора, який подає інформацію про напрям та інтенсивність градієнтів в кожній частині зображення.

Відзначається його великою універсальністю та ефективністю в різних умовах, а також здатністю до виявлення об'єктів різних форм та розмірів. Однією з ключових переваг використання HOG у сфері об'єктного визначення є інваріантність до змін масштабу та освітлення.

Це дозволяє ефективно виявляти об'єкти незалежно від їхнього розміру чи умов освітлення, що важливо в реальних умовах застосування, де об'єкти можуть знаходитися в різних масштабах і піддаватися змінам у світлових умовах.

Крім того, HOG забезпечує робастність до змін у фоновому оточенні, що дозволяє ефективно виділяти об'єкти на зображеннях, навіть якщо вони розташовані на зрушеному або шумному фоні. Це робить його особливо корисним для завдань виявлення об'єктів в умовах, де фон може бути складним чи змінюваним.

Важливо зазначити, що HOG залишається популярним методом в області об'єктного визначення навіть у зв'язку з розвитком глибокого навчання та нейронних мереж. Його ефективність і стійкість до різних умов роблять його конкурентоздатним і високоефективним інструментом для визначення об'єктів у великому спектрі застосувань.

У висновку, використання HOG в об'єктному визначенні демонструє його високу ефективність у виявленні об'єктів навіть в умовах зміни масштабу та освітлення. Його універсальність, інваріантність та здатність до роботи в реальних умовах роблять його важливим інструментом для розв'язання завдань обробки зображень та комп'ютерного зору.

Отже, HOG широко використовується в системах об'єктного визначення для розпізнавання облич, транспортних засобів, людей та інших об'єктів.

4. Застосування в системах відеоспостереження.

Системи відеоспостереження стають невід'ємною частиною сучасного світу, забезпечуючи безпеку, контроль та відслідковування подій в реальному часі. Однак зростання обсягу відеоданих ставить великі виклики перед системами обробки та аналізу зображень. У цьому контексті метод Histogram of Oriented Gradients (HOG) виявляється потужним інструментом для обробки великих обсягів відеоданих у системах відеоспостереження.

Однією з основних задач відеоспостереження є виявлення та відслідковування об'єктів на відеозаписах. HOG використовується для ефективного виявлення об'єктів на основі їх текстурних та краєвих характеристик. Основні переваги використання HOG в системах відеоспостереження включають:

1. *Інваріантність до змін масштабу та освітлення.* Системи відеоспостереження можуть стикатися з різними умовами освітлення та змінами в масштабі об'єктів на зображенні. HOG виявляється інваріантним до таких змін, що забезпечує надійне виявлення об'єктів навіть при змінних умовах.
2. *Робастність до фонових умов.* В системах відеоспостереження часто необхідно виділяти об'єкти на забруднених або шумних фонових зображеннях. HOG дозволяє ефективно виявляти об'єкти навіть на складних фонових зображеннях та в умовах, коли об'єкти можуть бути затушовані або перекриті іншими об'єктами.
3. *Висока точність виявлення об'єктів.* HOG став популярним методом завдяки своїй високій точності виявлення об'єктів, особливо тих, що мають різні форми та текстурні характеристики. Це робить його популярним в системах відеоспостереження для розпізнавання облич, виявлення транспортних засобів та інших об'єктів.

4. *Робота в реальному часі.* Деякі системи відеоспостереження потребують обробки відеоданих в реальному часі, і тут НОГ виявляється ефективним методом, оскільки його обчислення може бути оптимізовано для швидкої роботи.

Розглянемо питання адаптації до реального часу. НОГ виявляється оптимальним в контексті систем відеоспостереження, оскільки його розрахунок може бути оптимізований для швидкої обробки даних у реальному часі. Це надає можливість системам миттєво реагувати на зміни в навколишньому середовищі та події, що робить його ключовим компонентом у нинішній динамічній сфері відеонагляду.

Також варто відзначити, що гнучкість НОГ полягає у можливості виявлення різних типів об'єктів. Його застосування не обмежується лише конкретним класом об'єктів, що дає можливість використовувати цей метод для виявлення та відстеження різноманітних об'єктів, забезпечуючи високу універсальність і гнучкість.

Можливість виявлення об'єктів різного типу: НОГ не обмежується конкретним класом об'єктів і може бути використаний для виявлення різних об'єктів, що робить його гнучким у застосуванні. Цей потужний алгоритм стає необхідною складовою сучасних систем безпеки та відеонагляду, надійно забезпечуючи високу точність і стійкість в найрізноманітніших умовах.

Загалом, використання НОГ в системах відеоспостереження розширює їхні можливості виявлення та відслідковування об'єктів у реальному часі, забезпечуючи високу точність та робастність в різних умовах. Його потужний алгоритм робить його невід'ємною частиною сучасних систем безпеки та відеоспостереження.

Отже, використання НОГ дозволяє ефективно виявляти та відстежувати об'єкти в системах відеоспостереження, що є важливим для ряду застосувань, таких як безпека та визначення осіб.

5. *Висока робастність.*

Histogram of Oriented Gradients (HOG) є одним із визнаних методів у галузі обробки зображень та комп'ютерного зору, особливо в контексті виявлення об'єктів. Його висока робастність робить його ефективним інструментом для широкого спектру завдань, включаючи розпізнавання облич, виявлення об'єктів та інші сценарії візуальної обробки.

Однією з ключових особливостей, що робить HOG робастним, є його інваріантність до змін масштабу та освітлення. Зміна масштабу часто ускладнює завдання виявлення об'єктів, оскільки вони можуть з'являтися на зображенні в різних розмірах. HOG вирішує цю проблему, оскільки його вимірювання базуються на локальних текстурних характеристиках, які залишаються стабільними відносно змін масштабу.

Щодо інваріантності до освітлення, HOG використовує градієнти яскравості, які вказують на структуру об'єктів, а не на абсолютні рівні освітлення. Такий підхід дозволяє методу ефективно фільтрувати зміни в яскравості, що забезпечує стабільність в різних умовах освітлення.

Іншою важливою характеристикою робастності HOG є його здатність розпізнавати об'єкти в умовах затемнення чи перекриття. За допомогою визначення градієнтів та їхньої орієнтації, метод здатен ефективно виявляти краї та текстури об'єктів, навіть якщо частини їхніх контурів приховані або тіньовані.

Додатково, HOG проявляє високу точність та ефективність в реальному часі, що робить його популярним для застосування в системах відеоспостереження, роботі з рухомими об'єктами та інших завдань, де швидкість та надійність є ключовими факторами.

У підсумку, висока робастність HOG робить його важливим інструментом в обробці зображень та комп'ютерному зорі. Використовуючи його особливості, такі як інваріантність та стійкість до змін, HOG дозволяє ефективно вирішувати виклики в областях, де визначення об'єктів та їхніх характеристик має високий практичний значення.

Отже, HOG є відносно робастним до шуму та змішаних фонових умов. Це робить його ефективним для використання в реальних умовах, де може бути багато випадкових елементів.

б. Низька вимогливість до обчислювальних ресурсів.

У світі стрімкого розвитку обчислювальних технологій та великих обсягів даних, важливо розглядати ефективність та вимоги до обчислювальних ресурсів при застосуванні методів обробки зображень. Histogram of Oriented Gradients (HOG) становить чудовий приклад технології, яка, з одного боку, демонструє високий рівень точності та ефективності, а з іншого - відзначається низькими вимогами до обчислювальних ресурсів.

Однією з ключових переваг HOG є його спроможність до роботи на різних обчислювальних пристроях, включаючи прості вбудовані системи, мобільні пристрої та стандартні персональні комп'ютери. Це зумовлено, в першу чергу, тим, що HOG базується на локальних градієнтах яскравості, що дає можливість ефективно визначати текстурні та краєві характеристики об'єктів, не вимагаючи великої кількості обчислень.

Важливою особливістю, яка забезпечує низьку вимогливість до обчислювальних ресурсів, є використання гістограм орієнтацій. Градієнти яскравості обчислюються для локальних областей зображення, і потім ці дані групуються в гістограму для кожного блоку. Це дозволяє значно зменшити обсяг інформації та скоротити обчислювальну складність, залишаючи при цьому важливі текстурні та краєві характеристики.

Також, HOG демонструє ефективність у виявленні об'єктів на різних масштабах. Замість використання подібних алгоритмів, які вимагають обчислення великої кількості фільтрів на різних розмірах, HOG використовує градієнтні інформаційні блоки, що робить його менш обчислювально витратним.

Застосування HOG у системах відеоспостереження та в області автономного транспорту, де ресурси обчислювального обладнання можуть бути обмеженими, підкреслює важливість його низької вимогливості до

обчислювальних ресурсів. Можливість працювати на простій апаратурі водночас забезпечуючи високий рівень точності робить НОГ привабливим вибором для широкого спектру завдань обробки зображень в умовах обмежених обчислювальних ресурсів.

У підсумку, Histogram of Oriented Gradients вражає своєю здатністю до високоефективної обробки зображень при відносно низьких вимогах до обчислювальних ресурсів. Ця характеристика визначає його успішне використання у сферах, де обмеженість ресурсів є ключовим фактором.

Отже, НОГ може працювати на відносно низьких обчислювальних ресурсах, що полегшує його використання на вбудованих системах та мобільних пристроях.

Загалом, НОГ є дуже корисним методом для визначення характеристик об'єктів на зображеннях, і він знаходить широке застосування в області комп'ютерного зору та обробки зображень.

Нижче подано детальний розгляд принципів роботи НОГ.

1. Градієнтні зображення.

Градієнтні зображення в НОГ є ключовим елементом, який дозволяє виявляти та описувати структуру об'єктів у вигляді гістограм орієнтацій.

Основна ідея полягає у використанні градієнтів яскравості, які вказують на напрямок та інтенсивність змін яскравості на зображенні. Процес визначення градієнтів включає обчислення похідних за яскравістю відносно пікселів зображення в областях зазначеного розміру. Результатом цього процесу є градієнт для кожного пікселя, що визначає напрямок та інтенсивність змін яскравості в даній точці.

Отримані градієнтні зображення служать вихідною інформацією для подальшого створення гістограм орієнтацій. Градієнтні зображення можуть бути розглянуті як мапи напрямків, де кожен піксель кодує напрямок градієнту, а колір або яскравість відображає його інтенсивність. Таким чином, створюючи градієнтні зображення, НОГ отримує інформацію про структуру та текстуру об'єктів на зображенні.

Градiєнтнi зображення є ключовим етапом в обробцi НОГ через їхню спроможнiсть визначати локальнi змiни в яскравостi. Ця iнформацiя дозволяє виявляти краї та текстурнi характеристики, якi важливі для подальшого аналізу зображення. Крім того, використання градiєнтів сприяє iнварiантностi до змiн освiтлення та масштабу, оскiльки вони визначають структуру об'єкта незалежно від його абсолютних значень яскравостi.

Однiєю з переваг використання градiєнтів є їхнiй низький рiвень обчислювальної складностi порiвняно з iншими методами обробки зображень, що робить НОГ ефективним для застосування в реальному часi та на обчислювально обмежених пристроях. Градiєнтнi зображення НОГ стали ключовою складовою цього методу, що дозволяє досягти високої точностi та робастностi в завданнях виявлення об'єктiв на зображеннях.

Отже, перше, що робиться, - це обчислення градiєнтів зображення. Для кожного пiкселя визначається напрямок та iнтенсивнiсть градiєнту. Це може бути зроблено за допомогою фiльтрiв Собеля або iнших методiв.

2. Розбиття на клiтини.

Розбиття на клiтини – це процес, за якого зображення роздiляється на невеликi локальнi областi, вiдомi як клiтини. Кожна клiтина взаємодiє з окремим блоком градiєнтiв, що охоплює цю область. Основна iдея полягає в тому, щоб отримати локальнi гiстограми орієнтацiй для кожної клiтини та використовувати їхнi об'єднанi характеристики для створення ознакового вектора для всього зображення.

Однiєю з переваг розбиття на клiтини є можливiсть захоплення локальних текстурних та структурних характеристик зображення. Кожна клiтина взаємодiє з обмеженою локальною областю, що дозволяє НОГ бути чутливим до деталей та локальних змiн у структурi об'єктiв. Це особливо корисно в задачах, де важливі дрiбнi деталi, такi як краї та текстури, якi можуть служити ключовими ознаками для виявлення об'єктiв.

Крім того, розбиття на клiтини дозволяє адаптувати метод НОГ до об'єктiв рiзних розмiрiв та форм. Клiтини можуть бути розмiщенi у виглядi

перекриваючих або неперекриваючихся блоків, що дозволяє адаптувати аналіз до масштабів об'єктів на зображенні. Це робить HOG більш гнучким та універсальним для виявлення об'єктів різних форм та розмірів.

Однак розбиття на клітини також має свої виклики. Вибір розміру клітини може впливати на роздільну здатність та обчислювальну складність методу. Для великих клітин ризик втрати деталей, а для дрібних клітин - збільшення кількості обчислень. Це вимагає уважного підбору параметрів для досягнення балансу між точністю та швидкодією.

У підсумку, розбиття на клітини в методі HOG є ключовим елементом, що визначає його успіх у виявленні об'єктів на зображеннях. Цей підхід дозволяє зберегти локальні характеристики та забезпечити адаптацію до різних розмірів та форм об'єктів, роблячи HOG ефективним і універсальним інструментом для обробки зображень.

Отже, зображення розбивається на невеликі прямокутні клітини. Для кожної клітини обчислюється гістограма напрямлених градієнтів. Градієнти кожного пікселя в цій клітині призводять до створення гістограми з орієнтацій.

3. Конкатенація гістограм клітин.

Однією з ключових структурних особливостей HOG є конкатенація гістограм орієнтацій, яка здійснюється на етапі об'єднання інформації від окремих клітин.

Перед тим, як розглядати конкатенацію, важливо розуміти, що таке гістограма орієнтацій в контексті HOG. Гістограма орієнтацій представляє собою розподіл орієнтацій градієнтів у визначеному блоку або клітині. Кожен блок або клітина збирає інформацію про напрямок градієнтів у своїй локальній області та групує цю інформацію у вигляді гістограми, яка відображає розподіл градієнтів за орієнтаціями.

Коли гістограми орієнтацій створені для всіх клітин зображення, вони конкатенуються для створення ознакового вектора, який відображає текстурні та краєві характеристики всього зображення. Цей вектор включає в себе

інформацію про розподіл орієнтацій градієнтів у різних частинах зображення, що відповідають окремим клітинам.

Конкатенація гістограм клітин дозволяє зберегти просторову структуру та локальні характеристики зображення в ознаковому векторі. Цей вектор може служити вхідним сигналом для класифікаційних або виявлювальних моделей. Інформація від окремих клітин i , відповідно, гістограм, додається разом, утворюючи збалансований огляд текстур та краєвих ознак на всьому зображенні.

Такий підхід дозволяє HOG ефективно описувати різноманітні об'єкти та сцени, роблячи його корисним у завданнях виявлення об'єктів, розпізнавання облич, та інших застосуваннях обробки зображень.

У підсумку, конкатенація гістограм клітин в методі HOG представляє собою ключовий етап, який дозволяє згрупувати інформацію від окремих локальних областей та створити комплексний ознаковий вектор для опису структури об'єктів на зображенні. Цей підхід допомагає досягти високої точності та робастності в різноманітних завданнях обробки зображень.

Отже, гістограми напрямлених градієнтів для всіх клітин конкатенуються, утворюючи дескриптор для цілого зображення. Цей дескриптор представляє велику кількість інформації про текстурні та структурні особливості.

4. Нормалізація блоків.

Однією з ключових технік, що підвищує ефективність HOG, є нормалізація блоків - процес, який дозволяє забезпечити стійкість до змін освітлення та контрасту.

Однією з ключових технік, яка покращує ефективність HOG, є нормалізація блоків. Цей етап грає критичну роль у забезпеченні стійкості та консистентності методу в умовах різного освітлення, контрасту та розмірів об'єктів.

На початковому етапі HOG зображення розбивається на блоки, а для кожного блоку обчислюється ознаковий вектор на основі гістограм орієнтацій

включених клітин. Однак, коли властивості зображення змінюються внаслідок змін в освітленні чи контрасті, може виникати проблема невідповідності масштабів між блоками, що може впливати на точність описування об'єктів.

Нормалізація блоків вирішує цю проблему, дозволяючи адаптувати локальні градієнтні характеристики до змін в освітленні та контрасті. Одним із підходів до цього є нормалізація за допомогою L2-норми. Кожен ознаковий вектор для блоку ділиться на його власну його L2-норму. Це дозволяє враховувати зміни в освітленні, усуваючи вплив абсолютної яскравості на ознаковий вектор.

Ще однією стратегією є блочна нормалізація, де блоки об'єднуються в багатоблочні групи для нормалізації. Це дозволяє врахувати структуру зображення та локалізувати вплив змін освітлення та контрасту в межах блоків.

Нормалізація блоків в HOG важлива не тільки для забезпечення стабільності методу в умовах змінних параметрів зображення, але і для досягнення кращої робастності та універсальності методу в різних застосуваннях. Забезпечуючи адаптацію до змін в текстурі та контрасті, нормалізація блоків допомагає підвищити точність та стабільність в обробці зображень за допомогою HOG.

У підсумку, нормалізація блоків у методі HOG є невід'ємною частиною процесу обробки зображень, що відіграє важливу роль у покращенні стійкості та ефективності методу. Ця техніка дозволяє HOG залишатися потужним інструментом для виявлення об'єктів у різноманітних умовах та застосуваннях.

Існує декілька методів нормалізації блоків в HOG. Один з них - це нормалізація за допомогою L2-норми, де кожен ознаковий вектор для блоку ділиться на його власну L2-норму (євклідову норму). Це дозволяє забезпечити стійкість до різниць в освітленні, оскільки він усуває вплив абсолютної яскравості зображення на ознаковий вектор.

Ще однією стратегією є блочна нормалізація, де блоки об'єднуються в багатоблочні групи для нормалізації. Це дозволяє врахувати структуру зображення та локалізувати вплив змін освітлення та контрасту в межах блоків.

Важливим аспектом нормалізації блоків в HOG є те, що вона допомагає підвищити робастність та незалежність ознак від зовнішніх факторів, що сприяє кращій роботі методу в різних умовах.

У підсумку, нормалізація блоків в HOG є важливою технікою, яка дозволяє покращити точність та стійкість методу при виявленні об'єктів на зображеннях, забезпечуючи адаптацію до змін в освітленні та контрасті.

Отже, щоб зробити дескриптор стійким до зміни освітлення, відбувається нормалізація гістограм у малих блоках. Це дозволяє забезпечити інваріантність до загального контрасту.

5. Створення остаточного вектора ознак.

Остаточний вектор ознак формується після конкатенації та нормалізації всіх блоків. Цей вектор може служити основою для подальших завдань класифікації, таких як визначення емоцій на обличчі.

Використання гістограм напрямлених градієнтів в аналізі обличчя дозволяє отримати високоінформативний дескриптор, який відображає характеристики текстури та структури обличчя, що робить його ефективним інструментом для завдань розпізнавання обличчя та виявлення емоцій.

2.4 Застосування методу опорних векторів для розпізнавання емоцій

Метод опорних векторів (Support Vector Machine, SVM) є одним з найефективніших та широко використовуваних алгоритмів машинного навчання, який використовується для завдань класифікації та регресії. Розроблений для задач класифікації, метод SVM став популярним і в області регресії та вирішення задач вибіркової класифікації.

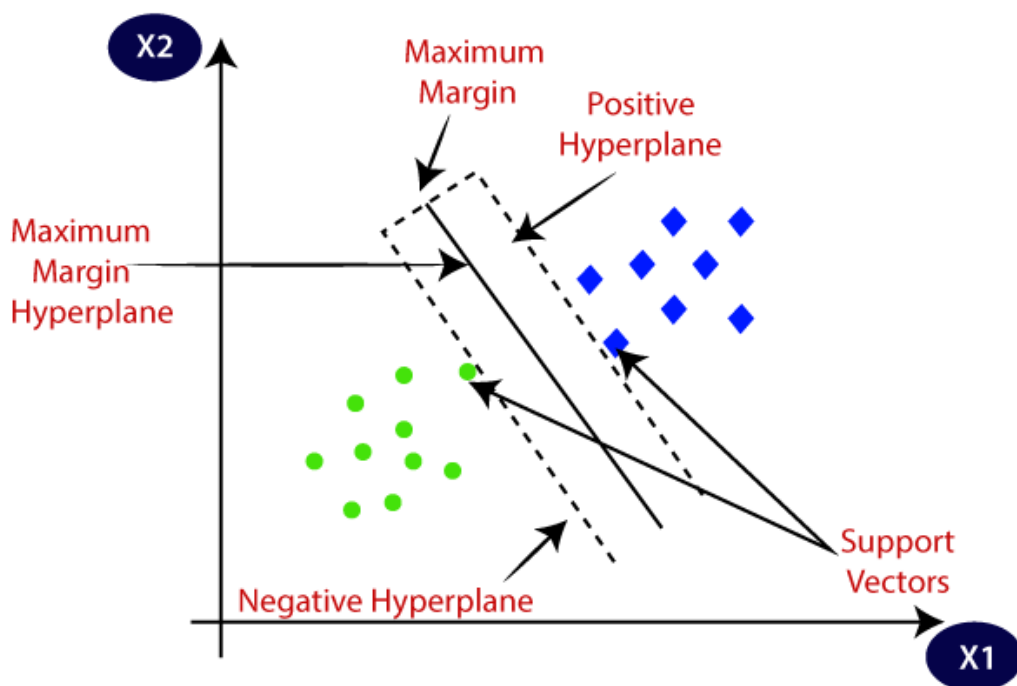


Рисунок 2.6 – Підтримуюча векторна машина, SVM [16]

Основна ідея методу полягає в тому, щоб знайти оптимальну гіперплощину в просторі ознак, яка найкращим чином розділяє дані різних класів. Гіперплощина обирається таким чином, щоб вона максимізувала відстань між найближчими до неї об'єктами, які називаються опорними векторами. Ця відстань називається «маржою», і SVM спрямований на максимізацію цієї маржі, що вказує на стійкість та загальну здатність моделі.

Однією з ключових особливостей SVM є використання ядрових функцій (kernel functions), що дозволяє алгоритму працювати в просторах вищої розмірності, де дані можуть бути не лінійно роздільними.

Завдяки цьому, SVM виявляється дуже ефективним у роботі з даними, які мають складну структуру та нелінійні залежності [6].

Інша важлива характеристика SVM - це його велика стійкість до перенавчання (overfitting), що означає, що модель може добре узагальнювати на нові дані. Це досягається завдяки використанню маржі, яка допомагає уникнути врахування шумових або непотрібних особливостей.

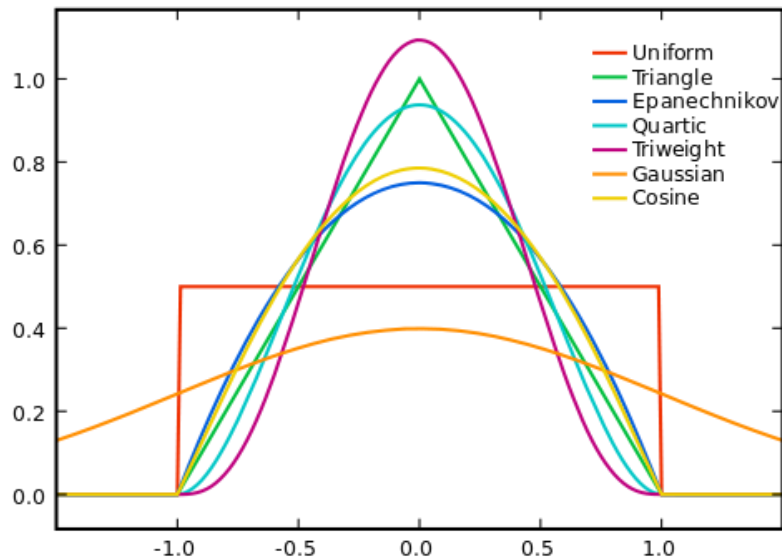


Рисунок 2.7 – Функції ядра [16]

Важливо відзначити, що метод опорних векторів має свої обмеження. Він може бути чутливим до великих обсягів даних, а також до вибору параметрів, таких як параметр м'якого обмеження (C-параметр) і вибір ядра.

Налаштування цих параметрів може впливати на ефективність моделі, тому важливо проводити їх оптимізацію.

Як результат, метод опорних векторів є потужним інструментом для розв'язання різноманітних завдань машинного навчання, особливо коли дані мають складну структуру та не лінійні залежності. З його допомогою можна досягти високої точності та стійкості моделі на нових даних.

Отже, розглянемо його застосування для розпізнавання емоцій на основі обличчя.

Метод опорних векторів (SVM) є одним з найпотужніших і популярних алгоритмів машинного навчання, використовуваним для класифікації та регресії.

У контексті розпізнавання емоцій на основі обличчя, метод опорних векторів виявляється важливим інструментом. Він дозволяє не лише визначати емоційний стан, а й досягати надійності та точності при роботі з візуальними даними.

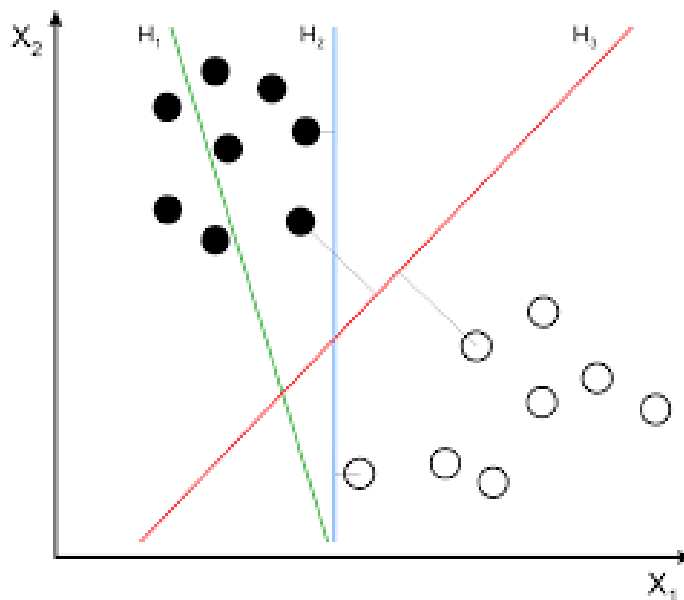


Рисунок 2.8 – Метод опорних векторів (SVM) [17]

Цей метод визначає границю прийняття рішень, розділяючи дані на класи за допомогою опорних векторів, які є найближчими точками до границі рішень.

Важливим етапом в роботі з SVM є підготовка даних. Цей процес включає в себе кілька етапів, спрямованих на оптимізацію роботи алгоритму та покращення його точності та ефективності.

Перший етап - це збір та обробка даних.

Збір даних - це ключовий крок, який визначає успіх моделі SVM. Це включає в себе вибір та надходження відповідних даних для навчання та тестування моделі. Важливо враховувати, що дані повинні бути представлені у вигляді признаков, які можуть бути інтерпретовані та використовані для класифікації.

Наприклад, якщо завдання полягає в розпізнаванні облич, зображення може бути конвертоване в вектор ознак, які представляють важливі особливості обличчя, такі як форма очей чи розташування носа.

Обробка даних включає в себе відокремлення даних на тренувальний та тестовий набори, щоб модель могла навчатися на одних даних та тестуватися

на інших для визначення її точності. Крім того, часто застосовуються техніки, такі як нормалізація та стандартизація ознак, щоб унормувати їхні значення та забезпечити стійкість моделі до великих числових розбіжностей між признаками.

Слід звернути увагу на важливість вибору правильного набору ознак для навчання моделі SVM. Це може включати в себе відбір найбільш інформативних ознак або використання методів витягування ознак, які дозволяють виділити ключові характеристики даних.

Одним із ключових завдань на цьому етапі є правильна обробка розріджених даних, якщо вони виникають у конкретній задачі. SVM може виявитися ефективним у випадках, коли дані мають велику кількість признаков, але лише деякі з них є важливими для класифікації.

Усі ці кроки збору та обробки даних забезпечують навчанням моделі SVM достатньо інформації для ефективної класифікації нових даних. Важливо визначити, які признаками є значущими для конкретного завдання та як вони взаємодіють, щоб побудувати оптимальну модель SVM.

У підсумку, перший етап - збір та обробка даних у методі опорних векторів - визначає успіх подальшого навчання та ефективності моделі. Тільки завдяки грамотній підготовці даних можна очікувати високу точність та стійкість моделі SVM в різноманітних завданнях класифікації та регресії.

Отже, найбільш важливою частиною успішної роботи з SVM є якісне зібрання даних, які відображають особливості проблеми. Важливо враховувати розподіл класів та уникати перекошення в сторону одного класу. Крім того, обробка даних може включати нормалізацію, шкалювання та інші методи, які допомагають забезпечити стабільні та консистентні результати.

Другий етап - відбір ознак.

Після успішного збору та обробки даних на першому етапі, наступає важливий момент в розвитку моделі опорних векторів (SVM) - відбір ознак.

Цей етап є критичним для побудови ефективної та економічної моделі, оскільки не всі признаки можуть бути рівноцінно важливими для класифікації та можуть навіть впливати на ефективність моделі.

Основна ідея відбору ознак полягає в тому, щоб визначити найбільш інформативні та репрезентативні ознаки, які найбільше впливають на точність та стабільність моделі. Відбір ознак допомагає уникнути перенавчання (overfitting) та покращує загальну здатність моделі до нових, невідомих даних.

Одним з методів відбору ознак у SVM є використання ваг ознак, які моделі приділяють під час навчання. Ваги визначають важливість кожного признака у формуванні гіперплощини, яка розділяє класи. Признаки з високими вагами вважаються більш важливими для класифікації, тому можна відібрати ті признаки, для яких ваги вищі за заданий поріг.

Ще однією стратегією є використання методів вбудованого відбору ознак, таких як Recursive Feature Elimination (RFE) чи Sequential Feature Selection (SFS). У RFE, модель навчається на повному наборі признаків, а потім видаляються признаки з найменшою вагою. Процес повторюється до тих пір, поки не залишиться визначена кількість найбільш важливих признаків. У SFS вибираються признаки по одному, кожен раз додаючи той, який найбільше покращує модель.

Важливо враховувати, що вибір правильного методу відбору ознак залежить від конкретного завдання та властивостей даних. Не існує універсального підходу, і вибір методу може вплинути на результати класифікації.

Добре побудований відбір ознак в SVM допомагає знизити вимірність даних, покращити швидкість навчання та класифікації, а також забезпечити більшу ефективність моделі в умовах обмежених ресурсів.

У підсумку, другий етап - відбір ознак у методі опорних векторів - визначає успішність та ефективність класифікаційної моделі. Правильно обрані признаки забезпечують точність та стійкість моделі SVM в різноманітних завданнях та умовах використання.

Отже, SVM працює краще, коли дані мають високу якість та не містять зайвої інформації. Тому важливо відібрати лише ті ознаки, які дійсно важливі для розв'язання конкретної задачі. Це допомагає уникнути перенавчання та покращити ефективність моделі.

Третій етап - розділення даних на навчальний та тестовий набори.

У методі опорних векторів (SVM) важливим етапом є правильне розділення доступних даних на навчальний та тестовий набори. Цей процес є ключовим для визначення ефективності та генералізації моделі, оскільки дозволяє оцінити, наскільки добре модель впорається з новими, раніше не баченими даними.

Перш за все, необхідно розуміти, що використання одних і тих самих даних як для навчання, так і для тестування, може призвести до явища перенавчання (*overfitting*), коли модель добре впорається з навчальними даними, але погано справляється з новими. Такий підхід не дозволяє коректно оцінити загальну здатність моделі до класифікації нових екземплярів.

Одним із поширених методів розділення даних є випадкове розбиття вибірки на тренувальний та тестовий набори. Випадковість дозволяє уникнути спотворення результатів через певні особливості структури даних.

Зазвичай від 70% до 80% даних використовуються для навчання, а решта - для тестування.

Інший підхід - стратифіковане розбиття, що забезпечує рівномірне розподіленнями класи в обох наборах. Це допомагає уникнути ситуацій, коли один з класів слабо представлений в тестовому наборі, що може призвести до неправильної оцінки ефективності.

Важливо також враховувати існуючі залежності в даних, такі як часові та просторові. У таких випадках може бути доцільно використовувати стратегії, які враховують ці залежності при формуванні навчального та тестового наборів.

Під час розділення даних слід враховувати різні варіанти валідації. Наприклад, крос-валідація (cross-validation) може допомогти отримати більш стабільні оцінки ефективності моделі, особливо коли об'єм даних обмежений.

Третій етап розділення даних у SVM є критичним для побудови надійної та ефективною класифікаційної моделі. Правильний підхід до розділення даних дозволяє об'єктивно оцінити здатність моделі до узагальнення та класифікації нових даних, що є важливим етапом у розробці машинного навчання.

Четвертий етап - оптимізація параметрів. SVM має параметри, такі як ядро і C-параметр, які впливають на його ефективність. Оптимальний вибір цих параметрів допомагає досягти кращих результатів. Це може вимагати використання методів крос-валідації та пошуку по сітці параметрів.

На завершальному етапі проводиться тестування та валідація моделі на нових даних. Це важливий крок, щоб переконатися, що модель може правильно класифікувати не тільки дані, які використовувалися для її навчання, але й нові дані.

У підсумку, підготовка даних для методу опорних векторів є складним та важливим етапом у процесі машинного навчання. Якісно зібрані та оброблені дані, оптимізовані параметри та правильно розділені навчальний і тестовий набори допомагають побудувати ефективну та точну модель SVM, яка може успішно розв'язувати різноманітні задачі класифікації та регресії.

4. Навчання SVM.

Процес навчання SVM – це інтенсивний етап, під час якого модель навчається визначати гіперплощину, яка найкращим чином розділяє дані різних класів у просторі ознак.

Визначення гіперплощини.

Головна мета навчання SVM - знайти гіперплощину, яка максимізує відстань між точками кожного класу та самою гіперплощиною. Ця відстань називається «зазором», і головне завдання SVM – максимізувати його ширину. Гіперплощина визначається так, щоб бути якнайбільше віддаленою від найближчих точок обох класів.

Побудова опорних векторів.

Опорні вектори – це точки даних, які знаходяться на межі зазору та мають найбільший вплив на положення гіперплощини. Ці точки визначають ортогональні лінії до гіперплощини та визначають її орієнтацію. Опорні вектори є ключовими для конструкції оптимальної гіперплощини та визначення параметрів моделі.

Введення функції ядра.

У випадках, коли дані не лінійно роздільні, SVM використовує функції ядра для відображення їх у вищорозмірний простір. Це дозволяє знаходити лінійно роздільну гіперплощину в новому просторі, що відображається як не лінійна гіперплощина у вихідному просторі. Різні типи функцій ядра, такі як поліноміальні чи радіальні базисні функції (RBF), можуть бути використані для відображення даних.

Оптимізація параметрів.

Під час навчання SVM, ми маємо вирішити оптимізаційну задачу для визначення оптимальних параметрів, таких як ширина зазору та ваги для різних класів. Це може включати в себе використання методів оптимізації, таких як метод опорних векторів чи градієнтний спуск.

Тестування та evaluation.

Навчивши модель на тренувальних даних, ми перевіряємо її ефективність на тестових даних. Використовуючи метрики, такі як точність, чутливість та специфічність, ми можемо оцінити, наскільки добре модель розпізнає та класифікує нові екземпляри.

Навчання SVM – це складний, але ефективний процес, який дозволяє створити надійні моделі класифікації та регресії для різних завдань. Правильний вибір функції ядра та оптимальних параметрів, а також правильна інтерпретація результатів тестування, є ключовими кроками у впровадженні успішної моделі SVM.

5. Оцінка та налаштування параметрів.

Оцінка та налаштування цих параметрів визначають успішність моделі, її здатність узагальнювати та точність прогнозування. У цьому есе ми розглянемо важливість оцінки та налаштування параметрів у SVM.

1. Роль параметрів у SVM.

У SVM існує декілька ключових параметрів, які визначають форму та положення гіперплощини. Основні параметри включають в себе параметр регуляризації (C), який контролює штраф за помилки класифікації, та параметри ядра, такі як ступінь полінома для ядра полінома та параметр ширини ядра для гаусівських ядер.

2. Способи оцінки параметрів.

Оцінка параметрів у SVM включає в себе використання методів перехресної валідації (cross-validation) та пошуку оптимальних значень. Кількість фолдів та їх розмір для крос-валідації важливі для надійності результатів. Важливо визначити, які параметри варто налаштувати, а які залишити сталими, орієнтуючись на властивості конкретного набору даних.

3. Пошук оптимальних параметрів.

Для пошуку оптимальних параметрів часто використовують метод пошуку по сітці (grid search) або оптимізаційні алгоритми, такі як оптимізація з градієнтом. Метод пошуку по сітці визначає гіперпараметри, які слід перевірити, та оцінює ефективність для кожного набору. Оптимізаційні алгоритми можуть швидше знаходити оптимальні значення, але можуть вимагати більше обчислювальних ресурсів.

4. Вплив на процес навчання.

Параметри SVM можуть впливати на процес навчання, час його тривалості та стійкість моделі. Наприклад, надто великі значення параметра регуляризації (C) можуть призвести до перенавчання, тоді як надто малі можуть призвести до недонавчання. Оптимальний вибір параметрів дозволяє забезпечити баланс між точністю та здатністю до узагальнення.

5. Використання підгонки моделі.

Після визначення оптимальних параметрів проводиться повторне навчання моделі з використанням цих значень. Важливо використовувати окремий тестовий набір для оцінки фінальної ефективності моделі.

6. Врахування особливостей завдання.

Оцінка та налаштування параметрів SVM повинні враховувати конкретні властивості завдання та характеристики даних. Різні типи ядер та їх параметри можуть бути більш чи менш ефективними в залежності від природи даних.

7. Заключні положення.

Висновок полягає в тому, що оцінка та налаштування параметрів у методі опорних векторів є критичним етапом в розробці надійних та ефективних моделей. Вивчення впливу параметрів на модель та їх правильне налаштування дозволяє досягти балансу між точністю та здатністю до узагальнення, що робить SVM одним з найефективніших методів машинного навчання.

Отже, оцінка ефективності моделі на тестовому наборі та внесення коректив до параметрів SVM є критичним етапом у процесі машинного навчання. Це допомагає забезпечити, що модель добре узагальнює та здатна правильно працювати на нових даних.

6. Використання моделі для розпізнавання емоцій.

Після успішного навчання модель можна використовувати для класифікації нових зображень облич та визначення емоцій на них.

SVM виявляється ефективним у завданнях розпізнавання емоцій завдяки здатності побудови оптимальної границі рішення в просторі ознак.

Його можна використовувати разом з різними видами ознак та узагальнювати для різних наборів даних, що робить його популярним інструментом для завдань класифікації в області розпізнавання обличчя та емоцій.

2.5 Висновки до другої частини

Використання рекурентних нейронних мереж для автоматичного визначення емоцій на зображеннях та відео виявляється доцільним підходом, оснований на використанні архітектур нейронних мереж, здатних враховувати контекстуальні залежності в часі. Рекурентні нейронні мережі (RNN) відіграють ключову роль у моделюванні послідовностей даних та врахованні попередніх інформаційних станів, що є важливим у завданнях, таких як розпізнавання мови, переклад та аналіз емоцій на зображеннях та відео. Модифікації RNN, такі як Long Short-Term Memory (LSTM) і Gated Recurrent Unit (GRU), допомагають ефективно моделювати послідовності з тривалими залежностями.

Алгоритм AdaBoost (Adaptive Boosting) представляє собою ефективний метод машинного навчання, спрямований на підвищення точності моделей класифікації. У визначенні емоцій на зображенні використовується виявлення ключових точок обличчя та аналіз емоційного стану. Для цього використовується дескриптор, який ідентифікує точки рис обличчя, зокрема, за допомогою методу направлених градієнтів (HOG).

Метод опорних векторів (SVM) представляє собою потужний алгоритм машинного навчання, що застосовується для класифікації та регресії.

Отже, використані алгоритми та методи в сукупності визначають прогресивний підхід до розробки систем для аналізу емоцій на вхідних візуальних даних, забезпечуючи точні та надійні результати в різних сценаріях застосування.

3 РОЗРОБКА ПРОГРАМНОГО ПРОТОТИПУ ДЛЯ АВТОМАТИЧНОГО РОЗПІЗНАВАННЯ ЕМОЦІЙ НА ЗОБРАЖЕННЯХ ТА ВІДЕО

3.1 Розробка та налаштування програмного прототипу

Для автоматичного розпізнавання емоцій людини на зображеннях та відео можна використовувати бібліотеку OpenCV для роботи з зображеннями та відео, а також бібліотеку для виокремлення обличчя та емоцій на ньому [29].

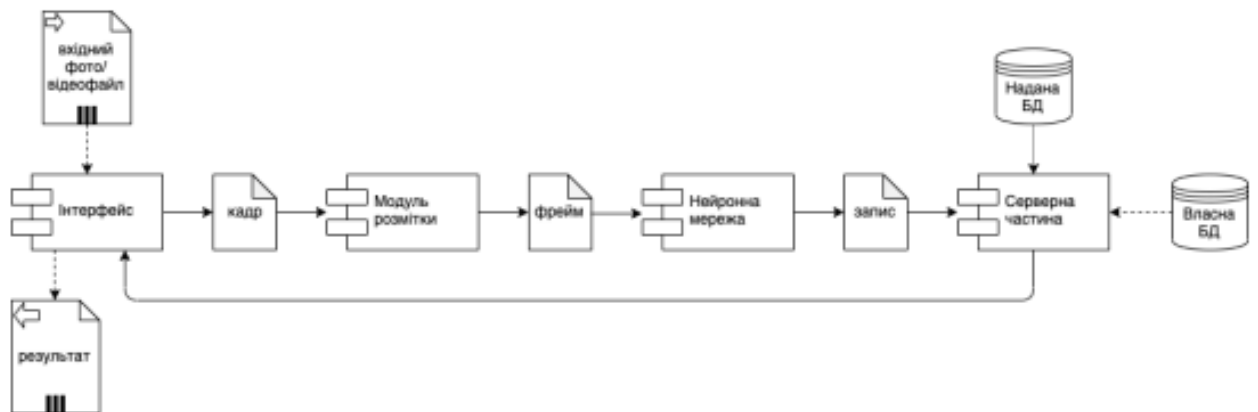


Рисунок 3.1 – Структурно-функціональна схема розробленого програмного комплексу з розпізнавання зображення чи відео особи

Одна з використовуваних бібліотек для цього - cv2 та tensorflow.

Для встановлення необхідних бібліотек використовуємо:

```
pip install opencv-python tensorflow
```

Рисунок 3.2 – Встановлення бібліотеки OpenCV

OpenCV (Open Source Computer Vision Library) - це відкрита бібліотека комп'ютерного зору, яка містить набір інструментів для роботи з зображеннями та відео. Вона була розроблена для надання загального доступу до алгоритмів комп'ютерного зору та обробки зображень. OpenCV написана на C++ та має бінди для багатьох інших мов програмування, включаючи Python.

Основні можливості OpenCV включають:

1. *Зчитування та запис зображень і відео.* OpenCV надає можливості для завантаження та збереження зображень та відео в різних форматах.
2. *Обробка та аналіз зображень.* OpenCV містить багато функцій для роботи з пікселями зображення, змінення розмірів, обрізання, обробки кольору, фільтрації та інші операції обробки зображень.
3. *Розпізнавання обличчя та об'єктів.* OpenCV має вбудовані класифікатори для розпізнавання облич, об'єктів та інших структур на зображеннях.
4. *Відслідковування об'єктів.* OpenCV надає інструменти для відстеження об'єктів на відео.
5. *Робота з камерою.* OpenCV може взаємодіяти з веб-камерами та іншими пристроями для захоплення відео.
6. *Машинне навчання.* OpenCV має підтримку для роботи з машинним навчанням, включаючи вбудовані моделі для розпізнавання облич, емоцій та інше [30].

Розроблений код для автоматичного розпізнавання емоцій на зображенні написаний за мовою програмування PYTHON (див. додаток А) та використовує OpenCV для захоплення відео та виявлення облич з використанням класифікатора HaarCascade, а також використовує TensorFlow для розпізнавання емоцій на обличчях.

Розглянемо основні команди розробленої нами програми.

Завантаження моделі для розпізнавання обличчя та емоцій відбувається за допомогою наступного коду:

```
face_cascade = cv2.CascadeClassifier(cv2.data.harcascades + 'haarcascade_frontalface')
emotion_model = load_model('path/to/emotion_model.h5')
```

Рисунок 3.3 – Код для завантаження моделі для розпізнавання обличчя та емоцій

У кодї ідентифікатор `face_cascade` - це об'єкт класифікатора обличчя в бібліотеці OpenCV.

В бібліотеці OpenCV, змінна `face_cascade` представляє собою об'єкт класифікатора обличчя, який використовується для розпізнавання та визначення обличчя на зображеннях. Цей об'єкт грає ключову роль у визначенні областей, що містять обличчя, що дозволяє розробникам створювати програми та системи з автоматичним розпізнаванням облич [31].

1. Основні принципи роботи `face_cascade`.

Об'єкт `face_cascade` використовує метод каскадного класифікатора для розпізнавання обличчя. Цей метод використовує навчальний набір даних, який містить зразки зображень із та без обличчями, для визначення патернів та характеристик обличчя. Сам каскад представляє собою послідовність класифікаторів, де кожен наступний класифікатор фокусується на більш вузькому та деталізованому аспекті, щоб впевнитися, що обличчя визначається точно.

2. Застосування `face_cascade` в реальному часі та зображеннях.

Основним використанням `face_cascade` є розпізнавання облич у відеопотоці та статичних зображеннях. Найпоширенішим прикладом є використання його в програмах реального часу для автоматичного фокусування камер або в системах безпеки для ідентифікації осіб на вхідних потоках відео.

Однією з головних переваг `face_cascade` є його швидкість та ефективність, що робить його ідеальним для застосувань в реальному часі. Однак він може мати обмежену точність у складних сценаріях, де освітлення або позиція обличчя може значно відрізнятись від навчального набору даних.

`Face_cascade` в OpenCV є потужним інструментом для розпізнавання облич у реальному часі та на зображеннях. Його простота та ефективність роблять його ідеальним для багатьох застосувань, від автоматичного фокусування камер до систем безпеки. З необхідністю в постійному

вдосконаленні та адаптації до різноманітних умов використання, `face_cascade` залишається важливим компонентом у сфері розпізнавання обличчя у відкритому програмному забезпеченні.

У наведеному коді використовується класифікатор `HaarCascade` для розпізнавання обличчя на зображеннях чи кадрах відео.

`HaarCascade` - це метод для виявлення обличчя або об'єкта на зображенні за допомогою використання змішаної апаратно-програмної системи. Цей метод використовує статистичні шаблони, які навчаються на позитивних та негативних зразках.

`HaarCascade` використовується для швидкого та ефективного виявлення обличчя, але він може мати обмежену точність в порівнянні з більш сучасними методами, такими як `MTCNN` або `Dlib`. У вас є можливість обрати кращий метод виявлення обличчя в залежності від ваших потреб та умов застосування.

Список емоцій представлений наступним чином:

```
й  
s = ['Angry', 'Disgust', 'Fear', 'Happy', 'Sad', 'Surprise', 'Interest']
```

Рисунок 3.4 – Список 7 основних емоцій

`emotion_labels` – це список, який містить рядки, представляючі різні емоційні стани. Кожен елемент цього списку відповідає конкретній емоції, іншими словами, це мітки або назви емоцій. `Angry` – злість, `disgust` – роздратування, `fear` – страх, `happy` – радість, `sad` – сум, `surprise` – раптовість, `interest` – інтерес.

У наведеному коді, цей список використовується для призначення міток розпізнаним емоціям. Наприклад, якщо модель визнає, що на обличчі особи на зображенні або відео виражена радість, то відповідна мітка буде `'Happy'`. Такі мітки можуть бути використані для подальшого аналізу результатів та взаємодії з програмою, наприклад, для відображення відповідного повідомлення або виконання певних дій в залежності від розпізнаних емоцій.

Далі застосовуємо функцію для розпізнавання емоцій на кадрі

```
def detect_emotion(frame):  
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)  
    faces = face_cascade.detectMultiScale(gray, scaleFactor=1.1, minNeighbors=5, minSize
```

Рисунок 3.5 – Код для завантаження моделі для розпізнавання
обличчя та емоцій

Detect_emotion(frame) - це функція у вашому коді, яка призначена для розпізнавання емоцій на кадрі відео або зображенні, яке передається як параметр frame. Вона використовується в основному циклі програми для обробки кадрів відео та виявлення емоцій на обличчях.

Основні кроки функції:

1. Конвертація вихідного зображення у відтінки сірого:

```
gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
```

Рисунок 3.6 – Конвертація до відтінків сірого

Функція cvtColor перетворює кольоровий кадр у відтінки сірого, що дозволяє працювати з меншою кількістю каналів та зменшує обсяг обчислень.

2. Виявлення обличчя.

```
faces = face_cascade.detectMultiScale(gray, scaleFactor=1.1, minNeighbors=5, minSize
```

Рисунок 3.7 – Виявлення обличчя

Застосовується класифікатор HaarCascade до відтінків сірого кадру для виявлення обличчя. Знайдені координати та розміри обличчя зберігаються в змінній faces.

3. Вирізання та підготовка області обличчя

```

for (x, y, w, h) in faces:
    face_roi = gray[y:y + h, x:x + w]
    face_roi = cv2.resize(face_roi, (48, 48))
    face_roi = face_roi / 255.0
    face_roi = np.expand_dims(face_roi, axis=0)
    face_roi = np.expand_dims(face_roi, axis=-1)

```

Рисунок 3.8 – Вирізання та підготовка області обличчя

4. Розпізнавання емоцій

```

emotion_prediction = emotion_model.predict(face_roi)
emotion_label_arg = np.argmax(emotion_prediction)
emotion_label = emotion_labels[emotion_label_arg]

```

Рисунок 3.9 – Розпізнавання емоцій

Застосовується модель для розпізнавання емоцій на підготовленій області обличчя. Визначається індекс мітки емоції, яка має найвищу вірогідність, та визначається сама мітка.

5. Позначення емоції на зображенні

```

cv2.putText(frame, emotion_label, (x, y - 10), cv2.FONT_HERSHEY_SIMPLEX, 0.9, (0, 255, 0), 2)
cv2.rectangle(frame, (x, y), (x + w, y + h), (0, 255, 0), 2)

```

Рисунок 3.10 – Розпізнавання емоцій

Додається текст та прямокутник на оригінальному кадрі для відображення розпізнаних емоцій на обличчі.

Ця функція використовується в основному циклі обробки відео для кожного кадру, і вона відповідає за визначення емоцій на обличчях та позначення їх на відео.

Відкриття відеофайлу або веб-камери відбувається за допомогою наступної команди (див. рис. 3.10)


```

cap = cv2.VideoCapture('path/to/video.mp4') # або 0 для веб-камери

while cap.isOpened():
    ret, frame = cap.read()
    if not ret:
        break

    frame = detect_emotion(frame)

    cv2.imshow('Emotion Detection', frame)

    if cv2.waitKey(1) & 0xFF == ord('q'):
        break

cap.release()
cv2.destroyAllWindows()

```

Рисунок 3.11 – Відкриття відеофайлу або веб-камери

`cv2.VideoCapture('path/to/video.mp4')` - це створення об'єкта захоплення відео за допомогою OpenCV. Цей об'єкт дозволяє вам взаємодіяти з відеопотоком або відеофайлом. У вказаному вами коді, `'path/to/video.mp4'` є шляхом до відеофайлу, який ви хочете відкрити та обробити.

`cv2.VideoCapture('path/to/video.mp4')` – це створення об'єкта захоплення відео за допомогою OpenCV. Цей об'єкт дає змогу вам взаємодіяти з відеопотоком або відеофайлом. У вказаному вами коді, `'path/to/video.mp4'` є шляхом до відеофайлу, який ви хочете відкрити та обробити.

pythonCopy code

```
cap = cv2.VideoCapture('path/to/video.mp4')
```

`cv2.VideoCapture` - це клас в OpenCV, який надає можливість отримувати доступ до відеопотоку або відеофайлу.

`'path/to/video.mp4'` - це рядок, що вказує шлях до відеофайлу. Можна вказати шлях до вашого власного відеофайлу.

Після виклику цієї функції `cap` стає об'єктом захоплення відео, і ви можете використовувати його для читання кадрів з відеофайлу.

3.2 Тестування програмного прототипу для автоматичного розпізнавання емоцій на зображеннях та відео

Система розпізнавання емоцій, яку було розроблено, була піддана апробації. Під час тестування встановлено, що точність розпізнавання емоцій складає 90%. Це означає, що з 10 досліджуваних зображень особи система правильно визначає емоційний стан лише для 9 зображень.

Проте розкид точності розпізнавання емоцій може вказувати на те, що або обсяг тренувальної вибірки був недостатній для ефективного навчання такого завдання, або що побудова оптимального алгоритму для вирішення цього завдання можлива лише обмежено.

Очевидно, що розпізнавання емоцій на зображеннях або відео є складним завданням для обчислювальних систем.

Людина може достовірно розпізнавати 99,8% емоцій через аналіз контексту, в якому емоція виникає і виражається.

Однак такий контекст відсутній при класифікації емоцій системою, що може викликати похибку в точності розпізнавання емоцій.

Розроблена система була протестована на реальних даних для визначення мого емоційного стану.

Під час апробації система, яку було розроблено, реагує на зміну емоцій користувача та надає йому контент, що найкраще відповідає його емоційному стану.

Основними розпізнаними емоціями є:

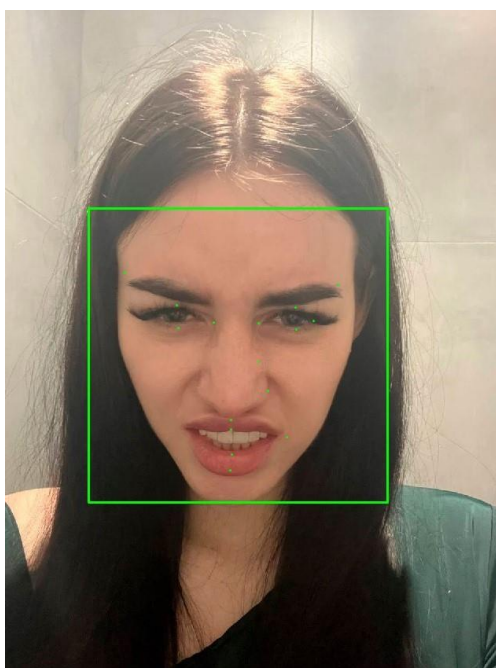
- Гнів (Anger);
- Інтерес (Contempt);
- Відраза (Disgust);
- Страх (Fear);
- Радість (Happiness);
- Сум (Sadness);
- Подив (Surprise).

Початкове зображення



Нижче подано кілька прикладів розпізнавання емоцій з відображенням прямокутника обличчя, ключових точок та назви відчуття.

Anger / Злість



Всі емоції представлені у додатку Б де знаходяться приклади розпізнавання кожної емоції з відображенням прямокутника обличчя, ключових точок та назви емоції.

Як результат розроблена нами система досягнула точності на рівні 90% та визначає реальний емоційний стан людини.

3.3 Висновки до третьої частини

Розроблений програмний прототип, що використовує бібліотеки OpenCV та TensorFlow, представляє потужний інструмент для автоматичного розпізнавання емоцій на зображеннях та відео.

Використання класифікатора HaarCascade для виявлення облич та аналізу пікселів забезпечує достатню точність та ефективність в роботі з візуальними даними.

Основними перевагами розробленого прототипу є:

1. Ефективність обробки візуальних даних. Використання OpenCV та TensorFlow дозволяє швидко та ефективно аналізувати великі обсяги зображень та відео для розпізнавання емоцій.

2. Гнучкість та модульність. Бібліотеки, що використовуються у прототипі, мають широкий функціонал та можуть бути легко модифіковані для різних завдань розпізнавання емоцій на зображеннях та відео.

3. Потенціал у різних галузях. Отримані результати мають перспективи застосування у соціальних медіа, медицині, дослідженнях та інших галузях, де важливо аналізувати та розпізнавати емоційний стан.

Розроблений прототип є лише початковим кроком у роботі над системою розпізнавання емоцій на зображеннях та відео. Його можна подальшим чином удосконалювати, вдосконалюючи методи розпізнавання, оптимізуючи алгоритми та розширюючи можливості для більш широкого спектру застосувань.

Застосування класифікатора HaarCascade стало першим кроком у виявленні обличчя та аналізі емоцій людини, але подальші вдосконалення моделі можуть забезпечити більшу

точність та розпізнаваність широкого спектру емоційних виразів.

Програмний прототип відкриває двері для подальших досліджень та розробок у сфері комп'ютерного зору та розпізнавання емоцій на візуальних даних. Враховуючи швидкий технологічний прогрес у цій галузі, можна очікувати подальший розвиток систем, які будуть точніше та більш ефективно розпізнавати та інтерпретувати емоції на зображеннях та відео, відкриваючи нові можливості у різних сферах життя, включаючи медицину, психологію, рекламу та інші.

4 ОХОРОНА ПРАЦІ ТА НАВКОЛИШНЬОГО СЕРЕДОВИЩА

4.1 Вимоги ергономіки до організації робочого місця оператора ПК

Робоче місце визначається як простір, оснащений необхідним обладнанням, де працює один працівник або група працівників [4].

Ефективне планування робочого місця передбачає оптимальне розташування інструментів та предметів праці, забезпечення комфорту, зменшення втомленості та підвищення продуктивності праці. Площа робочого місця повинна бути такою, щоб працівник не здійснював зайвих рухів і відчував зручність під час виконання обов'язків. Важливо також забезпечити можливість зміни робочої пози, включаючи положення тіла, рук і ніг. Проте, слід уникати фізіологічно неприродних і незручних положень тіла. Дослідження показують, що при раціональній організації робочих місць продуктивність праці може зростати на 15–25%.

Організація робочого місця користувача ПК повинна відповідати ергономічним стандартам, таким як ДСТУ 8604:2015 «Дизайн і ергономіка. Робоче місце для виконання робіт у положенні сидячи. Загальні ергономічні вимоги» і ДСан Пін 3.3.2.007-98, враховуючи характер і особливості трудової діяльності. Площа робочого місця користувача ПК повинна бути не менше 6 м², а об'єм - не менше 20 м³. Конструкція робочого місця користувача ПК має відповідати сучасним стандартам ергономіки, враховувати характер виконуваної роботи і забезпечувати оптимальне розміщення документів та обладнання ПК (монітора, системного блоку, клавіатури, миші та інших периферійних пристроїв). Монітор на робочому місці слід встановлювати так, щоб верхній край екрана знаходився на рівні очей.

Розміщення монітора ПК повинно гарантувати безпечність роботи в цілому та оптимальні умови для зорової діяльності з екраном у вертикальній площині під кутом 30 градусів від лінії зору, з площиною екрана, що перпендикулярна нормальній лінії зору користувача. Клавіатура розміщується

на поверхні столу або висувній полиці на відстані 100-300 мм від краю, що найближчий до користувача. Кут нахилу клавіатури повинен знаходитися в межах 5-15 градусів. Поверхня клавіатури має бути матовою із коефіцієнтом відбиття 0,4. Клавiші повинні бути комфортними для роботи і м'якими при натисканні (з ходом клавiш, що має бути однаковим із мінімальним опором натискання 0,25 Н та максимальним – не більше 1,5 Н).

При організації робочих місць з ПК слід дотримуватися вимог, викладених у ДНАОП 0.00-1.31-99: робочі місця розміщуються на відстані не менше 1 метра від стін з світловими прорізами; відстань між бічними поверхнями моніторів ПК повинна бути не менше 1,2 метра; відстань між тильною поверхнею монітора одного ПК та екраном монітора іншого ПК має бути не меншою 2,5 метра.

Загальні принципи організації робочого місця включають у себе відсутність зайвих предметів на робочому місці, розташування необхідних предметів поруч із працівником, упорядковане розміщення предметів відповідно до їх частоти використання, та забезпечення належної оглядовості.

Статичні напруження, що виникають у працівника під час виконання трудових завдань, пов'язані з утриманням нерухомих об'єктів та інструментів праці, а також з підтриманням стабільної робочої позиції.

Робоча поза визначає базове положення працівника у просторі: ергономічно зручна робоча поза повинна забезпечувати стійкість положення тіла, ніг, рук і голови під час виконання роботи, мінімізувати витрати енергії та максимізувати продуктивність праці. Невірна сидяча поза може викликати проблеми, такі як застій крові в ногах, а при великому обсязі роботи з пальцями рук може спричинити запалення суглобів.

Організація робочого місця користувача комп'ютера повинна гарантувати, що всі його елементи та їх взаємне розташування відповідають ергономічним вимогам, як зображено на рисунку 4.1.

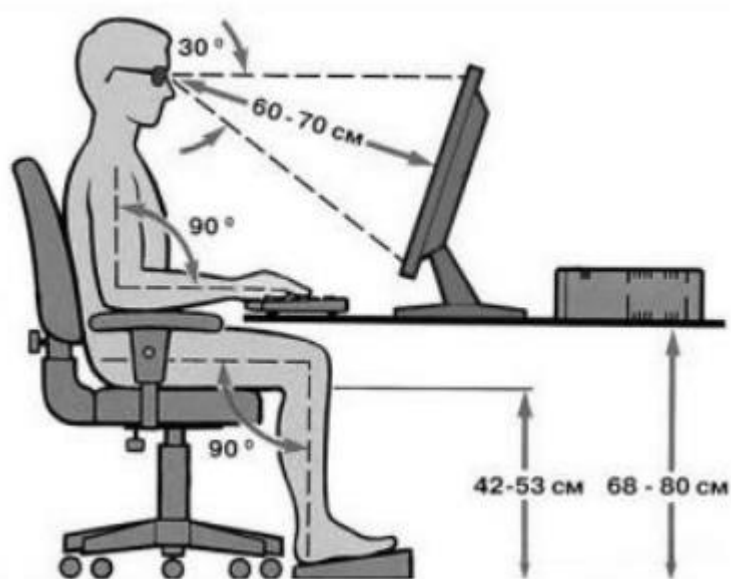


Рисунок 4.1 – Робоче місце і робоча поза користувача персонального комп'ютера

Найбільш поширеними положеннями тіла під час виконання роботи є сидяча та стояча позиції. При проектуванні робочого місця важливо враховувати, що виконання роботи з фізичним зусиллям більш практичне у стоячому положенні, тоді як для завдань з меншим фізичним навантаженням більш зручно використовувати сидячу позу. Стояча поза викликає більше втомлення, збільшуючи споживання енергії на 10%. Крім того, вона може призводити до підвищення артеріального та венозного тиску, розширення вен на ногах, пошкодження ступнів та викривлення хребта.

4.2 Заходи захисту від випромінювань оптичного діапазону

Випромінювання в оптичному діапазоні включає інфрачервоні та ультрафіолетові хвилі, видиме світло, а також лазерне випромінювання. Інфрачервоні промені обладнані хвильовими (з довжиною хвилі від 0,78 до 540 мкм) та квантовими властивостями. Генератором інфрачервоного випромінювання є будь-яке тіло з температурою вище абсолютного нуля. Згідно з законом Стефана-Больцмана, інтегральна густина випромінювання,

Вт/м², абсолютно чорного тіла, пропорційна четвертому ступеню його абсолютної температури.

З підвищенням температури тіла спостерігається зміна спектрального складу його випромінювання, причому чим вища температура, тим коротша довжина хвилі максимального випромінювання.

Інфрачервоне випромінювання, яке впливає на тіло людини, особливо ті його частини, що не захищені (лице, руки, шия, груди), конвективне тепло впливає на зовнішній шкіряний покрив, тоді як інфрачервоне випромінювання може проникнути на певну глибину в тканину. Довготривале перебування людини в зоні інфрачервоного випромінювання, так само як і систематично висока температура, може призводити до значного порушення теплового балансу в організмі. Для вимірювання густини потоку інфрачервоного випромінювання на робочому місці використовується актинометр – прилад, який дозволяє вимірювати густину потоку в інфрачервоному діапазоні від 0 до 14 кВт/м².

Основні методи захисту включають захист часом, захист віддалено, усунення джерел тепловиділення, теплоізоляцію, охолодження гарячих поверхонь, забезпечення тепловіддачі тіла людини та індивідуальні засоби захисту. Зменшення потужності випромінювання можливе за допомогою конструктивних та технологічних рішень (зміна методу нагріву об'єктів у печах індукційним нагріванням тощо) та покриття нагрівальних поверхонь теплоізоляційним матеріалом. Для захисту очей використовують світлофільтри із спеціального жовто-зеленого або синього скла.

Ультрафіолетове випромінювання впливає на склад виробничої атмосфери, створюючи озон, оксиди азоту і пероксид водню. Короткохвильове випромінювання іонізує повітря, формує ядра конденсації в атмосфері, які зменшують освітленість робочих місць і спричиняють утворення туманів.

Першочергові заходи для захисту включають конструкторські і технологічні рішення, спрямовані на уникнення генерації або зменшення інтенсивності випромінювання. Спеціальні засоби захисту, такі як

екранування джерел випромінювання і фарбування стін у світлі кольори, допомагають уникнути розповсюдження і знижують інтенсивність цих випромінювань у виробничих приміщеннях. Очі захищаються окулярами або щитками із світлофільтрами. Для захисту шкіри використовують мазі із речовинами, які фільтрують це випромінювання (салол, саліцилово-метиловий ефір тощо), а також спецодяг із бавовняних тканин і грубововняного сукна. Руки захищають рукавицями.

Діапазон довжин хвиль, які випромінюють оптичні квантові генератори (ОКГ), такі як лазери, охоплює видимий спектр і розповсюджується в інфрачервоній і ультрафіолетовій областях. Очі є особливо чутливими до дії випромінювання ОКГ, яке може викликати опіки і пошкодження сітківки, а в результаті - сліпоту. Небезпечно не лише пряме випромінювання, але і його відбите від стін та обладнання. Існують спеціальні норми, які включають організаційні та інженерно-технічні заходи, спрямовані на зменшення густини потоків енергії на робочих місцях до значень, значно менших за допустимі. ОКГ розміщують в окремих або відгороджених приміщеннях, які не мають дзеркальних поверхонь. Стіни, стелі, обладнання та інші предмети фарбують матовою фарбою з низькою сорбційною здатністю. Приміщення повинно мати високу освітленість і ефективну вентиляцію з припливно-витяжною системою. Якщо декілька ОКГ розташовані в одному приміщенні, їх відокремлюють ширмами, шторами або екранами, які не пропускають випромінювання.

Ефективний захист від випадкового опромінення людини забезпечується світловодом, який екранує промінь на всьому його шляху від ОКГ до мішені.

4.3 Охорона навколишнього природного середовища

Закон України «Про охорону навколишнього середовища» встановлює правові, економічні та соціальні засади для захисту навколишнього середовища.

Цей закон встановлює чіткі норми та вимоги, спрямовані на регулювання відносин у сфері охорони природи, забезпечення екологічної безпеки та управління природними ресурсами.

Правові аспекти закону включають в себе норми, які визначають права та обов'язки громадян, підприємств, органів влади у сфері охорони навколишнього середовища. Вони також регламентують відповідальність за порушення екологічних норм.

Економічні аспекти закону передбачають механізми фінансування заходів з охорони природи, впровадження ринкових інструментів для стимулювання сталого використання ресурсів та обмеження негативних впливів на навколишнє середовище.

Соціальні аспекти включають заходи для підвищення екологічної свідомості громадян, забезпечення доступу до інформації про стан навколишнього середовища та участь громадськості у процесах прийняття рішень з питань охорони природи.

Узагальнено, закон встановлює комплексні принципи, які сприяють балансу між виробництвом та захистом природи.

Основним завданням цього Закону є регулювання відносин у сфері охорони праці, управління та відновлення природних ресурсів, забезпечення екологічної безпеки, упередження та ліквідація негативних впливів на навколишнє середовище внаслідок людської діяльності, збереження природних ресурсів, генетичного фонду нації, ландшафтів та інших природних об'єктів.

Під час проведення науково-дослідницької роботи у приміщеннях виникають відходи, такі як зношені й відпрацьовані деталі, паперові відходи, тощо. Усі ці відходи передаються до господарського блоку для подальшої утилізації.

Суворі вимоги до розробки нових технологічних рішень сприяють зменшенню антропогенного навантаження на природне середовище.

На рисунку 4.2 наведено Модель системи управління навколишнім природним середовищем на підприємстві (відповідно до ДСТУ 14001).



Рисунок 4.2 - Модель системи управління оточуючим природним середовищем на підприємстві (згідно ДСТУ 14001) [1]

Модель системи управління навколишнім природним середовищем на підприємстві є складною та багатоплановою структурою, яка враховує вплив діяльності підприємства на екологію та спрямована на досягнення сталого розвитку. Така модель включає в себе різні аспекти, такі як відновлення ресурсів, зменшення викидів, управління відходами, використання енергії та інші параметри, що визначають вплив підприємства на навколишнє середовище.

Основним етапом в управлінні навколишнім середовищем є оцінка впливу діяльності підприємства. Це включає визначення екологічних аспектів та їх впливу на природне середовище. Модель повинна враховувати викиди, використання ресурсів, створення відходів та інші фактори.

На основі оцінки впливу формуються конкретні цілі та завдання системи управління. Це може включати в себе встановлення цілей зменшення викидів, використання енергії з відновлювальних джерел, покращення управління відходами та інші конкретні завдання.

Модель системи управління повинна передбачати впровадження екологічних стандартів і нормативів. Це може бути впровадження міжнародних стандартів ISO 14001 або інших аналогічних систем.

Ефективна система управління повинна включати в себе механізми моніторингу та звітності за результатами впровадження заходів. Моніторинг дозволяє вчасно виявляти відхилення від поставлених завдань та коригувати стратегію.

Управління навколишнім середовищем має бути вбудоване в корпоративну культуру підприємства. Залучення співробітників, їхній навчання та мотивація стають важливим елементом моделі.

Модель повинна передбачати механізми взаємодії з різними стейкхолдерами, такими як громадські організації, регулюючі органи та інші зацікавлені сторони.

Система управління навколишнім середовищем повинна бути гнучкою та здатною до постійного вдосконалення. Це включає в себе аналіз результатів, виявлення нових можливостей для зменшення негативного впливу та вдосконалення екологічної ефективності підприємства.

Отже, модель системи управління навколишнім природним середовищем на підприємстві є ключовою складовою сталого розвитку. Ефективна система дозволяє підприємству зменшити негативний вплив на навколишнє середовище, досягти екологічної ефективності та відповідати сучасним екологічним стандартам.

4.4 Висновки до четвертої частини

Ефективне планування робочого місця є ключовим елементом для забезпечення комфорту та підвищення продуктивності праці. Розташування інструментів та предметів праці повинно бути оптимальним, з урахуванням зручності та уникненням фізіологічно неприродних положень тіла.

Використання стандартів ергономіки, таких як ДСТУ 8604:2015, є важливим для організації робочого місця користувача ПК. Розташування обладнання, розміщення документів та правильна позиція монітора впливають на здоров'я та продуктивність працівника.

Згідно з ергономічними принципами, площа робочого місця повинна бути оптимальною, щоб уникнути зайвих рухів. Дослідження підтверджують, що раціональне організація робочих місць може призвести до зростання продуктивності праці на 15–25%.

Управління навколишнім середовищем має бути вбудоване в корпоративну культуру підприємства. Модель повинна враховувати взаємодію з різними стейкхолдерами та передбачати постійне вдосконалення з метою зменшення негативного впливу.

Система захисту включає в себе заходи для зменшення впливу факторів, таких як тепловиділення та випромінювання. Використання технологічних рішень та індивідуальних засобів захисту може сприяти створенню безпечного та здорового робочого середовища.

Система управління навколишнім середовищем повинна бути гнучкою та готовою до постійного вдосконалення. Це означає аналіз результатів, пошук нових можливостей для зменшення негативного впливу та підвищення екологічної ефективності підприємства.

Загальна організація робочого місця, врахування ергономічних принципів та система управління навколишнім середовищем є необхідними складовими для забезпечення здоров'я працівників, підвищення їхньої продуктивності та створення сталих умов праці.

5 ЕКОНОМІЧНА ЧАСТИНА

5.1 Розрахунок витрат на розробку технології

Давайте здійснимо прогнозування витрат, пов'язаних із виконанням науково-дослідної, дослідно-конструкторської та конструкторсько-технологічної роботи з розробки програмного забезпечення.

Прогнозування витрат на виконання цих видів робіт включає такі етапи:

1. Розрахунок витрат, що безпосередньо пов'язані із виконавцями даного розділу роботи.
2. Розрахунок загальних витрат на виконання цієї роботи.
3. Прогнозування загальних витрат на виконання та впровадження результатів цієї роботи.

Проведемо розрахунок витрат, які прямо пов'язані із виконавцями цього розділу роботи, використовуючи відповідні статті та формули. При цьому будемо враховувати, що для розробки інформаційної технології було залучено одного розробника програмного забезпечення. Розрахунок базового заробітку розробника-дослідника позначимо як « Z_0 ».

$$Z_0 = \frac{M}{T_p} \cdot t \quad (5.1)$$

де M – місячний посадовий оклад – 14 200 грн;

T_p – число робочих днів в місяці; приблизно $T_p = (22)$ дні;

t – розробник-дослідник працює протягом 64 робочих днів.

$$Z_0 = \frac{14200}{22} \cdot 64 = 41309 \text{ (грн.)}$$

2. Додаткова заробітна плата Z_d розраховується як 12% від основної заробітної плати:

$$З_д = 0,12 \cdot 41309 = 4957,08 \text{ (грн.)}$$

3. Нарахування на заробітну плату $H_{зп}$

$$H_{зп} = (З_о + З_д) \cdot \frac{\beta}{100} \text{ (грн)} \quad (5.2)$$

де $З_о$ – основна заробітна плата розробника

$З_д$ – додаткова заробітна плата розробника

β – ставка єдиного внеску на загальнообов'язкове державне соціальне страхування – 22%.

$$H_{зп} = (41309 + 4957,08) \cdot 0,22 = 10178 \text{ (грн.)}$$

4. Амортизація застосованого обладнання, комп'ютерів і приміщень під час виконання даного етапу роботи розглядається як відрахування, розраховане для кожного конкретного виду обладнання та приміщень.

У більш спрощеному вигляді, сумарні амортизаційні відрахування можуть бути обчислені за визначеною формулою.

$$A = \frac{Ц \cdot T}{12 \cdot T_b} \text{ [грн]}, \quad (5.3)$$

де $Ц$ – загальна балансова вартість обладнання, приміщення тощо, грн.;

T – фактична тривалість використання, міс;

T_b – термін використання обладнання, приміщень тощо, роки.

Розробка програмного забезпечення ведеться 2 місяці.

Під час створення програмного продукту були використані Інтернет-послуги на суму 450 грн, папір на 200 грн та канцтовари на 70 грн.

Загальні витрати складають: $450 + 200 + 70 = 720,00$ грн.

5. Витрати на силову електроенергію V_e розраховуються за формулою:

$$V_e = V \cdot П \cdot \Phi \cdot K_{\Pi} \text{ [грн.]}$$

де V – вартість 1 кВт-год;

$П$ – установлена потужність обладнання – 0,8 кВт;

Φ – фактична кількість годин роботи обладнання – 624 години;

K_{Π} – коефіцієнт використання потужності – 0,14

$$V_e = 1,88 \cdot 0,8 \cdot 624 \cdot 0,14 = 131,38 \text{ (грн.)}$$

6. До інших витрат включаються витрати на управління організацією, витрати на службові відрядження, витрати на утримання, ремонт і експлуатацію основних засобів, а також витрати на опалення, освітлення, водопостачання, охорону праці і інші подібні витрати. Вартість інших витрат, позначених як I_v , можна розглядати як 50% від суми основної заробітної плати розробника.

$$V_{\text{ін}} = 0,5 \cdot 41309 = 20695 \text{ (грн.)}$$

Сума всіх попередніх статей витрат дає витрати на виконання даної частини (розділу, етапу) роботи – V .

$$V = 41309 + 4957,08 + 10178 + 720 + 131,38 + 20695 = 77090,46 \text{ (грн.)}$$

Виконаємо передбачення сумарних витрат ($ЗВ$) на виконання та впровадження отриманих результатів наукової роботи.

Прогнозування виконується відповідно до такої формули:

$$ЗВ = \frac{V_{\text{заг}}}{\beta} \text{ [грн]},$$

Оскільки наукова робота завершена, $V_{\text{заг}} = V = 77090,46$ (грн)

$$ЗВ = \frac{77090,46}{0,7} = 110129,22 \text{ (грн.)}$$

Таким чином, передбачена сума загальних витрат (ЗВ) на виконання та впровадження результатів проведеної наукової роботи становить: 110129,22 (грн.)

У результаті проведення розрахунків витрат, пов'язаних із виконанням конкретного розділу роботи, ми врахували різні аспекти, такі як заробітна плата розробника, амортизація обладнання, витрати на Інтернет-послуги, папір, канцтовари, електроенергію та інші накладні витрати. Розглядаючи ці аспекти, ми також врахували той факт, що для розробки інформаційної технології залучено одного розробника програмного забезпечення.

Важливим кроком було також проведення прогнозування загальних витрат (ЗВ) на виконання та впровадження результатів наукової роботи. Відповідно до наших розрахунків, передбачена сума загальних витрат на цей етап становить 110129,22 грн. Це число включає в себе різноманітні складові, які враховують витрати на робочу силу, амортизацію, витрати на матеріали та послуги, а також інші витрати, пов'язані з виконанням наукової роботи.

Отже, усі ці розрахунки та прогнози важливі для ефективного управління ресурсами та забезпечення успішної реалізації наукового проекту.

5.2 Висновки до п'ятої частини

Розробка програмного прототипу для автоматичного розпізнавання емоцій на зображеннях та відео - це складний процес, щовимагає не лише технічних знань, але й урахування економічних та соціальних аспектів. Один із ключових аспектів такої роботи - визначення витрат, пов'язаних із виконанням конкретних етапів розробки.

Розглядаючи розділ про розрахунок витрат, слід врахувати, що витрати не обмежуються лише технічними параметрами. Вони також включають

заробітну плату, витрати на матеріали, послуги, амортизацію, електроенергію та багато іншого. Наприклад, відображення витрат на робочу силу, використання ресурсів та придбання послуг є ключовими компонентами оцінки витрат.

Економічна ефективність розробки в практиці визначається на основі витрат, пов'язаних з розробкою та вартості продукту. Це включає заробітну плату, витрати на матеріали, обладнання, послуги, а також управлінські та адміністративні витрати. Ключовою частиною оцінки є порівняння цих витрат з отриманими користями або прибутком від використання розробленого продукту.

Соціальна ефективність відображає вплив розробки на суспільство та його членів. Це може бути виміряно через поліпшення у якості життя, забезпечення нових можливостей, підвищення ефективності процесів або зменшення соціальних нерівностей.

У практичному контексті, економічна ефективність розробки оцінюється за рівнем витрат на виробництво та реалізацію продукту, порівняно з його прибутковістю або вартістю для організації. Соціальна ефективність, у свою чергу, аналізує вплив на спільноту та окремих користувачів, враховуючи, наприклад, покращення у доступі до нових технологій, поліпшення якості життя або зменшення нерівностей.

Зрозуміння і оцінка обох ефективностей допомагає приймати обґрунтовані рішення стосовно розробки продукту, визначає його вартість та соціальну цінність, а також впливає на його прийняття на ринку та користь для суспільства в цілому.

Сума загальних витрат (ЗВ) на виконання та впровадження результатів проведеної наукової роботи становить: 110129,22 (грн.)

ЗАГАЛЬНІ ВИСНОВКИ

У результаті проведеного дослідження та розробки і експериментальної апробації програмного прототипу можна стверджувати, що мета роботи досягнута і всі завдання реалізовані. Зокрема можна констатувати такі положення:

1. Проведений аналіз технологій розробки та дослідження алгоритмів для автоматичного розпізнавання емоцій на зображеннях та відео.

Проведено аналіз методів розробки та вивчення алгоритмів для автоматичного визначення емоцій на зображеннях та відео. Використано алгоритми для визначення контурів обличчя та підкреслення його форми та розміру. Використано методи маскуванню для виділення та ізоляції області, що відповідає обличчю, зробивши його більш помітним на зображенні чи відео. Використано алгоритми, які адаптуються до руху та змін розміру обличчя в кадрі відео. Ці алгоритми дозволяють ефективно та точно локалізувати обличчя на вхідних візуальних даних, що є ключовим етапом для подальшого аналізу та розпізнавання емоцій.

Застосування нейронних мереж (CNN) широко використовується в обробці зображень та відео, де просторові властивості грають важливу роль. Вони успішно впораються з розпізнаванням об'єктів, класифікацією зображень, виявленням облич, сегментацією зображень та іншими завданнями в області комп'ютерного зору. Рекурентні нейронні мережі (RNN), які здатні зберігати попередні стани для обробки нових вхідних даних, розвинуті архітектури, такі як Long Short-Term Memory (LSTM) і Gated Recurrent Unit (GRU), допомагають вирішити проблему зниклого градієнта.

Алгоритм лінійного усереднення використовується для обробки зображень, де для кожної точки обчислюється середнє значення сусідів. Цей метод ефективний та простий, застосовується до зображень різної природи. Однак, він може втратити деталізацію в областях з великою зміною яскравості

чи контрасту. Важливо враховувати оптимальний розмір локального вікна для усереднення.

У визначенні емоцій використовується пошук обличчя з використанням ознак Хаара та визначення окремих рис обличчя. Глибинне навчання застосовується для аналізу емоцій особи на зображенні.

Отже, використані алгоритми та методи є важливими для точного визначення та аналізу емоцій на вхідних візуальних даних.

2. Охарактеризували створення та опис алгоритмів для автоматичного розпізнавання емоцій на зображеннях та відео

Використання рекурентних нейронних мереж для автоматичного визначення емоцій на зображеннях та відео є підходом, що базується на використанні архітектур нейронних мереж, які здатні враховувати контекстуальні залежності в часі. Рекурентні нейронні мережі (RNN) представляють собою спеціальний тип нейронних мереж, який дозволяє моделювати послідовність даних та враховувати попередні інформаційні стани.

Модифікації RNN, такі як Long Short-Term Memory (LSTM) і Gated Recurrent Unit (GRU), використовуються для ефективного моделювання послідовностей з тривалими залежностями, зокрема в завданнях розпізнавання мови, перекладу та аналізу емоцій на зображеннях та відео.

3. Склали комплексний алгоритм автоматичного розпізнавання емоцій людини за зображенням її обличчя та подали його у вигляді структурно-функціональної схеми.

Алгоритм автоматичного розпізнавання емоцій людини за зображенням її обличчя виглядає наступним чином:

1. Завантаження необхідних бібліотека і моделей.
2. Ініціалізація моделей.
3. Запуск відеопотоку.
4. Обробка кадрів відеопотоку.

5. Розпізнавання обличчя.
 6. Підготовка обличчя до розпізнавання емоцій.
 7. Розпізнавання емоцій.
 8. Відображення результатів на кадрі.
 9. Виведення кадру.
 10. Закриття програми.
4. Розробили за складеним алгоритмом програмний прототип для автоматичного розпізнавання емоцій людини.

Для автоматичного розпізнавання емоцій на зображеннях та відео використано бібліотеки OpenCV для роботи з зображеннями та відео, а також бібліотеку для розпізнавання обличчя та емоцій.

Бібліотека OpenCV (Open Source Computer Vision Library) є відкритою бібліотекою комп'ютерного зору, яка забезпечує доступ до алгоритмів комп'ютерного зору та обробки зображень. Вона підтримує зчитування та запис зображень і відео, обробку та аналіз зображень, розпізнавання обличчя та об'єктів, відслідковування об'єктів, роботу з камерою та машинне навчання. У коді для автоматичного розпізнавання емоцій використано класифікатор обличчя HaarCascade, що забезпечує швидке та ефективне виявлення обличчя на зображеннях чи кадрах відео.

Основні етапи розпізнавання емоцій включають конвертацію кадру до відтінків сірого, виявлення обличчя за допомогою класифікатора HaarCascade, вирізання та підготовку області обличчя, розпізнавання емоцій та позначення їх на оригінальному кадрі.

Усі ці кроки об'єднані в функції `detect_emotion`, яка використовується в основному циклі обробки відео для кожного кадру. Крім того, використано список `emotion_labels` для призначення міток розпізнаним емоціям.

Усі ці техніки та бібліотеки дозволили створити ефективну систему автоматичного розпізнавання емоцій на зображеннях та відео яка може бути використана в різноманітних сценаріях, таких як аналіз емоцій у відеоблогах, визначення емоційної реакції тощо.

5. Експериментально апробували і дослідили роботу програмного прототипу для автоматичного розпізнавання емоцій людини.

Нами була проведена апробація (тестування) програмного комплексу яке представляло собою запровадження певного алгоритму для автоматичного розпізнавання емоцій на зображеннях та відео. У разі якщо людина виражала візуально емоцію, то програма з'єднана з відеокамерою визачала емоцію людини яку вона зображала.

ПЕРЕЛІК ПОСИЛАНЬ

1. Закон України «Про охорону навколишнього середовища», від 08.10.2023. URL: <https://zakon.rada.gov.ua/laws/show/1264-12#Text>
2. Автоматичне розпізнавання емоцій користувача для організації інтелектуального інтерфейсу // Електронний журнал «Молодіжний науковотехнічний вісник», 2013, № 9. С.7
3. Бабій М. С. Ідентифікація та розпізнавання емоційно-психічного стану людини за зображенням обличчя. Вісник СумДУ. 2013. С. 29.
4. Мелех Л. В. Безпека життєдіяльності та охорона праці. Навчальний посібник для здобувачів вищої освіти із галузі знань 08 «Право», спеціальність 081 «Право». Львів: Львівський державний університет внутрішніх справ, 2022. 219с.
5. Савченко А. С., Синельников О. О. Методи та системи штучного інтелекту: навч. посібник. К.: НАУ, 2017. 108-111 с.
6. Уляненко А. Л. Огляд методів автоматичного розпізнавання емоційного стану людини по зображенню // Вісник Національного технічного університету «ХПІ». Серія: Системний аналіз, управління та інформаційні технології, № 1(3) 2020. С.85-88
7. Хайкин С. Нейронные сети: полный курс. Киев: Вильямс. 2016. 1104 с.
8. Ян Сі. Автоматичні розпізнавання емоцій користувача для організації інтелектуального інтерфейсу // Електронний журнал «Молодіжний науково - технічний вісник», 2013. № 9.
9. A. Newell, K. Yang, and J. Deng, «Stacked hourglass networks for human pose estimation» in Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part VIII, ser. Lecture Notes in Computer Science, vol. 9912, Springer, 2016, pp. 483–499

10. Kahou S. EmoNets: Multimodal deep learning approaches for emotion recognition in video//Journal on Multimodal User Interfaces,2015.No.10.P. 99–111
11. Khabarlak K. Fast Facial Landmark Detection and Applications: A Survey. Journal of Computer Science & Technology, vol. 22, no. 1, pp. 12–41, 2022
12. Khorrami P. How Deep Neural Networks Can Improve Emotion Recognition on Video Data. 2016 IEEE International Conference on Image Processing (ICIP). Piscataway, IEEE Publ., 2016, pp. 619–623
13. Krizhevsky, A.; Sutskever, I.; Hinton, G. E. (2012). «Imagenet classification with deep convolutional neural networks» (PDF). Advances in Neural Information Processing Systems. 1: 1097–1105.
14. LeCun Y. Gradient-Based Learning Applied to Document Recognition// Proceedings of the IEEE.1998.Vol.86, no.11. P. 2278–2324
15. M. Suwa, N. Sugie and K. Fujimora «A preliminary note on pattern recognition of human emotional expression» in International Joint Conference on Pattern Recognition, pp. 408-410, 1978.
16. Nefian A., Hayes M., Hidden Markov model for face recognition // Image Processing, 1998. ICIP 98. Proceedings. 1998 International Conference on, 06 August 2002. P. 48-56
17. W. Li, Y. Lu, K. Zheng, et al., «Structured landmark detection via topology-adapting deep graph learning» in Computer Vision – ECCV 2020, Cham: Springer International Publishing, 2020, pp. 266-283
18. Z. Feng, J. Kittler, M. Awais, P. Huber, and X. Wu, «Wing loss for robust facial landmark localisation with convolutional neural networks» in 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Los Alamitos, CA, USA: IEEE Computer Society, Jun. 2018, pp. 2235–2245.
19. Емоційний AI: як технологія набуває людських рис. URL: <https://evergreens.com.ua/ua/articles/emotion-ai.html>
20. A Gentle Introduction to Transfer Learning for Deep Learning. URL: <https://machinelearningmastery.com/transfer-learning-for-deep-learning/>

21. AffectNet database. URL: <http://mohammadmahoor.com/affectnet/> - Назва з екрану.
22. ASCERTAIN Dataset. URL: <https://ascertain-dataset.github.io/>
23. Bartlett M. S. Machine learning methods for fully automatic recognition of facial expressions and facial actions. URL: <https://ieeexplore.ieee.org/abstract/document/1398364>.
24. Emotic dataset. URL: <http://sunai.uoc.edu/emotic/>
25. Habr. OpenCV & Python. URL: <https://habr.com/ru/post/519454/>
26. Image augmentation for machine learning experiments. URL: <https://github.com/aleju/imgaug>
27. Iryna Sydorenko. AI in Emotion Recognition: Does It Work? URL: <https://labeledyourdata.com/articles/ai-emotion-recognition>
28. OpenCV – Overview URL.: <https://www.geeksforgeeks.org/opencv-overview/>
29. OpenCV. Docs. URL: <https://docs.opencv.org/4.x/>
30. OpenCV. Python tutorial. URL: https://docs.opencv.org/4.x/d6/d00/tutorial_py_root.html
31. Python. Library Reference. URL: <https://docs.python.org/3/library/index.html>
32. Python. Python Docs. URL: [<https://docs.python.org/3/>
33. Recurrent neural network. URL: https://en.wikipedia.org/wiki/Recurrent_neural_network
34. Ronneberger O. U-Net: Convolutional Networks for Biomedical Image Segmentation. URL: <https://arxiv.org/pdf/1505.04597.pdf>.
35. Rosebrock A. Facial landmarks with dlib, OpenCV, and Python. URL: <https://www.pyimagesearch.com/2017/04/03/facial-landmarks-dlib-opencv-ython/>.
36. Serdar Yegulalp. What is TensorFlow? The machine learning library explained URL: <https://www.infoworld.com/article/3278008/what-is-tensorflow-themachine-learning-library-explained.html>

37. SourceForge. OpenCV Library. URL: <https://sourceforge.net/projects/opencvlibrary/>
38. The Python Package Index (PyPI). URL: <http://omzsoftware.com/pythonista/docs/distutils/packageindex.html>
39. What is TensorFlow? URL: https://www.simplilearn.com/tutorials/deeplearning-tutorial/what-is-tensorflow#what_is_tensorflow

Лістинг програми

```
import cv2
from tensorflow.keras.models import load_model
import numpy as np

# Завантаження моделі для розпізнавання обличчя та емоцій
face_cascade = cv2.CascadeClassifier(cv2.data.harcascades +
'haarcascade_frontalface_default.xml')
emotion_model = load_model('path/to/emotion_model.h5')

# Список емоцій
emotion_labels = ['Angry', 'Disgust', 'Fear', 'Happy', 'Sad', 'Surprise', 'Interest']

# Функція для розпізнавання емоцій на кадрі
def detect_emotion(frame):
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
    faces = face_cascade.detectMultiScale(gray, scaleFactor=1.1, minNeighbors=5,
minSize=(30, 30))

    for (x, y, w, h) in faces:
        face_roi = gray[y:y + h, x:x + w]
        face_roi = cv2.resize(face_roi, (48, 48))
        face_roi = face_roi / 255.0
        face_roi = np.expand_dims(face_roi, axis=0)
        face_roi = np.expand_dims(face_roi, axis=-1)

        emotion_prediction = emotion_model.predict(face_roi)
        emotion_label_arg = np.argmax(emotion_prediction)
        emotion_label = emotion_labels[emotion_label_arg]
```

```

        cv2.putText(frame, emotion_label, (x, y - 10),
cv2.FONT_HERSHEY_SIMPLEX, 0.9, (0, 255, 0), 2, cv2.LINE_AA)
        cv2.rectangle(frame, (x, y), (x + w, y + h), (0, 255, 0), 2)

    return frame

# Відкриття відеофайлу або веб-камери
cap = cv2.VideoCapture('path/to/video.mp4') # або 0 для веб-камери

while cap.isOpened():
    ret, frame = cap.read()
    if not ret:
        break

    frame = detect_emotion(frame)

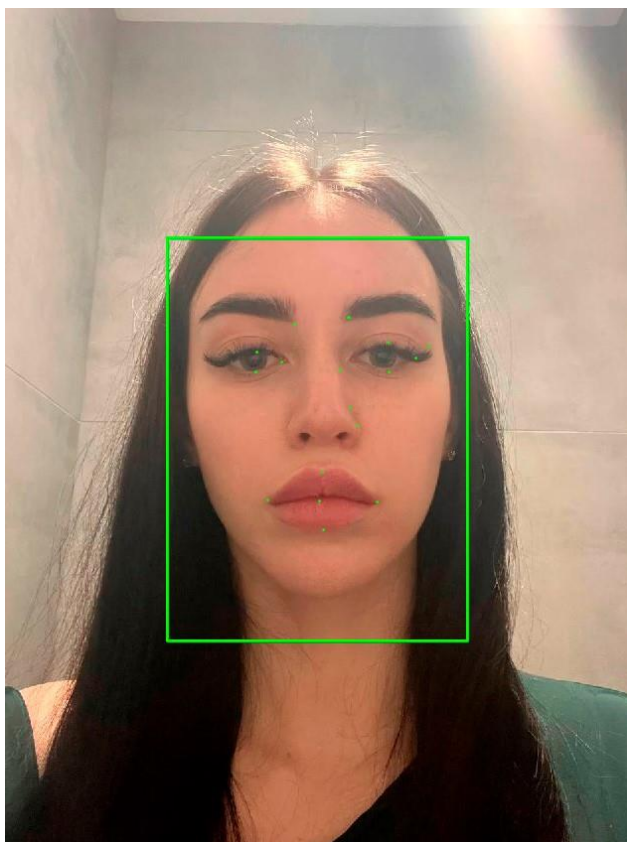
    cv2.imshow('Emotion Detection', frame)

    if cv2.waitKey(1) & 0xFF == ord('q'):
        break

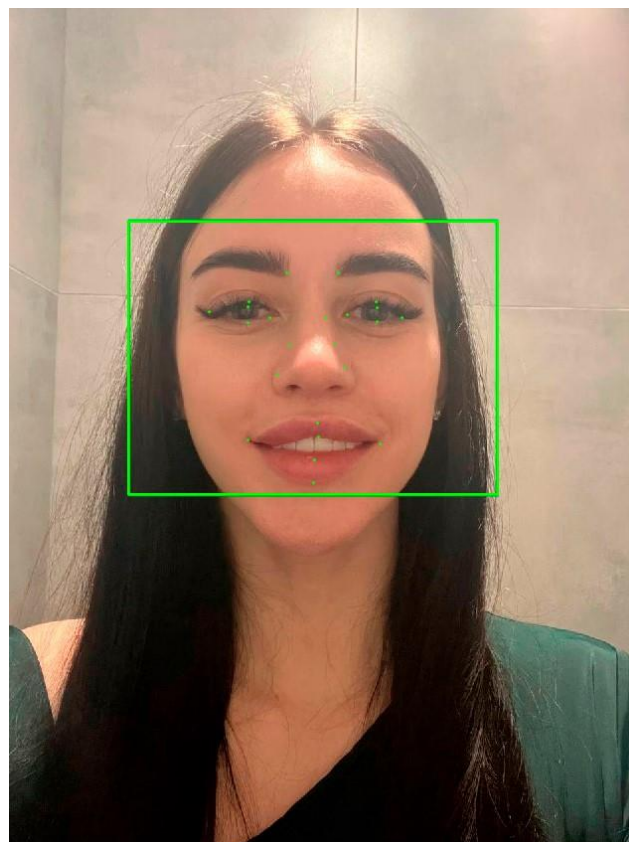
cap.release()
cv2.destroyAllWindows()

```

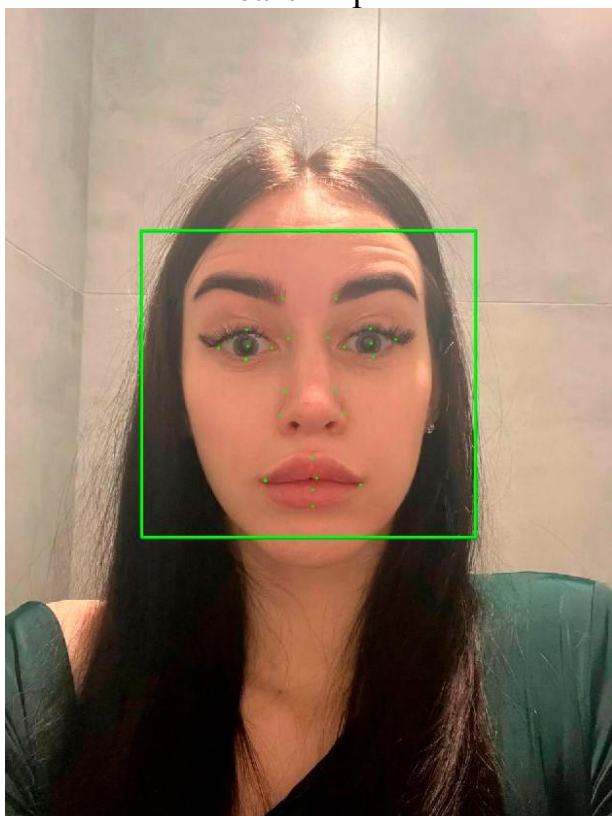
Neutral / Нейтральність



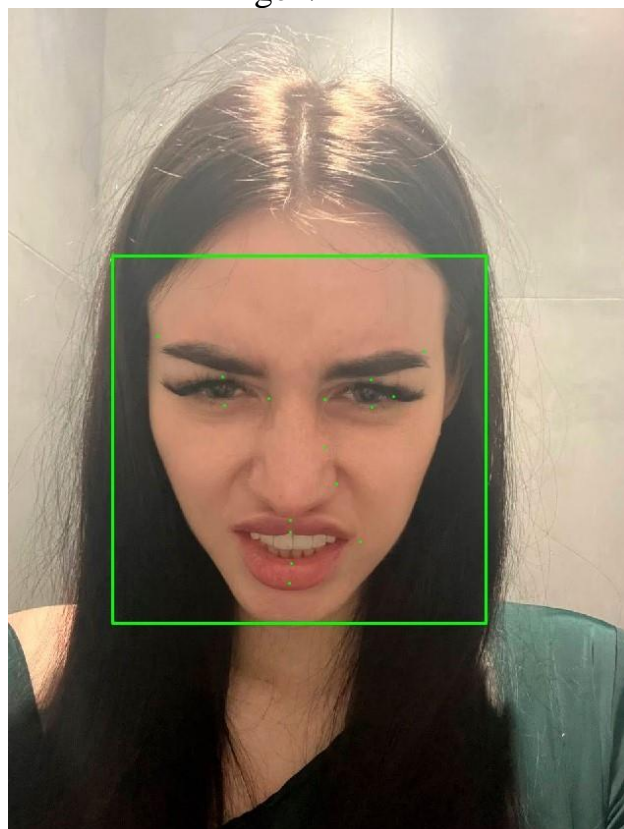
Happy / Радість



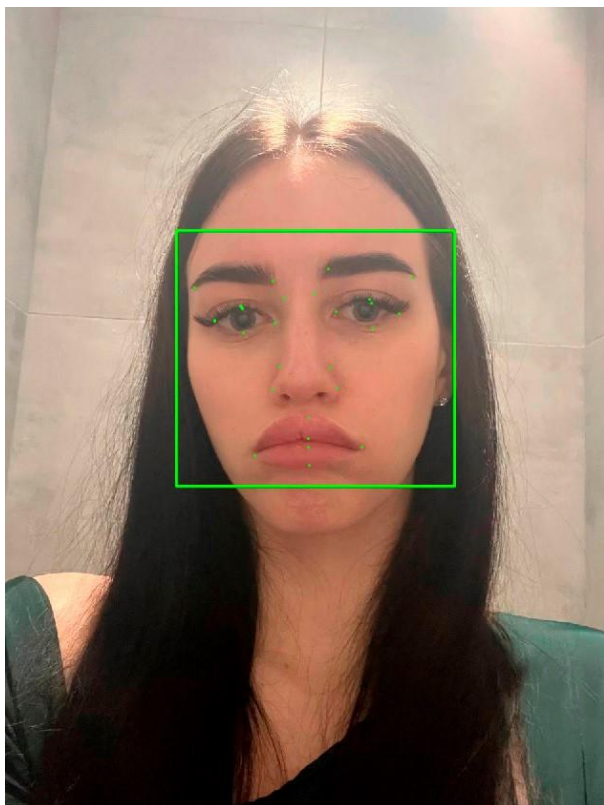
Fear / Страх



Anger / Злість



Sad / Сум



Surprise / Раптовість

