

Міністерство освіти і науки України

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «КИЄВО-МОГИЛЯНСЬКА АКАДЕМІЯ»

Кафедра мультимедійних систем факультету інформатики



ТРИВИМІРНИЙ ПРОСТІР ВЕБ-ДИЗАЙНУ: COMPUTER VISION

Виконав студент 4 курсу
БП «Інженерія програмного забезпечення»
Жулкевський Владислав Дмитрович

Керівник курсової роботи
канд. фіз-мат. наук, доцент,
Афонін Андрій Олександрович

Київ 2021

ІНДИВІДУАЛЬНЕ ЗАВДАННЯ

Міністерство освіти і науки України
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «КИЄВО-МОГИЛЯНСЬКА АКАДЕМІЯ»
Кафедра мультимедійних систем факультету інформатики

ЗАТВЕРДЖУЮ

Зав. кафедри мультимедійних
систем, доцент, к.ф.-м.н.

_____ О.П.Жежерун
„_____” _____ 2021 р.

ІНДИВІДУАЛЬНЕ ЗАВДАННЯ

на курсову роботу

студенту 4-го курсу, факультету інформатики

Жулкевському Владиславу Дмитровичу

Тема: Тривимірний простір веб-дизайну: Computer Vision

Зміст ТЧ до курсової роботи:

Зміст

Анотація

Вступ

1 Механізм розпізнавання емоцій на основі Computer Vision

2 Алгоритми машинного навчання для розпізнавання емоцій

3 Розробка програмного прототипу з використанням функціоналу
розпізнавання емоцій

Висновки

Список літератури

Додатки

Дата видачі „_____” _____ 2021 р. Керівник _____
(підпис)

Завдання отримав _____
(підпис)

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапу курсового проекту (роботи)	Термін виконання
1.	Отримання завдання на курсову роботу	15.10.2020
2.	Огляд літератури за темою роботи	19.10.2020- 24.12.2020
3.	Перевірка існуючих засобів машинного навчання та computer vision для визначення емоцій за фотографіями та відео	11.01.2021- 01.02.2021
4.	Ініціалізація та розробка серверної частини з функціоналом розпізнавання емоцій	02.02.2021- 21.02.2021
5.	Створення функціоналу запису відео користувача та надсилання запитів на серверну частину	14.02.2021- 01.03.2021
6.	Тестування та виправлення помилок прототипу	22.02.2021- 19.03.2021
7.	Написання розділу «Механізм розпізнавання емоцій на основі Computer Vision»	22.03.2021- 29.03.2021
8.	Написання розділу «Алгоритми машинного навчання для розпізнавання емоцій»	29.03.2021- 05.04.2021
9.	Написання розділу «Розробка програмного прототипу з функціоналом розпізнавання емоцій»	05.04.2021- 08.04.2021
10.	Створення презентації	08.04.2021- 09.04.2021
11.	Остаточне оформлення пояснювальної роботи та презентації	10.04.2021- 11.04.2021

12.	Подання роботи на кафедру для перевірки на плагіат	12.04.2021
13.	Захист курсової роботи	19.04.2021 - 25.04.2021

ЗМІСТ

ІНДИВІДУАЛЬНЕ ЗАВДАННЯ	2
КАЛЕНДАРНИЙ ПЛАН.....	3
АНОТАЦІЯ	6
ПЕРЕЛІК ТЕРМІНІВ ТА УМОВНИХ ПОЗНАЧЕНЬ	7
ВСТУП.....	8
1 МЕХАНІЗМ РОЗПІЗНАВАННЯ ЕМОЦІЙ НА ОСНОВІ COMPUTER VISION.....	11
1.1 Опис предметної області	11
1.2 Основні концепції розпізнавання обличчя та емоцій	12
1.3 Алгоритми попередньої обробки зображень.....	14
1.4 Алгоритми комп'ютерного зору для знаходження обличчя та його рис на зображенні.....	18
1.5 Знаходження ключових точок та розпізнавання емоцій	22
2 АЛГОРИТМИ МАШИННОГО НАВЧАННЯ ДЛЯ РОЗПІЗНАВАННЯ ЕМОЦІЙ	25
2.1 Алгоритм AdaBoost.....	25
2.2 Гістограма напрямлених градієнтів	27
2.3 Support Vector Machine	28
3 РОЗРОБКА ПРОГРАМНОГО ПРОТОТИПУ З ВИКОРИСТАННЯМ ФУНКЦІОНАЛУ РОЗПІЗНАВАННЯ ЕМОЦІЙ	31
3.1 Основні засоби розробки.....	31
3.2 Реалізація прототипу	32
ВИСНОВКИ	35
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	36
ДОДАТКИ	38
Додаток А	38
Додаток Б.....	39
Додаток В	40
Додаток Г.....	43

АНОТАЦІЯ

У цій роботі досліджується вплив технологічних рішень на досвід користувачів. Для того, щоб прийняти рішення щодо оновлення сторінки, відбувається розпізнавання емоції людини в конкретний момент часу з використанням Computer Vision та машинного навчання. За допомогою отриманої інформації генеруються нові елементи на сторінці (шрифт, кольори, картинки і т.д.) та переформатовується сама сторінка відповідно до настрою цієї людини. В результаті цієї курсової роботи, буде отримано інструмент, який надаватиме змогу аналізувати емоції відвідувачів веб сайту і використовувати їх для покращення UI/UX застосунків та зміни поведінки користувачів не тільки в майбутньому, а робити це динамічно в реальному часі.

ПЕРЕЛІК ТЕРМІНІВ ТА УМОВНИХ ПОЗНАЧЕНЬ

Ендпоінт – конкретний URL для прийому та обробки запиту на серверній частині.

Слабкий класифікатор – класифікатор, який не може самостійно визначити об'єкт, тому його комбінують разом із іншими слабкими класифікаторами для утворення сильного класифікатору.

Сильний класифікатор – класифікатор, який здатен самостійно визначити належність об'єкту до певної групи об'єктів.

FrontEnd – клієнтська частина застосунку.

BackEnd – серверна частина застосунку.

FormData – набір даних з HTML-форми.

Скріншот – знімок екрана.

UI (інтерфейс користувача) – ряд візуальних елементів сторінок, що дозволяють людині взаємодіяти з продуктом чи послугою.

UX (досвід користувача) – це враження, які має людина, взаємодіючи з усіма аспектами певного продукту чи послуги.

Action Vuex – метод, який взаємодіє зі сховищем Vuex, замінюючи дані стану застосунку за архітектурою Flux.

Фреймворк - це програмне середовище, своєрідний каркас, який використовується для того, щоб істотно полегшити процес об'єднання певних компонентів при створенні програм.

Токен – унікальний ідентифікатор, представлений у вигляді стрічки.

Promise - об'єкт, який представляє можливе успішне завершення (або невдачу) асинхронної операції та її результуюче значення.

ВСТУП

Активний розвиток технологій штучного інтелекту призвів до того, що ці технології наразі широко застосовуються в багатьох областях. Комп'ютер як обчислювальна машина здатний швидко і точно оброблювати задані йому параметри та дані для виконання певної задачі. Таким чином застосування технологій штучного інтелекту дає змогу покращити людське життя і допомагає виконувати завдання за межами людських здібностей. Штучний інтелект - це комплексна наука, яка має багато складових дисциплін. Одною з таких дисциплін є Computer Vision (комп'ютерний зір), який фокусується на наданні комп'ютерам можливості ідентифікувати та обробляти об'єкти на зображеннях та відео так само, як це роблять люди. Автоматична ідентифікація образів (тексту, обличчя, предметів тощо) за допомогою програмного забезпечення є одним з важливих напрямків розвитку технологій штучного інтелекту. З іншого боку останніми роками світ активно рухається у напрямку того, щоб взаємодія користувачів із програмним забезпеченням ставала більш органічною. При цьому перевагу надають тим застосункам, які викликають позитивні відчуття. Для досягнення цих цілей приділяється увага до таких складових програмного забезпечення, як UI/UX дизайн. Саме тому мета цієї роботи полягає в тому, щоб навчитись миттєво реагувати на зміну емоцій та змінювати поведінку користувачів. Це, зокрема, може бути досягнуто шляхом динамічної зміни контенту на веб сторінці відповідно до настрою користувача.

Мета дослідження полягає у ознайомленні із механізмами розпізнавання емоцій за фото людей, принципами та специфікою цього механізму, а також у створенні серверної частини з функціоналом розпізнавань емоцій для надання інформації про емоції людини у конкретний час користування застосунком. Основними завданнями даного дослідження є:

- висвітлення основних концепцій розпізнавання емоцій на обличчі за допомогою Computer Vision;

- аналіз алгоритмів машинного навчання під час виявлення рис обличчя та емоцій людини;
- створення серверної частини веб-застосунку з використанням функціоналу розпізнавання емоцій.

Об'єктом дослідження є алгоритм розпізнавання емоцій та принципи, за якими UI/UX впливає на настрої користувача.

У цій роботі дослідження проводилися шляхом аналізу документації використаних бібліотек, фреймворків, модулів, перегляд статей за темою дослідження, порівняння різних підходів до розпізнавання емоцій. Також було розроблено застосунок на основі висвітлених даних у цій роботі. Створення схем відбувалося з допомогою онлайн-сервісу для побудови діаграм та схем Lucidchart.

До джерел дослідження відносяться такі електронні ресурси як наукові статті з теми дослідження та документація опрацьованих у цій роботі бібліотек. Усі ресурси зазначені у списку використаних джерел.

Курсова робота складається з 3 розділів.

Перший розділ висвітлює алгоритм розпізнавання емоцій та основні складові цього алгоритму з боку комп'ютерного зору. У підрозділах розглядаються алгоритми попередньої обробки зображень та алгоритм пошуку ознак Хаара для знаходження рис обличчя.

В другому розділі наведено алгоритми машинного навчання, які використовуються під час пошуку ознак Хаара та класифікації емоцій користувача. Зокрема, показано як працюють AdaBoost та SVM алгоритм зсередини, а також продемонстровано гістограму напрямлених градієнтів, яка лежить в основі алгоритму SVM для класифікації емоцій.

У третьому розділі наводиться реалізація серверної частини прототипу відстежування емоцій користувачів. Висвітлюється потік даних та реалізація зчитування емоцій з відеопотоку браузера. Описуються використані технології та наводяться висновки дослідження.

У результаті виконання дослідження та написання курсової роботи створено прототип програмного продукту, який призначений для відображення різноманітного контенту веб-сторінок за настроєм користувача.

1 МЕХАНІЗМ РОЗПІЗНАВАННЯ ЕМОЦІЙ НА ОСНОВІ COMPUTER VISION

1.1 Опис предметної області

Наразі розробка програмних продуктів рухається у сторону того, аби зробити користування цими продуктами якомога комфортнішим. Чим краще продуманий дизайн застосунку для користувачів, тим більша ймовірність, що користувач буде ним користуватися. Зосередження на дизайні також допомагає покращити широкий діапазон бізнес-показників програмного продукту, таких як утримання користувачів, їх лояльність та продуктивність. Разом з цим, досвід всесвітньовідомої компанії Google показав, наскільки велику роль дизайн відіграє при розробці програмного забезпечення. У 2014 році компанія змінила відтінок синього кольору, який використовується в рекламних посиланнях та під час пошуку, що в результаті принесло компанії додаткові двісті мільйонів доларів доходу на рік [1]. Працівники компанії провели дослідження який із відтінків синього найбільше подобається звичайним користувачам їхнього пошуковика за допомогою техніки А/В тестування. Це техніка, яка полягає в тому, що користувачу надається два варіанти схожого продукту і вони обирають той, який їм найбільше подобається.

У ході дослідження впливу дизайну на користувачів, у роботі Ольги Кірдяєвої був проведений детальний аналіз як користувацький інтерфейс та користувацький досвід впливає на емоції людини. В результаті аналізу авторкою виокремлено наступні висновки:

- a) Анімація та інтерактивні елементи викликають захват.
- b) Забагато тексту на сторінці збиває з пантелику та заважає сприйняттю.
- c) Яскраві кольори привертають увагу, але при цьому відштовхують, якщо їх забагато.
- d) Зображення людей або будь-які знайомі образи викликають зацікавленість і позитивну реакцію.

- e) Незвичні візуальні рішення привертають увагу, проте можуть заважати сприйняттю та викликати негативні емоції.
- f) Усвідомлено чи ні, нам хочеться повторювати те, що ми бачимо [2].

На основі даних висновків було вирішено розробити застосунок, який зможе впливати на UI/UX аспекти, спираючись на настрій користувача та намагаючись водночас вплинути на нього. За основу застосунку було обрано макет сторінок сайту інтернет-магазину «Rozetka». В прототипі ми поставили задачу реалізувати зміну об'єктів інтерфейсу до емоцій користувача. Згідно з цим, в залежності від того, які емоції переживає користувач, ми будемо генерувати такі елементи інтерфейсу, аби допомогти покращити його настрій та вплинути на його подальшу поведінку.

1.2 Основні концепції розпізнавання обличчя та емоцій

Автоматичне розпізнавання емоцій дозволяє надати можливість оцінити настрій людини, а для цього спочатку потрібно вирішити наступні задачі:

1) Захоплення та відстежування обличчя. У цій задачі потрібно визначити чи присутнє обличчя людини на фотографії чи відео. Якщо обличчя дійсно є, то програма повинна отримати розмір обличчя та координати його положення на зображенні. В той час відстежування продовжується для подальшого визначення розміру та положення обличчя, які можуть змінюватися протягом часу.

2) Розпізнавання та опис особливостей обличчя. Ця задача вимагає визначення положення рис обличчя, таких як очі, ніс, рот, брови тощо, а також визначення точок та форм цих рис.

3) Класифікація. Відповідно до інформації про риси обличчя, визначеній у другій задачі, необхідно встановити яка емоція присутня на обличчі конкретної людини.

Аналіз виразів обличчя є непростим напрямком досліджень в області розпізнавання образів, в основному через складність отримання точних ознак обличчя та його виразів. Ця складність спричинена невеликою розбіжністю у

характерних особливостях між різними виразами обличчя, наприклад, відкритий рот означає, що це може бути як і плач, так і здивування. Існуючі методи розподілу використовують три характерні ознаки для розпізнавання: рівень сірого кольору, рухи окремих точок та частоти зображення. Різні емоції приводять до різних відтінків сірого кольору на цифровому зображенні обличчя користувача. Однак аналіз цієї характерної ознаки вимагає серйозної попередньої цифрової обробки для нормалізації вхідного зображення. Рух визначає інформацію про зміну положення точок певної риси на обличчі. Частоти використовуються для визначення різниці між різними цифровими зображеннями обличчя в частотному регіоні. Методи, вказані вище, розвивалися з методів розпізнавання обличчя, але застосовуються і для розпізнавання емоцій людини.

З іншого боку, методи розпізнавання можна також розділити на три області: цілісні і локальні розпізнавання, вилучення деформації та рухів, геометричні характеристики і характеристика зовнішності. Методи цілісних розпізнавань аналізують емоції людського обличчя в цілому, щоб потім знайти різницю між різними зображеннями. Така техніка застосовується у наступних методах розпізнавання: Independent Component Analysis (ICA), Local Feature Analysis (LFA), Fisher Actions, Principal Component Analysis (PCA), Hidden Markov Models, Fisher's Linear Discriminants (FLD), та кластерному аналізі. Методи локальних розпізнавань аналізують окремі частини обличчя, наприклад рот, брови і очі. Типові методи: Facial Actions Code System (FACS), Local PCA, Вейвлет-перетворення і нейронні мережі. Методи вилучення деформації і рухів для розпізнавання беруть за основу зміни рис обличчя, коли змінюються різні вирази: Active Shape Model (ASM), Point Distribution Model (PDM). Методи геометричних характеристик засновані на формі і положенні різних частин обличчя людини, ці методи допомагають витягти характерні вектори, які представляють геометричні характеристики обличчя. Хоч і усі вказані методи дають можливість розпізнати певні характеристики обличчя чи емоції людини, однак вони не завжди дозволяють зробити це в реальному часі з достатнім рівнем

точності на комп'ютері з невеликими обчислювальними ресурсами. Саме тому був розроблений наступний метод розпізнавання емоцій користувача [3].

Процес розпізнавання емоцій на основі цього методу ділиться на наступні етапи (рисунок 1.1):

1. Спочатку зчитуємо зображення користувача та оброблюємо його засобами доступних бібліотек для покращення якості знаходження обличчя та його рис;
2. Визначаємо координати положення обличчя на зображенні.
3. Знаходимо характерні риси обличчя, такі як очі, ніс, губи, брови тощо;
4. Знаходимо ключові точки кожної риси обличчя, які були знайдені та класифіковані на попередньому етапі;
5. Визначаємо якій емоції притаманні окремі ключові точки та отримуємо саму емоцію.

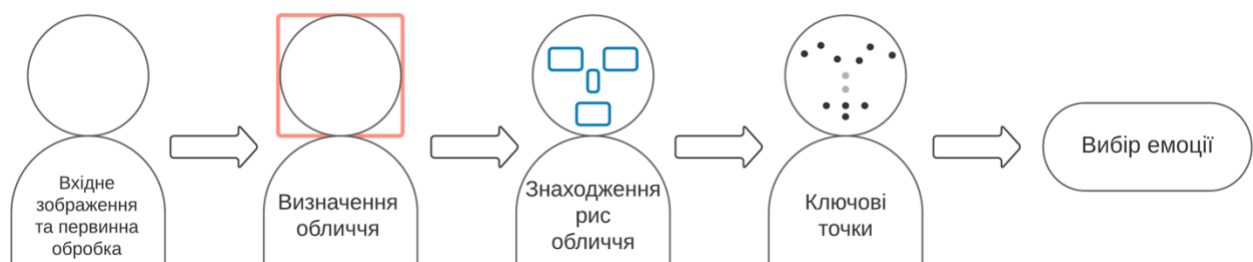


Рисунок 1.1 - Алгоритм розпізнавання емоції на зображенні

У наступних підрозділах буде окремо проаналізовано кожен із наведених етапів.

1.3 Алгоритми попередньої обробки зображень

Перший етап є важливим для того, аби отримати картинку та провести певні зміни з нею. Ці зміни допомагають поліпшити якість зображення та максимально адаптувати це зображення для автоматичного аналізу. Наразі існує

безліч алгоритмів попередньої обробки зображень, проте у роботі хочу навести наступні три алгоритми:

- а) Лінійне усереднення сусідніх точок - найпростіший вид алгоритмів попередньої обробки, який використовується для видалення шуму із зображення. Основна ідея алгоритму полягає в тому, щоб знайти середнє арифметичне значення точок в деякій області в якості нового значення точки [4].

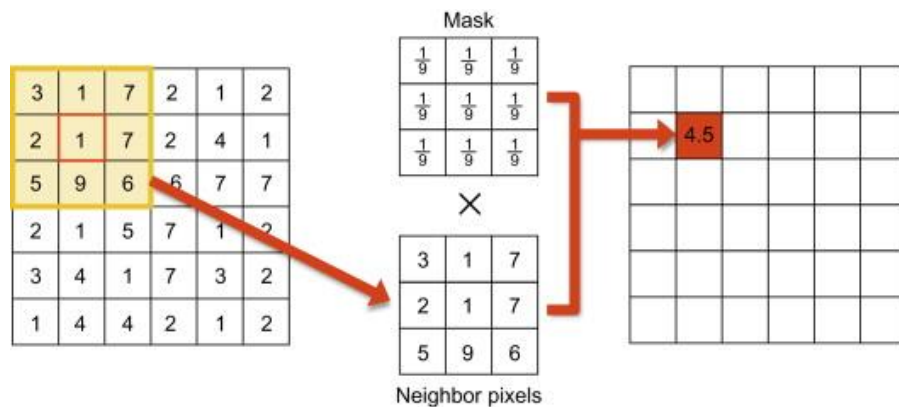


Рисунок 1.2 - Застосування лінійного усереднення на зображенні

Така фільтрація реалізується за допомогою обходу пікселів зображення матрицею згортки, що має вигляд, представлений на рис. 1 (підпис “Mask”). Також різновидом лінійного усереднення є розмиття за Гаусом, який використовує наступну формулу для створення матриці згортки -

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}, \quad (1)$$

де G – результат функції, який буде вписуватися в матрицю вибраної користувачем розмірності,

x - це відстань від початку координат в осі абсцис,

y - це відстань від початку координат у осі ординат,

σ - стандартне відхилення розподілу Гауса, яка задає силу розмиття. Чим більше відхилення було обрано, тим сильніше розмиття.

Для прикладу, саме так буде виглядати маска 3x3 та значенням відхилення 1 (рис. 1.3).

0.077847	0.123317	0.077847
0.123317	0.195346	0.123317
0.077847	0.123317	0.077847

Рисунок 1.3 - Маска 3x3 за функцією Гауса з значенням відхилення 1

Таким чином, обравши цю маску та застосувавши її до кожного пікселя зображення, можна також досягнути розмиття та позбавлення шуму, як і було наведено вище для звичайного лінійного усереднення. Цей алгоритм краще позбавляється шуму на зображенні, проте за великих розмірностей матриць або великого значення відхилення можливе сильне розмиття контурів обличчя, чого не варто допускати.

- б) Лінійне розтягування кольорової гістограми зображення – метод, який допомагає здійснити корекцію контрастності зображення. Відтак рівням вхідного зображення, що лежать в інтервалі найменшого та найбільшого значення яскравості зображення, можна присвоїти нові значення з метою охопити весь можливий інтервал змін яскравості, тобто з значення 0 до 255 [5].

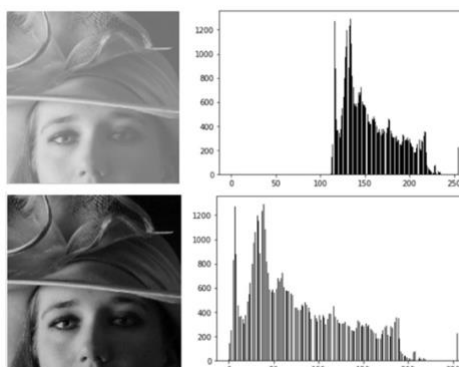


Рисунок 1.4 - Різниця у контрастності зображення після лінійного розтягування гістограми яскравості

- с) Алгоритм нарощування областей – алгоритм, який використовується для сегментації зображення. В основі цього алгоритму лежить ідея, коли спочатку обирається точка відліку певної області та відбуватиметься нарощування цієї області зображення за допомогою перевірки сусідніх пікселів на однорідність (рисунок 1.5). Таким чином, якщо даний піксель та піксель-сусід мають малу різницю у яскравості, то вони відносяться до однієї спільної області. Мета алгоритму - розподіл зображення на окремі частини для подальшого використання, наприклад, для визначення до якого виду належать об'єкти.

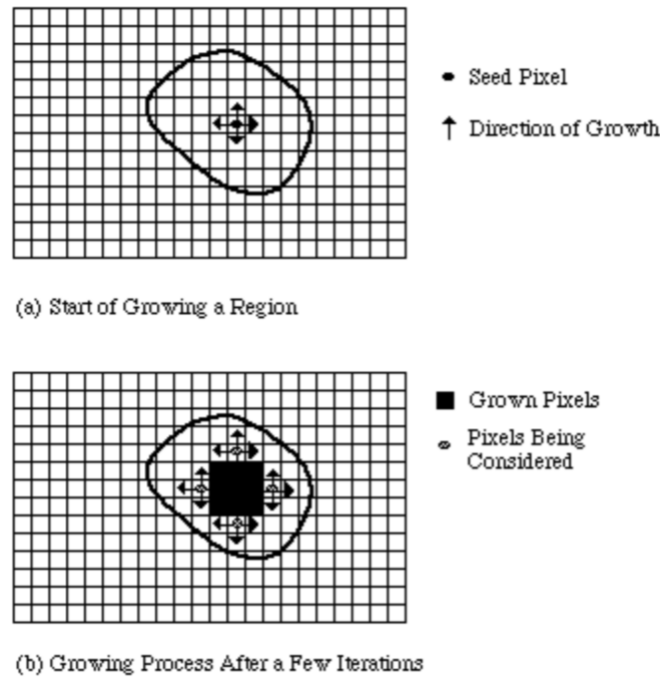


Рисунок 1.5 - Алгоритм нарощування областей

Усі наведені алгоритми реалізовані у бібліотеці OpenCV, і ми можемо скористатися наступними функціями для виконання вище наведених алгоритмів:

- `cv.GaussianBlur` – для видалення шуму на зображенні;
- `cv.equalizeHist` – для корекції контрастності зображення;
- `cv.watershed` – для сегментації зображення.

1.4 Алгоритми комп'ютерного зору для знаходження обличчя та його рис на зображенні

На другому етапі алгоритму розпізнавання емоцій відбувається пошук обличчя користувача за допомогою характерних ознак Хаара. Такий самий алгоритм використовується і на третьому етапі для знаходження окремих рис обличчя. Сам алгоритм можна поділити на чотири етапи:

а) Розрахунок ознак Хаара

В основі цього розрахунку лежить метод Віюлі-Джонса, які запропонували ввести так звані примітиви Хаара. Ці примітиви представляють собою

розбиття певної прямокутної області на різні прямокутні підобласті. Розрахунок включає підсумовування інтенсивності пікселів у кожній підобласті та обчислення різниці між сумами. Різниця значень підобластей стає значенням цього примітиву. Це дозволяє замінити звичайну піксельну систему зображення на систему на основі ознак Хаара. Під час майбутніх обчислень, ми використовуватимемо саме таку систему для виділення обличчя та його характеристик. Приклади примітивів Хаара можна побачити на рисунку 1.6.

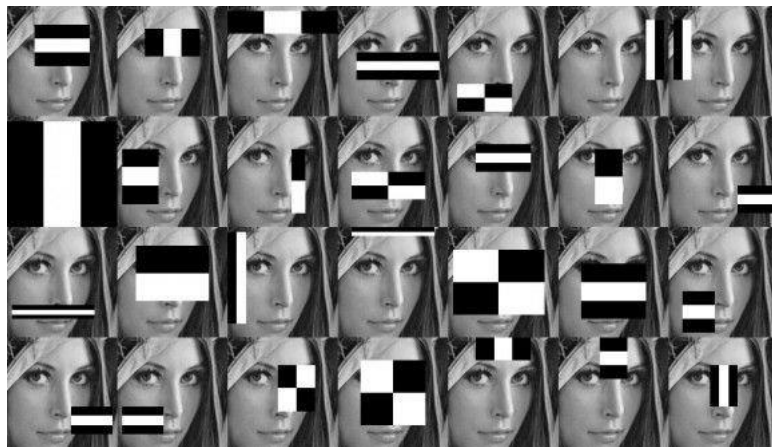


Рисунок 1.6 - Приклади ознак Хаара

б) Створення інтегральних образів

Попередній етап пропонує нам чудову систему для визначення класифікаторів обличчя. Проте для того, щоб такі обчислення не проводилися надто довго для цілого зображення, попіксельно вираховуючи значення, було запропоновано виділяти інтегральне зображення. Інтегральне зображення – це проміжне представлення зображення, де кожному пікселю вхідного зображення встановлюється значення суми верхніх лівих пікселів з вхідного зображення. Щоб детальніше зрозуміти як працює принцип обчислення значень інтегрального зображення розглянемо рисунок 1.7. Спочатку знаходимо значення кожного пікселя у інтегральному зображенні. Для цього використовуємо формулу 2.

$$ii(x, y) = \sum_{x' \leq x, y' \leq y} i(x', y'), \quad (2)$$

де $ii(x, y)$ – значення з інтегрального зображення,

$i(x, y)$ – значення з вхідного зображення.

Проте якщо використати формули 3 та 4, то можна отримати значення усіх пікселів інтегрального зображення за один прохід по матриці значень.

$$s(x, y) = s(x, y - 1) + i(x, y), \quad (3)$$

$$ii(x, y) = ii(x - 1, y) + s(x, y), \quad (4)$$

де $s(x, y)$ – кумулятивна сума рядка x ,

$$s(x, -1) = 0,$$

$$ii(-1, y) = 0.$$

Після того як ми знайшли значення інтегрального зображення, можемо отримати значення у підобласті-прямокутнику D . Сума пікселів у прямокутнику D обчислюється з чотирма посиланнями на матрицю значень з інтегрального зображення. Значення інтегрального зображення в місці 1 - це сума пікселів у прямокутнику A . Значення в місці 2 – сума значень інтегрального зображення A та B , в місці 3 – A та C , в місці 4 – A , B , C та D . Остаточна сума значень прямокутника D вираховується як різниця сум $(4 + 1) - (2 + 3)$ з інтегрального зображення. Таким чином, можна швидше отримати значення кожної підобласті. Враховуючи, що на одне зображення може бути застосовано приблизно 180,000 ознак Хаара, то вираховувати кожен раз суму в прямокутнику попідсильно займатиме досить великий час. Але з застосуванням інтегрального зображення ми зменшуємо складність обчислення до 4 звернень до матриці інтегрального зображення на одну ознаку [6].

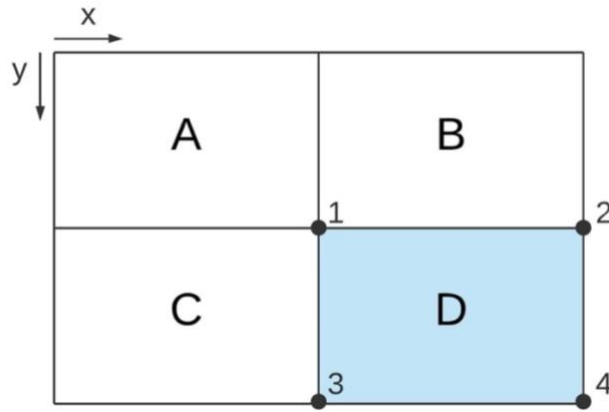


Рисунок 1.7 - Обчислення значення суми пікселів у області D

с) Використання AdaBoost

Важливо зауважити, що не всі ознаки Хаара будуть потрібні при виявленні об'єктів. Важливими є лише ті ознаки, які належать самому об'єкту. Тому для визначення найкращих ознак, що представляють обличчя та його риси, із сотень тисяч ознак Хаара використовується алгоритм AdaBoost. Він, по суті, вибирає найкращі ознаки та тренує класифікатори для їх використання. Алгоритм використовує комбінацію слабких класифікаторів, щоб створити один сильний класифікатор, який він зможе використовувати для виявлення обличчя та його рис.

Детальніше про роботу алгоритму розказано у другому розділі.

д) Впровадження каскадних класифікаторів

На цьому етапі класифікатори визначають чи був знайдений потрібний об'єкт у певній області зображення. Якщо об'єкту не було знайдено, то він просто переходить до наступного регіону для знаходження об'єктів. Детальніше у 2 розділі.

Підсумовуючи, у цьому підрозділі ми розглянули алгоритм пошуку характерних ознак Хаара та AdaBoost. Метою цих двох алгоритмів є визначення та знаходження обличчя та його окремих рис на вхідному зображенні.

1.5 Знаходження ключових точок та розпізнавання емоцій

Отже, за алгоритмом розпізнавання емоцій, представленому у підрозділі 1.1, наступним етапом є знаходження ключових точок рис обличчя. Принцип отримання цих точок полягає в наступному: перейти від кольорового зображення до напівтонового (grayscale), потім до бінарного, знайти границі рис обличчя і отримати ключові точки (рис. 4). Перехід від кольорового зображення до напівтонового виконується зазвичай на кроці захоплення і відстежування обличчя. Для знаходження границь об'єктів на зображенні можна застосувати алгоритм Канні, а після нього застосувати гістограму напрямлених градієнтів для виявлення ключових точок. Алгоритм поділений на 4 обов'язкові етапи: видалення шуму за функцією Гауса, виділення градієнту інтенсивності границь за методом Соболя, вилучення не максимумів з зображення, подвійна порогова фільтрація та трасування областей неоднозначності. Алгоритм видалення шуму за функцією Гауса було наведено у розділі 1.2, а всі інші етапи розписано детальніше далі.

Метод полягає у застосуванні двох масок до кожного пікселя зображення, які зображено на рисунку 1.8.

-1	-2	-1
0	0	0
1	2	1

Gx

-1	0	1
-2	0	2
-1	0	1

Gy

Рисунок 1.8 - Маски для методу Соболя

За допомогою цих масок можна вирахувати складові градієнта G_x та G_y , перемноживши значення пікселів на значення маски та просумувати усі перемножені значення. Далі за формулою 5 можна знайти значення кожного пікселя для виділення контурів зображень. На даному етапі отримуємо проміжне зображення як на рис 1.9. Також на цьому етапі проводиться вирахування напрямів градієнту за формулою 6.

$$f = \sqrt{G_x^2 + G_y^2}, \quad (5)$$

$$\theta = \arctg\left(\frac{G_y}{G_x}\right), \quad (6)$$

де G_x – горизонтальна складова градієнта,

G_y – вертикальна складова градієнта,

f – результуюче значення величини градієнту пікселя,

θ – напрям градієнту у градусах.

Після знаходження напрямів градієнту та значень пікселів за методом Соболя, ми вилучаємо пікселі (встановлюємо чорний колір), які мають значення менше, ніж значення двох пікселів-сусідів по напрямку градієнта. Для цього користуємося такими правилами:

- якщо кут градієнта пікселя дорівнює 0° , то порівнюємо пікселі у східному та західному напрямках;
- якщо округлений кут градієнта дорівнює 45° , то порівнюємо пікселі у напрямках північного заходу та південного сходу;
- якщо кут градієнта пікселя дорівнює 90° , то порівнюємо пікселі у північному та південному напрямках;
- якщо кут градієнта пікселя дорівнює 135° , то порівнюємо пікселі у напрямках південного заходу та північного сходу [7].

На наступному кроці потрібно зробити порогову фільтрацію зображення, зокрема виділити сильні та слабкі пікселі. За встановленими значення слабого та сильного порогу, ми порівнюємо значення кожного пікселя з цими порогами.

Якщо значення більше за сильний поріг, тоді ми привласнюємо максимальне значення цьому пікселю. Якщо значення більше за слабкий поріг, але менше за сильний, тоді ми залишаємо поточне значення даного пікселя. Якщо значення менше за слабкий поріг, то ми встановлюємо мінімальне значення для цього пікселя. Останнім кроком алгоритму Канні є трасування області неоднозначності. Для кожного пікселя, який не отримав максимального чи мінімального значення з попереднього кроку потрібно перевірити твердження чи є поряд з ним пікселі з максимальними значеннями. Якщо твердження вірне, тоді пікселю встановлюється максимальне значення. І навпаки, якщо твердження хибне, тоді пікселю встановлюється мінімальне значення. Таким чином, в результаті усіх кроків алгоритму Канні отримується зображення з чітко вираженими границями з границями рис обличчя. Результати кожного кроку зображені у додатку А.

Отже, останнім кроком після знаходження границь рис обличчя є застосування глибинного навчання для визначення емоцій людини на зображенні. Детальніше розглядатиметься у розділі 2.2.

Підсумовуючи, у розділі були розглянути основні алгоритми комп'ютерного зору, які застосовуються під час процесу розпізнавання емоцій на зображенні. Зокрема, було деталізовано кроки таких алгоритмів, як лінійне усереднення точок, розмиття за Гаусом, лінійне розтягування кольорової гістограми, розрахунок ознак Хаара, а також алгоритм Канні. Усі ці алгоритми необхідні для обробки вхідного зображення та вилучення основних рис обличчя. Ці риси обличчя передаватимуться певним класифікаторам, які і будуть визначати тип емоцій людини на зображенні на етапі машинного навчання.

2 АЛГОРИТМИ МАШИННОГО НАВЧАННЯ ДЛЯ РОЗПІЗНАВАННЯ ЕМОЦІЙ

2.1 Алгоритм AdaBoost

Як вже було зазначено в пункті 1.3, для знаходження рис обличчя нам треба обрати такі ознаки Хаара, які найкраще описують обличчя та його риси. Для цього застосовується алгоритм AdaBoost, мета якого – побудова сильних класифікаторів з набору слабких (рис. 2.1).

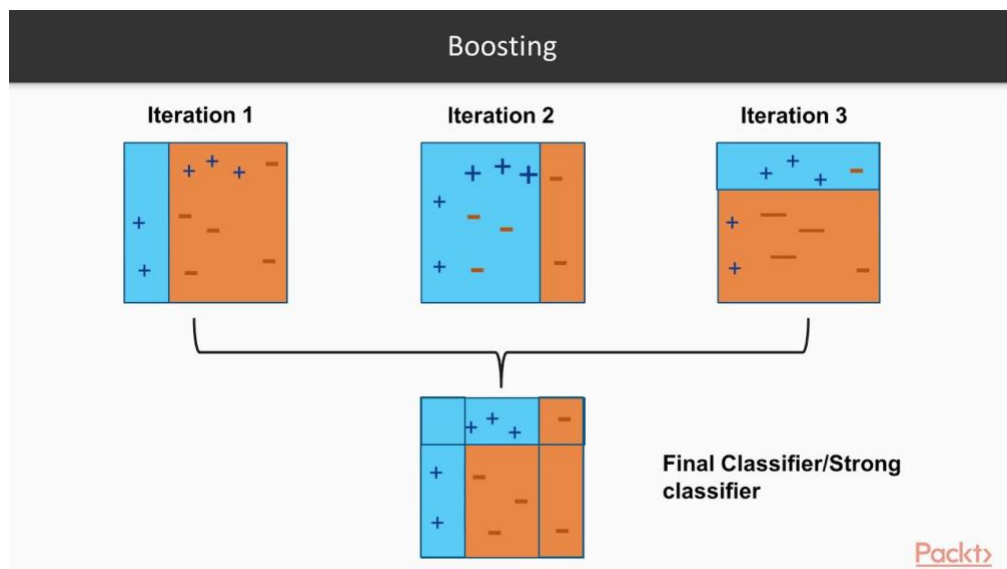


Рисунок 2.1 - Об'єднання слабких класифікаторів в сильний

Сильний класифікатор виражається формулою

$$F(x) = \alpha_0 f_0 + \alpha_1 f_1 + \dots + \alpha_n f_n, \quad (7)$$

де $F(x)$ – сильний класифікатор,

α_i – вага ознаки Хаара для сильного класифікатора,

f_i – певна ознака Хаара,

x – вхідне зображення.

У наведеній формулі, α_n представляє вагу відповідної ознаки об'єкта. Вага - це спосіб вказати, наскільки ознака важлива для класифікації даного об'єкту. Якщо вага α_i певної ознаки дорівнює нулю, то це означає, що f_i не має значення при класифікації цільового зображення. Прикладом нульової ваги може бути фон в алгоритмі виявлення обличчя. Більшість ознак, які виявляються у фоновій частині зображення, матимуть нульову вагу, оскільки фон змінюватиметься від картинки до картинки [8].

Мета Adaboost - оновити ваги конкретних ознак, щоб максимізувати ймовірність правильної класифікації шуканого об'єкта, одночасно усуваючи подібні ознаки Хаара, які не допомагають у процесі виявлення. Ознаки, які надають класифікатору більшу кількість істинних-позитивних (шуканий об'єкт, класифікований як шуканий) та істинно-негативних значень (шуканий об'єкт, класифікований як зайвий), отримують більшу вагу, оскільки вони, швидше за все, будуть унікальними для шуканого об'єкта. З іншого боку, ознаки, що призводять до великої кількості помилково-позитивних (зайвий об'єкт, класифікований як шуканий) і помилково-негативних значень (зайвий об'єкт, класифікований як зайвий) отримують менші ваги з можливістю присвоєння вагам значення нуль [9].

Після визначення сильних класифікаторів, настає стадія застосування техніки каскадних класифікаторів. Суть цієї техніки полягає у тому, що для виявлення певної риси обличчя використовується окремі набори класифікаторів, які застосовуються поетапно. Зважаючи на це, на початкових етапах перевіряються простіші ознаки Хаара та застосовуються відповідні класифікатори, а на більш пізніх етапах, які включають у себе досить складні ознаки, на області зображення знаходяться більш дрібні деталі обличчя або його рис. Якщо початковий етап не виявляє нічого у певній області, алгоритм пропускає цю область та переходить до наступної області. Таким чином буде заощаджено багато часу, оскільки зайві області не оброблятимуться на більшості етапів [10].

Обробка другого етапу розпочнеться лише тоді, коли елементи на першому етапі будуть виявлені на зображенні. Процес розпізнавання триває аж поки не будуть пройдені усі етапи, після чого об'єкт класифікується як обличчя.

Підсумовуючи, у цьому підрозділі було розглянуто алгоритм AdaBoost, який застосовується для підсилення класифікації рис обличчя. Цей алгоритм є складовою розпізнавання об'єктів за методом Віоли-Джонса з застосуванням ознак Хаара та надає змогу розпізнати ці об'єкти з мінімальною похибкою. Саме тому він є одним із найпоширеніших методів розпізнавання обличчя.

2.2 Гістограма напрямлених градієнтів

Останніми етапами для виявлення емоцій на зображенні є етапи виявлення ключових точок рис обличчя та вибір емоцій. Для цього нам знадобиться певний дескриптор, який зможе виявляти точки рис обличчя та визначати емоцію.

Для виявлення ключових точок, скористаємося гістограмою напрямлених градієнтів. У пункті 1.4 ми знайшли величини та кути градієнта кожного пікселя. За допомогою цих значень ми можемо вирахувати та побудувати гістограми, які показуватимуть розподіл значень величини та кутів градієнта. Для цього ми візьмемо значення цих показників з кожного пікселя та розподілимо їх у 9 підкатегорій, значення кутів яких змінюються від 0 до 160 з кроком у 20. Перевіряємо значення кута градієнта пікселя та знаходимо два найближчі значення з 9 підкатегорій. Знаходимо різницю кута з кутами двох підкатегорій, ділимо цю різницю на 20, множимо на значення величини градієнта та сумуємо з тим значення, що має підкатегорія на даний момент (див. рис. 2.2). В результаті отримуються вектори суми значень величини градієнта.

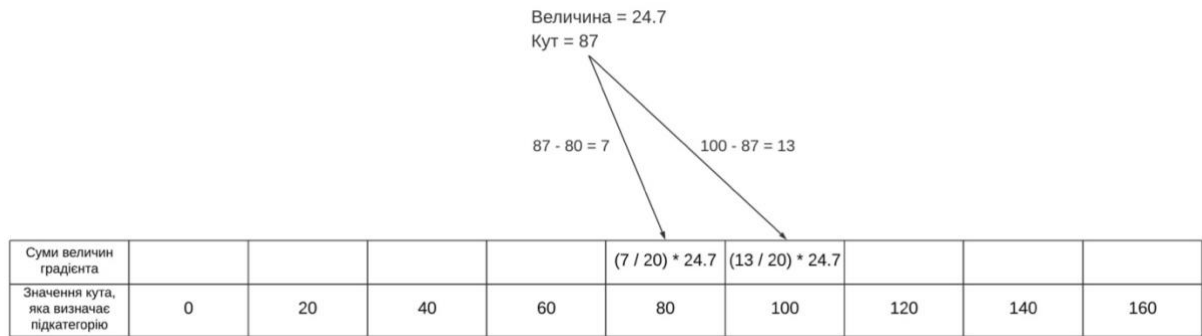


Рисунок 2.2 - Метод розподілу значень градієнта на гістограмі

Такі вектори шукаються для областей 8x8 пікселів, які потім комбінуються у області 16x16 та для них вираховуються відповідні вектори з застосуванням нормалізації. Таким чином, ми утворюємо вектори значень градієнта з 36 підкатегоріями, які і є дескрипторами. Візуалізація даних дескрипторів показана в Додатку Б.

2.3 Support Vector Machine

Після того, як ми визначили дескриптори використовуючи гістограму напрямлених градієнтів, можна перейти до етапу розпізнавання емоцій. На цьому етапі, нам знадобиться метод опорних векторів (SVM - Support Vector Machine), який застосовуватиме визначені дескриптори на минулому кроці для класифікації емоцій за рисами обличчя.

Метод опорних векторів (SVM) - це керований алгоритм машинного навчання, який може використовуватися як для класифікації, так і для регресії, однак він здебільшого використовується в задачах класифікації. В алгоритмі SVM ми будемо кожен елемент даних як точку в n-мірному просторі, значення кожного об'єкта є значенням певної координати. Потім знаходиться гіперплощина, яка краще всього диференціює класи об'єктів (рис. 2.3). Як тільки така площина була знайдена, проводиться класифікація об'єктів [11].

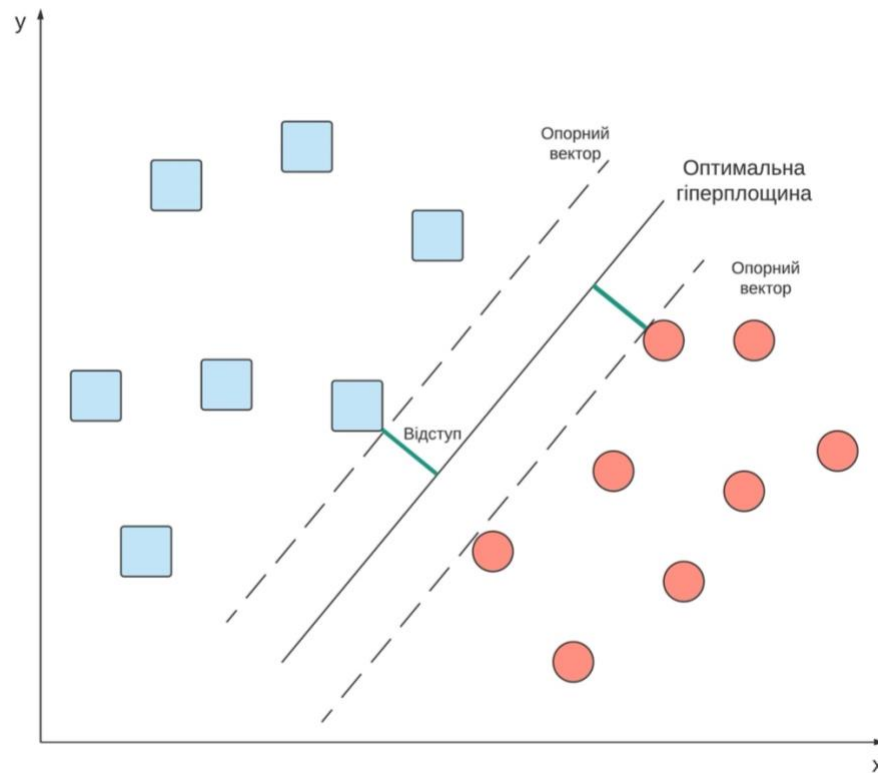


Рисунок 2.3 - Визначення гіперплощини

Гіперплощина - це $n-1$ мірна підплощина в n -вимірному просторі, яка розділяє простір на окремі частини. Для того, щоб знайти цю гіперплощину, шукаються точки на графіку, які розташовані до лінії поділу найближче. Ці точки називаються опорними векторами. Потім, алгоритм обчислює відступ між опорними векторами і розділяє площиною. Основна мета алгоритму - максимізувати значення відступу. Кращою гіперплощиною вважається така гіперплощина, для якої цей відступ є найбільшим [12]. Таким чином, алгоритм, навчаючись на навчальній вибірці, будує таку гіперплощину, яка б надалі допомагала класифікувати раніше невідомий об'єкт за заданими ознаками. У нашому випадку цими ознаками виступають вектори градієнтів зображення, які визначатимуть тип емоції за певними рисами обличчя. Коли гіперплощина побудована, алгоритм розподіляє вхідні ознаки для класифікації, і відповідно до того, по яку сторону лежать точки з нашими ознаками, визначається тип емоції.

Підсумовуючи, у цьому розділі ми детальніше роздивилися застосування алгоритмів машинного навчання, які беруть участь у розпізнаванні емоцій.

Зокрема, були розглянуті SVM та AdaBoost алгоритми, які часто використовуються разом з методом ознак Хаара. AdaBoost дозволяє вибрати найкращі слабкі класифікатори та об'єднати їх у сильні для визначення обличчя на зображенні, а SVM разом з обчисленням гістограм напрямлених градієнтів застосовується для класифікації емоцій за рисами обличчя.

3 РОЗРОБКА ПРОГРАМНОГО ПРОТОТИПУ З ВИКОРИСТАННЯМ ФУНКЦІОНАЛУ РОЗПІЗНАВАННЯ ЕМОЦІЙ

3.1 Основні засоби розробки

Для серверної частини, завдання якої отримати емоції користувача з відеопотоку, були використані наступні засоби розробки:

- Node.js – це кросплатформене середовище виконання на базі JavaScript із відкритим вихідним кодом. За допомогою Node.js створюється серверна частина застосунку.
- Axios – це HTTP-клієнт для Node.js та браузера на основі Promise-ів. Використовується для надсилання HTTP запитів та отримання даних з зовнішніх джерел.
- Formidable – модуль Node.js для парсингу даних форми. У застосунку використовувався для спрощення отримання відео-файлів з отриманих даних форми.
- FFmpeg – бібліотека з відкритим вихідним кодом, які дозволяють записувати, конвертувати і передавати цифрові аудіо- та відеозаписи в різних форматах.
- FFProbe – у поєднанні з FFmpeg використовується для обробки відео-файлів. Вона також дозволяє збирати і відображати інформацію про медіафайли.
- Fs – використовується для операцій запису та зчитування файлів з файлової системи.
- Dotenv – модуль, який завантажує змінні середовища з файлу .env у process.env. Використовується здебільшого для розгортання застосунку на локальному та хмарному середовищах та зберігання токенів для доступу до зовнішніх API.
- Docker - це програмна платформа, яка дозволяє швидко створювати, тестувати та розгортати програми. Docker пакує програмне забезпечення в

стандартизовані блоки, які називаються контейнерами, що містять усе необхідне для запуску програмного забезпечення, включаючи бібліотеки, системні інструменти, код та середовище виконання. Для розгортання BackEnd-у був використаний образ Node Alpine, а для FrontEnd-у – Node та Nginx.

- Azure Cognitive Services Face – використаний API для розпізнавання емоцій, який надається у доступ від Microsoft Azure.

На клієнтській частині для отримання відеопотоку з камери користувача використовувалися MediaStream Recording API.

3.2 Реалізація прототипу

Для початку розглянемо детальну структуру обміну даними застосунку на рисунку 3.1.

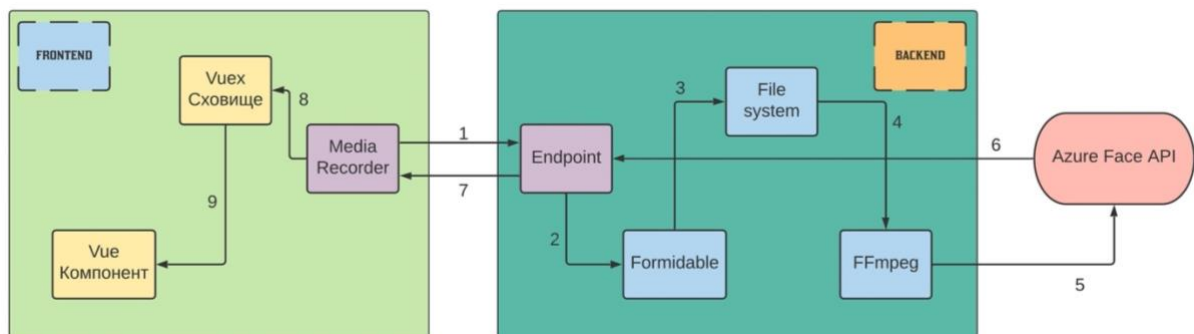


Рисунок 3.1 – Потік даних застосунку

Для того, щоб зреагувати на зміну емоції користувача, спочатку потрібно отримати доступ до камери користувача. Цю функцію на себе бере MediaRecorder, який перевіряє чи може бути наданий доступ до відеокамери. Якщо доступ було надано, ініціалізується вхідний потік та починається записуватися відео кілька секунд. У програмі відразу задається відповідний EventListener, який спрацює, коли вхідний потік зупиниться. У цьому EventListener-і дані, записані з відеокамери, будуть конвертуватися у бінарний

тип даних, тобто Blob. Цей Blob потім додається до даних форми (FormData) та за допомогою Axios надсилається на серверну частину. На серверній частині Endpoint приймає запит та автоматично оброблює його через модуль Formidable для зручного доступу до файлів у FormData. Коли Formidable завершить обробку файлу, FFMpeg зчитає цей файл з тимчасового сховища. Далі файл зберігається в файлову систему за допомогою FFMpeg метода output та зчитується знову за допомогою метода input. Це спричинено тим, що модуль FFMpeg не може відразу отримати позначки часу відео, отриманого через модуль Formidable. Після того, як відеофайл знову зчитався з файлової системи, отримуються скріншоти з відео за допомогою метода takeScreenshots, які зберігаються у файлову систему. Після цього викликається метод, який зчитуватиме скріншоти з файлової системи та надсилатиме їх у тілі запиту до Azure Face API. Для того щоб розпізнати емоції людини використовується Detect Endpoint, який окрім самих емоцій, надає дані про кількість виявлених облич та їх характеристики, наприклад, область та усі ключові точки обличчя, стать, вік, чи носить людина окуляри, нахил голови на фото, якого кольору волосся тощо. Таким чином, отримавши цей об'єкт з Face API, ми передаємо його назад до клієнтської частини у блок з MediaRecorder. На серверній частині залишається тільки видалити усі проміжні відео та фото, використані для визначення емоції користувача на відео. MediaRecorder, отримавши необхідні нам значення, викликає так званий Action Vuex, який записує дані про обличчя у сховище даних застосунку. З цього сховища Vue компоненти можуть отримувати поточну емоцію користувача та проводити зміни з інтерфейсом та елементами сторінок.

В результаті виконання цих всіх операцій застосунок має можливість реагувати на зміну емоцій користувача та надавати йому найбільш відповідний контент. Основними емоціями, які розпізнаються є:

- Злість (Anger);
- Зневага (Contempt);
- Огида (Disgust);

- Страх (Fear);
- Радість (Happiness);
- Нейтральність (Neutral);
- Сум (Sadness);
- Здивування (Surprise).

У додатку В наводяться приклади розпізнавання кожної емоції з відображенням прямокутника обличчя, ключових точок та назви емоції.

Відповідно до емоцій користувача потрібно створити різні відображення елементів для покращення користувацького досвіду. У застосунку при змінах емоції змінюються банери та кольори кнопок, а також змінюються написи у модальному вікні відгуків (додаток Г). Відтак, коли користувач посміхається, користувач бачить вікно у куту сторінки із усміхненим обличчям і йому пропонується написати відгук про те, що саме йому сподобалося. Якщо користувач сумує, йому пропонується написати відгук про те, що саме викликало сум.

ВИСНОВКИ

У цій роботі було розглянуто кожен етап алгоритму розпізнавання емоції за допомогою комп'ютерного зору та машинного навчання, зокрема були описані основні ідеї алгоритму пошуку обличчя та його рис за допомогою ознак Хаара і алгоритмів AdaBoost для покращення вибірки класифікаторів та Support Vector Machine для класифікації емоцій користувача.

Також було розроблено програмний застосунок з функціоналом для розпізнавання емоцій користувачів та подальшої зміни контенту веб-сторінки відповідно до розпізнаних емоцій. Такий напрямок розробки допомагає досягти кращої взаємодії з користувачем та є дуже перспективним в галузі програмних продуктів. Він поєднує складні алгоритми комп'ютерного зору та машинного навчання з різноманітними напрацюваннями із боку дизайну користувацького інтерфейсу та користувацького досвіду (UI/UX). Покращення комфорту користування додатку та виклик позитивних емоцій грають важливу роль для бізнес-моделі застосунку.

Напрацювання в даній роботі можна застосувати у багатьох напрямках. Окрім сфери e-commerce, до якої належать різні інтернет-магазини, такі як Rozetka, Amazon, Hotline тощо, розпізнавання емоцій можливе для додатків у будь-якій сфері, від сфери охорони здоров'я на прикладі додатку для психологічної підтримки до сфери розваг. Застосування способів проведення змін у дизайні продукту від відчуттів користувача надає більше можливостей для отримання зворотного зв'язку та покращення продукту. Відповідно, інвестуючи в такі напрацювання, можна збільшити привабливість застосунку, відтак збільшуючи кількість користувачів та загальний дохід від даного продукту.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Why Google has 200m reasons to put engineers over designers [Електронний ресурс]. Режим доступу: <https://www.theguardian.com/technology/2014/feb/05/why-google-engineers-designers>.
2. Ольга Кірдяєва. Тривімірний простір веб-дизайну: UI/UX / Ольга Кірдяєва.
3. Ян Сі. Автоматичні розпізнавання емоцій користувача для організації інтелектуального інтерфейсу / Ян Сі. - Електронний журнал «Молодіжний науково - технічний вісник», 2013. - № 9.
4. Попередня обробка зображень [Електронний ресурс]. Режим доступу: <https://api-2d3d-cad.com/processingimages/>.
5. Samir Khanal. Contrast Stretching [Електронний ресурс]. Режим доступу: <https://samirkhanal35.medium.com/contrast-stretching-f25e7c4e8e33>.
6. Paul Viola. Rapid object detection using a boosted cascade of simple features / Michael Jones, Paul Viola. - Accepted Conference on Computer Vision and Pattern Recognition, 2001.
7. Canny Edge Detector [Електронний ресурс]. Режим доступу: https://en.wikipedia.org/wiki/Canny_edge_detector.
8. Ben Mauss. AdaBoost in Image Classification [Електронний ресурс]. Режим доступу: <https://levelup.gitconnected.com/adaboost-in-image-classification-8dcc1799e53d>.
9. Aishwarya Singh. Feature Engineering for Images: A Valuable Introduction to the HOG Feature Descriptor [Електронний ресурс]. Режим доступу: <https://www.analyticsvidhya.com/blog/2019/09/feature-engineering-images-introduction-hog-feature-descriptor/>.
10. Girija Shankar Behera. Face Detection with Haar Cascade [Електронний ресурс]. Режим доступу: <https://towardsdatascience.com/face-detection-with-haar-cascade-727f68dafd08>.

11. Sunil Ray. Understanding Support Vector Machine (SVM) algorithm from examples (along with code) [Електронний ресурс]. Режим доступу: <https://www.analyticsvidhya.com/blog/2017/09/understaing-support-vector-machine-example-code/>.
12. Короткий огляд алгоритму машинного навчання Метод Опорних Векторів (SVM) [Електронний ресурс]. Режим доступу: <https://habr.com/ru/post/428503/>.

ДОДАТКИ

Додаток А (обов'язковий)

Покрокове застосування алгоритму Канні

1. Оригінальне зображення



2. Розмиття Гауса



3. Метод Соболя



4. Вилучення не максимумів



5. Порогова фільтрація



6. Результуюче зображення



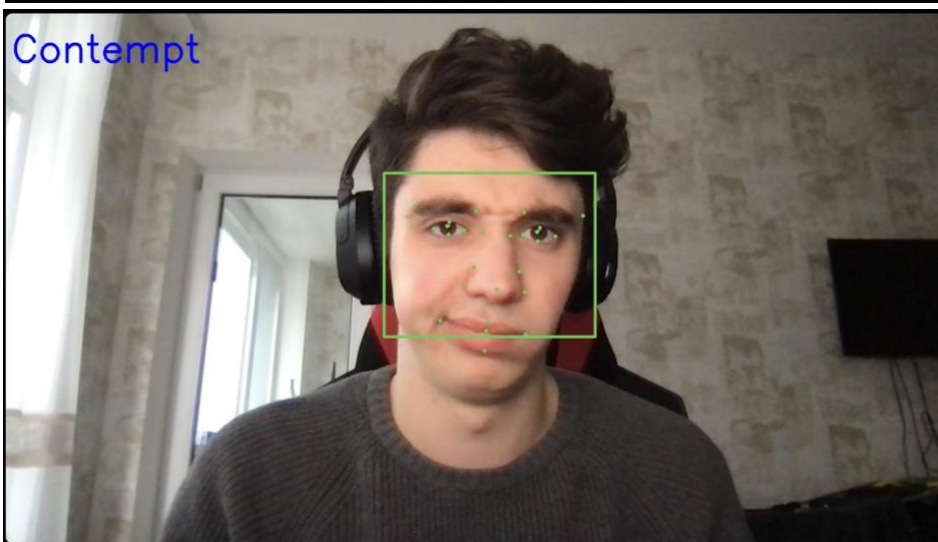
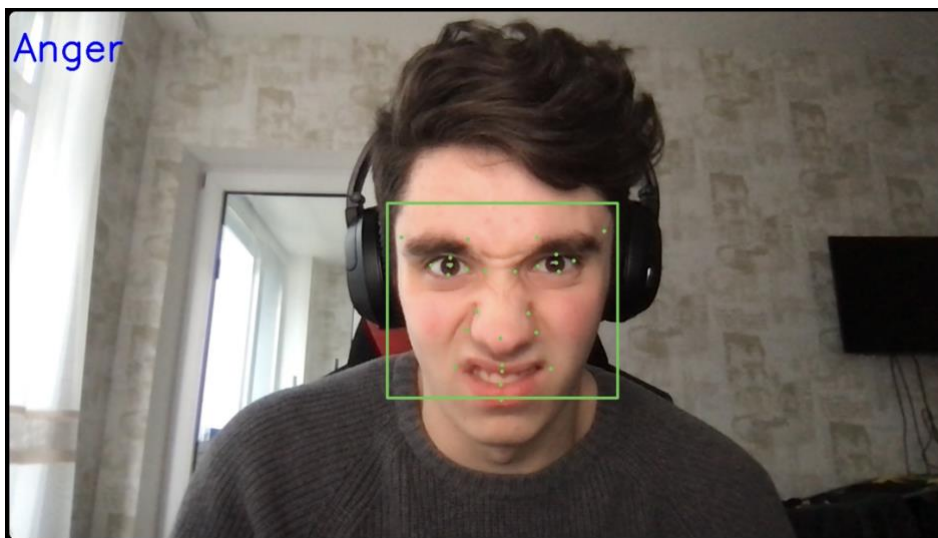
Додаток Б
(обов'язковий)

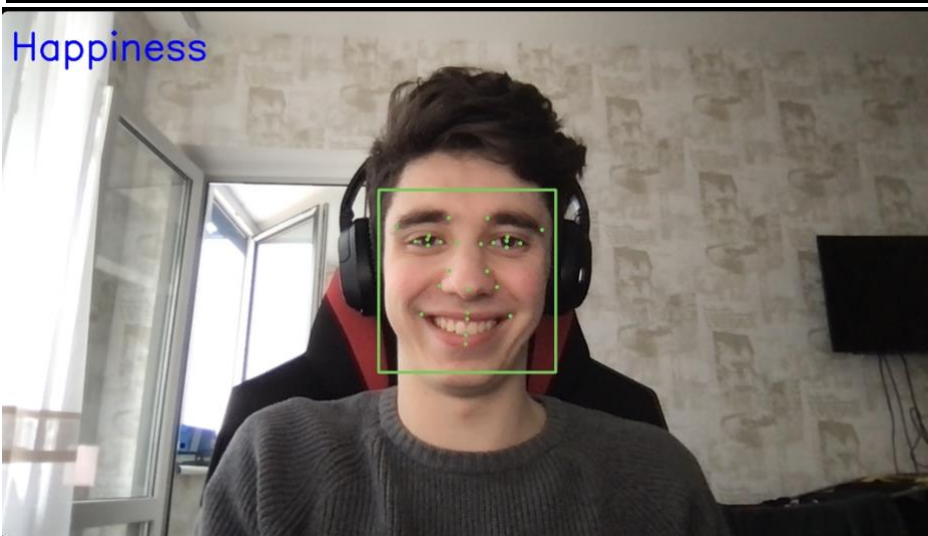
Візуалізація гістограми напрямлених градієнтів

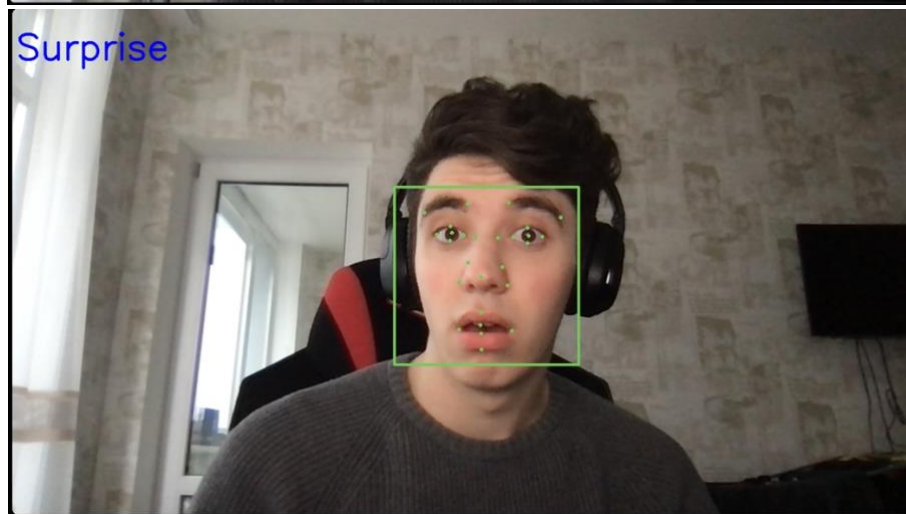
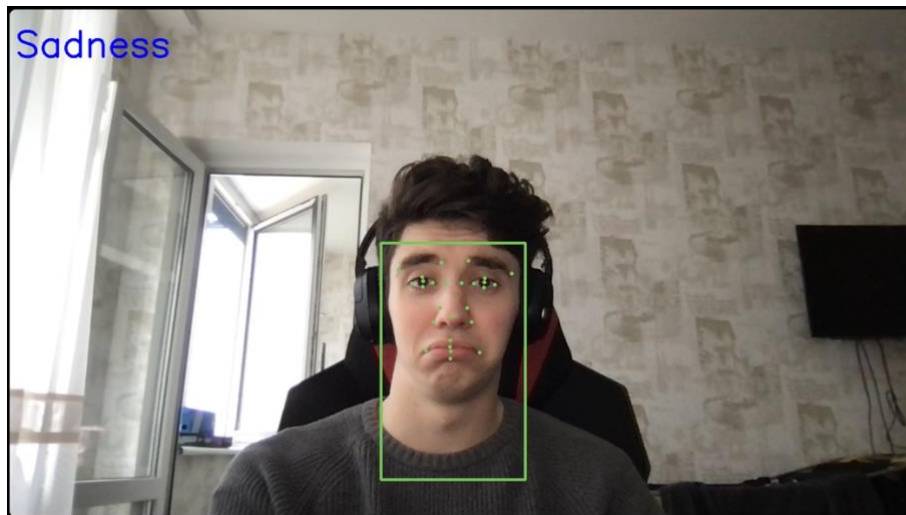


Додаток В
(обов'язковий)

Виділення прямокутника обличчя, ключових точок та емоцій

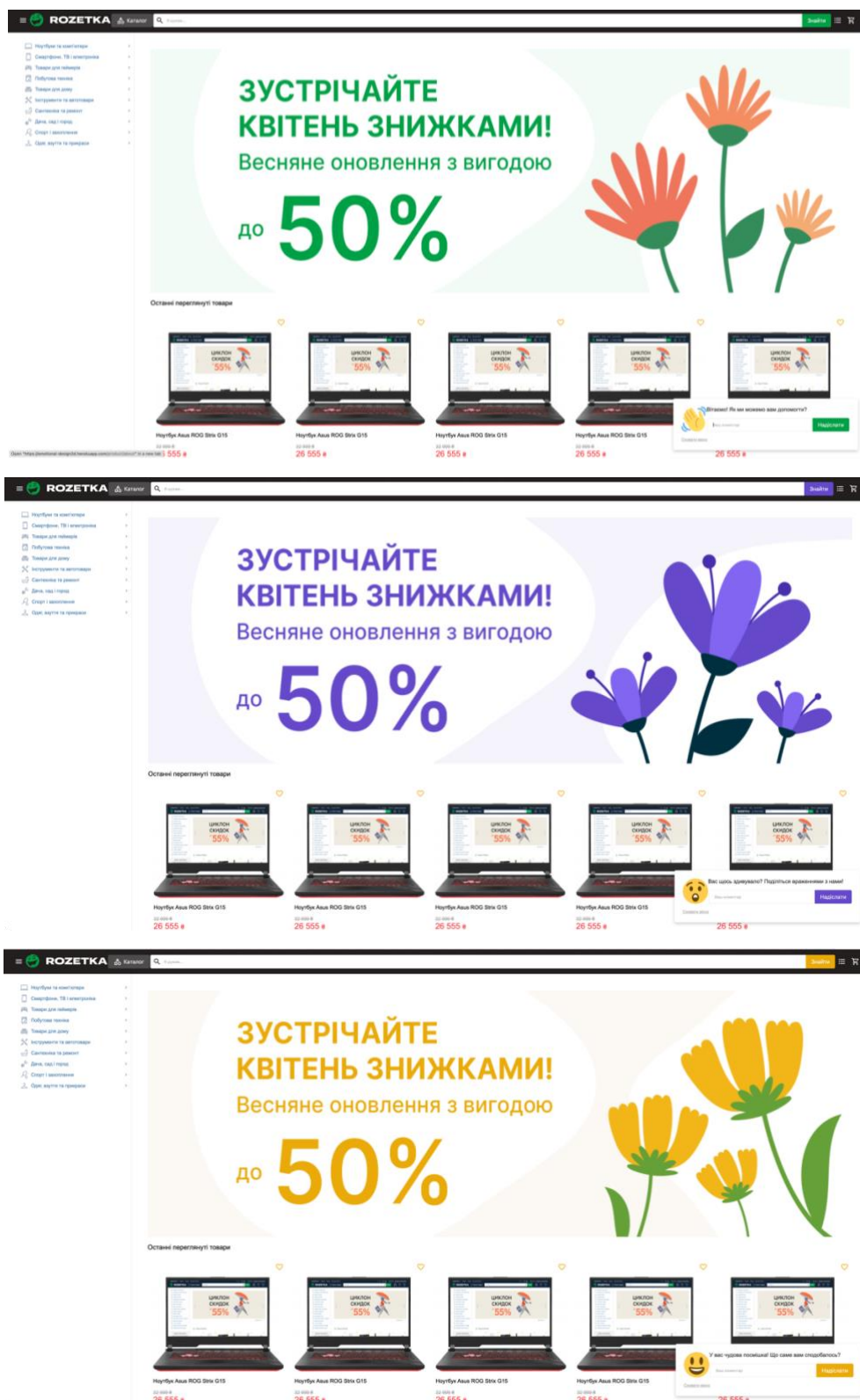






Додаток Г (необов'язковий)

Скріншоти застосунку при розпізнаванні різних емоцій (нейтральність, здивування, радість, сум, страх, огида, злість та зневага)



ROZETKA [Каталог](#)

- Ноутбуки та комп'ютери
- Смартфони, ТВ та інтернет
- Товари для побуту
- Побутова техніка
- Товари для дому
- Інструменти та автозапчасти
- Садівництво та ремонт
- Діти, іграшки
- Спорт і активність
- Спеціалізовані товари

ЗУСТРІЧАЙТЕ КВІТЕНЬ ЗНИЖКАМИ!

Весняне оновлення з вигодою до **50%**



Останні переглянуті товари

				
22 000 ₴ 26 555 ₴	22 000 ₴ 26 555 ₴	22 000 ₴ 26 555 ₴	22 000 ₴ 26 555 ₴	22 000 ₴ 26 555 ₴

Що ви хочете у вас таку реакцію?

ROZETKA [Каталог](#)

- Ноутбуки та комп'ютери
- Смартфони, ТВ та інтернет
- Товари для побуту
- Побутова техніка
- Товари для дому
- Інструменти та автозапчасти
- Садівництво та ремонт
- Діти, іграшки
- Спорт і активність
- Спеціалізовані товари

ЗУСТРІЧАЙТЕ КВІТЕНЬ ЗНИЖКАМИ!

Весняне оновлення з вигодою до **50%**



Останні переглянуті товари

				
22 000 ₴ 26 555 ₴	22 000 ₴ 26 555 ₴	22 000 ₴ 26 555 ₴	22 000 ₴ 26 555 ₴	22 000 ₴ 26 555 ₴

Ваша реакція? Оцініть наш виступ професійно!

ROZETKA [Каталог](#)

- Ноутбуки та комп'ютери
- Смартфони, ТВ та інтернет
- Товари для побуту
- Побутова техніка
- Товари для дому
- Інструменти та автозапчасти
- Садівництво та ремонт
- Діти, іграшки
- Спорт і активність
- Спеціалізовані товари

ЗУСТРІЧАЙТЕ КВІТЕНЬ ЗНИЖКАМИ!

Весняне оновлення з вигодою до **50%**



Останні переглянуті товари

				
22 000 ₴ 26 555 ₴	22 000 ₴ 26 555 ₴	22 000 ₴ 26 555 ₴	22 000 ₴ 26 555 ₴	22 000 ₴ 26 555 ₴

Ваша реакція? Оцініть наш виступ професійно!

ROZETKA [Каталог](#)

- Найти по категориям
- Смартфоны, ТВ, ноутбуки
- Товары для дома
- Популярные товары
- Товары для детей
- Аксессуары к автомобилям
- Косметика и уход
- Дача, сад и огород
- Спорт и развлечения
- Дом, уют и традиции

ЗУСТРІЧАЙТЕ КВІТЕНЬ ЗНИЖКАМИ!

Весняне оновлення з вигодою

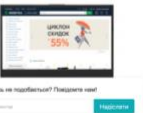
до 50%



Останні переглянуті товари







Навуфк Asus ROG Strix G15 26 555 ₴

Навуфк Asus ROG Strix G15 26 555 ₴

Навуфк Asus ROG Strix G15 26 555 ₴

Навуфк Asus ROG Strix G15 26 555 ₴

Навуфк Asus ROG Strix G15 26 555 ₴

Ваш смак не підпадає? Пошукайте інші!

Ваш смак не підпадає? Пошукайте інші!

Ваш смак не підпадає? Пошукайте інші!

Ваш смак не підпадає? Пошукайте інші!

Ваш смак не підпадає? Пошукайте інші!

ROZETKA [Каталог](#)

- Найти по категориям
- Смартфоны, ТВ, ноутбуки
- Товары для дома
- Популярные товары
- Товары для детей
- Аксессуары к автомобилям
- Косметика и уход
- Дача, сад и огород
- Спорт и развлечения
- Дом, уют и традиции

ЗУСТРІЧАЙТЕ КВІТЕНЬ ЗНИЖКАМИ!

Весняне оновлення з вигодою

до 50%



Останні переглянуті товари







Навуфк Asus ROG Strix G15 26 555 ₴

Навуфк Asus ROG Strix G15 26 555 ₴

Навуфк Asus ROG Strix G15 26 555 ₴

Навуфк Asus ROG Strix G15 26 555 ₴

Навуфк Asus ROG Strix G15 26 555 ₴

Ваш смак не підпадає? Пошукайте інші!

Ваш смак не підпадає? Пошукайте інші!

Ваш смак не підпадає? Пошукайте інші!

Ваш смак не підпадає? Пошукайте інші!

Ваш смак не підпадає? Пошукайте інші!