КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА
Факультет комп'ютерних наук та кібернетики
Кафедра інтелектуальних програмних систем

Кваліфікаційна робота
на здобуття ступеня бакалавра
за спеціальністю 121 Програмна інженерія
на тему:

СТВОРЕННЯ МИСТЕЦТВОЗНАВЧОГО ІНСТРУМЕНТУ ДЛЯ ШВИДКОГО ОТРИМАННЯ ІНФОРМАЦІЇ З ЗОБРАЖЕНЬ ПРОДУКТІВ ХУДОЖНЬОЇ ТВОРЧОСТІ У ФОРМІ БОТА В TELEGRAM

Виконав студент 4-го курсу
Даніїл ОСОКА _____ (підпис)

Науковий керівник:
доцент, кандидат фіз.-мат. наук
Лариса КАТЕРИНИЧ _____ (підпис)

Засвідчую, що в цій роботі немає запозичень з праць інших авторів без відповідних посилань.
Студент _____(підпис)

Роботу розглянуто й допущено до захисту
на засіданні кафедри інтелектуальних програмних систем
«25» травня 2022 р.,
протокол No 10
Завідувач кафедри
Олександр ПРОВОТАР _____(підпис)

Київ – 2022

РЕФЕРАТ

Обсяг роботи 32 сторінки, 8 ілюстрацій, 11 джерел посилань.

БОТ В TELEGRAM, МЕДІАННИЙ ФІЛЬТР, МИСТЕЦТВОЗНАВЧИЙ ІНСТРУМЕНТ, НАДАННЯ ІНФОРМАЦІЇ ПРО ОБ'ЄКТИ МИСТЕЦТВА, ПОПЕРДНЯ ОБРОБКА ЗОБРАЖЕННЯ, РОЗПІЗНАВАННЯ ОБ'ЄКТІВ МИСТЕЦТВА, РОЗПІЗНАВАННЯ ОБ'ЄКТІВ НА ЗОБРАЖЕННІ, СМАРТФОН, GOOGLE CLOUD VISION, HEROKU, OPEN CV, PY TELEGRAM BOT API, PYTHON, WIKI ART.

Об'єктом розробки є спрощення процесу отримання інформації про об'єкти культури шляхом створення мистецтвознавчого інструменту, що дозволить швидко дізнатися деталі про мистецькі витвори в незалежності від їх місцезнаходження в світі.

Метою роботи є створення інструменту для туристів, що дозволяє спростити процес отримання інформації про об'єкти культури.

Методи розроблення: попередня обробка зображення, розпізнавання за допомогою попередньо тренованих нейронних мереж, робота зі спеціалізованими інтернет-каталогами з об'єктами мистецтв, розробка зручного користувацького інтерфейсу. Інструменти розроблення: мова розробки Python версії 3.8, інтегроване середовище розробки PyCharm Profesional 2022.2 EAP, сервіс Google Cloud Vision, бот в Telegram, система контролю версій Git зі зберіганням коду на віддаленому сервері GitHub, хостинг Heroku.

Результати роботи: Спроектовано та розроблено програмний продукт, що отримав простий інтерфейс і бажаний функціонал, система допомагає користувачу отримати інформацію про мистецький об'єкт за фотознімком

цього об’єкту. Система показала свою працездатність, а також хороший рівень гнучкості в плані ракурсів та якості зображень мистецьких об’єктів.

Можливі сфери застосування: “кишеньковий гід” для кожного туриста, що бажає пізнавати культурну складову територій які він відвідує; зручний застосуноґк для шкільних екскурсій на різні мистецькі виставки та в картинні галереї, що спрощує процес освіти.

Є доцільним продовжити просування в розробці даного програмного продукту. Основним, що може бути покращено, є попередня обробка зображення, наприклад додання можливості вирівнювати перспективу може підвищити гнучкість застосунку в питанні розпізнавання об’єктів мистецтва в кадрі з відхиленням по осям обертання відносно самого витвору мистецтва.

ЗМІСТ

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАЧЕННЯ

QR-код — матричний штрих-код. Основна перевага QR-коду — це легке розпізнавання сканувальним обладнанням

IDE — Integrated Development Environment, інтегроване середовище розробки

VCS — Version Control System, система контролю версій

API — Application Programming Interface, інтерфейс прикладного програмування

PaaS — Platform as a service, платформа як послуга

URL — Uniform Resource Locator, єдиний вказівник на ресурс

ВСТУП

Оцінка сучасного стану об'єкта розробки. Актуальність роботи та підстави для її виконання. Впродовж сторічь у художніх галереях та різноманітних виставках мистецьких об'єктів використовувалися доволі прості, хоча діючі, способи донесення інформації про презентований об'єкт мистецтва. Серед таких методів були друковані тексти під картинами та архітектурними пам'ятками, а також екскурсоводи. Такі підходи й досі широко використовуються по всьому світу, багаторічний досвід показав дієвість таких методів.

Проте, в сучасному світі з'являється все більше аналогів, що допомагають сильно спростити і оптимізувати дану роботу. Серед таких можна виділити: аудіо-гіди, QR-коди з посиланнями на інтернет-ресурси, системи кодувань об'єктів мистецтва для швидкого пошуку в інтернет каталогах тощо. І це дійсно є великою оптимізацією, адже нам не треба витрачати ресурси на друк табличок та заробітну плату екскурсоводам, але такі підходи не є уніфікованими по всьому світу, що поскладнює життя кінцевому туристу. Отже, туристам потрібен такий інструмент, що став би у нагоді кожному туристу, що бажає вивчити культурну складову території, що він відвідує. Головне в даному інструменті має бути те, що він буде уніфікованим для будь-якої мистецької виставки або галереї, а також різноманітних вуличних художніх об'єктів, що зазвичай не мають ніяких підписів, чи інформації про них, навіть якщо вони є дуже відомими. Такий інструмент також має мати інтуїтивно зрозумілий і простий інтерфейс.

Мета і завдання роботи. За мету роботи взято потребу створити такий інструмент для туристів, що дозволяє спростити процес отримання інформації про об'єкти культури. Даний інструмент повинен мати максимально простий та інтуїтивно зрозумілий інтерфейс. Також він має безвідмовно працювати в

незалежності від того в якій галереї знаходиться об’єкт мистецтва. Ще важливо, щоб такий інструмент був доступний на якомога більшій кількості платформ. Якщо коротко, то даний інструмент має бути доступних кожному, хто бажає ним скористатися.

Для досягнення поставленої мети необхідно виконати наступні завдання:

- виявити найзручніший спосіб взаємодії з системою для туриста;
- проаналізувати наявні засоби для розробки програмної частини, та обрати найзручніші серед аналогів;
- спроектувати та розробити програмну частину;
- протестувати отриману систему.

Об’єкт, методи й засоби розробки. Об’єктом розробки є спрощення процесу отримання інформації про об’єкти культури шляхом створення мистецтвознавчого інструменту, що дозволить швидко дізнатися деталі про мистецькі витвори в незалежності від їх місцезнаходження в світі.

Для розробки було обрано наступні засоби та інструменти:

- В якості мови розробки було обрано Python версії 3.8;
- В якості IDE було обрано PyCharm Profesional 2022.2 EAP;
- Для розпізнавання зображень було використано Google Cloud Vision;
- В якості інтерфейсу користувача було створено бота в Telegram за допомогою бібліотеки-обгортки над API для керування ботами в даному месенджері;
- В якості VCS використано Git, для зберігання коду на віддаленому сервері обрано GitHub

Можливі сфери застосування. Створений інструмент має стати у нагоді кожному туристу, що бажає вивчати культурну складову місцевості,

що він відвідує. Також дана система може стати зручним застосунком для шкільних екскурсій на різні мистецькі виставки та в картинні галереї, таким чином спростити процес освіти.

РОЗДІЛ 1. ПЕРЕДУМОВИ СТВОРЕННЯ ІНСТРУМЕНТА

1.1 Розширений огляд теми та її актуальності

Впродовж довгого часу для ознайомлення з об'єктами мистецтва у людства було два дієвих підходи: екскурсовод та друкований або карбований опис біля об'єкту мистецтва. Такі методи досі сильно розповсюджені по всьому світу, проте така тенденція починає потроху змінюватись. В умовах стрімкого розвитку технологій багато провідних мистецьких галерей та культурних виставок змінюють ці методи на більш сучасні.

І це дійсно має сенс. Для друку або карбування описів потрібні матеріали, що не є надто екологічно, в сучасних реаліях це доволі важливе питання. Він того, щоб виробляти таблички з описом для об'єктів мистецтва відмовляється на користь створення відносно малих зображень з QR-кодами з посиланнями на інтернет-ресурси або кодами для спеціалізованих каталогів, які можна відкрити за допомогою смартфону. З екскурсоводами ситуація наступна: це просто невигідно в порівнянні з існуючими аудіо-гідами в довгостроковій перспективі, адже людям треба платити заробітну плату постійно; також треба мати в штаті людей які знають різні мови аби задовольнити якомога більшу кількість туристів. Саме тому багато стаціонарних мистецьких галерей активно закупають аудіо-гіди і записують на них інформацію про виставлені витвори мистецтва. Також в скарбничку аудіо-гіда потрапляє перевага в якості можливості переслухати звукову доріжку, тож якщо відвідувач пропустив якусь цікаву інформацію — він зможе її переслухати.

Такі підходи донесення інформації про об'єкти мистецтва вже є сильним покращенням відносно тих, які використовували протягом останніх сотень років, проте вони все ще мають певні недоліки. Серед їх слабких сторін можна виокремити наступні:

Кожна галерея або виставка має свої певні особливості у підходах до донесення такої інформації до відвідувача, того туристу доволі часто приходиться знайомитись з трохи іншим способом видобування потрібної інформації, це викликає небажану затрату часу на звикання до нового способу взаємодії, а звідси і погіршення настрою;

У випадку якщо об'єкт мистецтва знаходиться поза межами виставкових павільйонів і розташований просто на вулиці, як, наприклад, картини зображені на Берлінській Стіні, то варіанти з аудіо-гідами і зовсім відпадають, лишається варіант використанням QR-кодів, що впроваджено далеко не у всіх вуличних об'єктів культури.

Таким чином, виникає потреба створення програмного продукту, що був би позбавлений зазначених вище недоліків і став би у нагоді кожному туристу, що бажає ознайомитися с культурною спадщиною відвіданої місцевості.

1.2 Бачення рішення проблеми

В сучасному світі тяжко уявити людину, що не користується смартфоном. Згідно зі статистикою, наразі в світі 83.72 відсотки від усіх людей користуються смартфонами. Якщо брати відсоток від усіх людей, що носять мобільний телефон (таких наразі 91.54 відсотки від усіх людей світу), то не важко підрахувати, що серед користувачів стільникового зв'язку 91,47 відсотка має смартфон.[8] Така статистика наштовхує на думку, що рішення має бути втілене першочергово для смартфонів, адже це те, що турист має під рукою майже завжди.

Використовуючи дане програмне рішення, користувач повинен мати можливість отримати інформацію про мистецький об'єкт без надання додаткової інформації, окрім тієї, яку від може отримати, дивлячись на художній витвір, тобто його зовнішній вигляд. Таким чином, запропоноване

програмне рішення має за зображенням культурного об'єкту видавати необхідну інформацію про нього.

1.3 Аналіз існуючих аналогів

Серед існуючих рішень можна виділити готовий продукт компанії Google — Google Lense. Він дозволяє аналізувати надані йому зображення, виділяти на них об'єкти, тощо. Це більш універсальний інструмент для аналізу зображень ніж є метою створити.

Слід перерахувати недоліки такого рішення:

- не спеціалізований для роботи з об'єктами мистецтва;
- доступний лише для користувачів операційної системи Android, для користувачів IOS він є недоступним, тож об задовольнити якомога більшу частину користувачів, необхідно створити кросплатформене рішення;
- Неможливість збереження історії запитів: в даному застосунку можна лише отримати інформацію за поточний запит. Історію минулих запитів він не зберігає.

1.4 Обрання інструментів та технологій розробки

Оскільки інтерфейс користувача має бути доволі простим, то виникла думка про те, щоб використати якісь існуючі кросплатформені інструменти, з якими більшість користувачів будуть знайомі, замість того, щоб створювати застосунок для смартфона повністю з нуля, а також займатися забезпеченням його роботи на різних платформах. На роль такого інструменту прекрасно підходять боти в месенджерах, а саме вони вміють завантажувати зображення, або ж робити його за допомогою вбудованої в смартфон камери, задавати прості команди, а за потреби ще й створювати кнопки і обробляти їх

натискання. Більш того, використання месенджера є для сучасних користувачів доволі буденною справою, багато людей використовують свій смартфон здебільшого для переглядання та переписувань в месенджерах, тут навіть звичайні дзвінки відходять на другий план[9]. Навіть перегляд звичайних новин стає більш звичним через месенджери. Таким чином мінімальні апаратні вимоги до нашого застосунку стають еквівалентними до таких в обраному нами месенджері.

Серед месенджерів було обрано Telegram, як мессенджер, чия аудиторія наразі зростає найшвидше[10]. Також таке рішення дозволить розповсюдити програмний продукт на велику кількістю платформ, серед яких не тільки мобільні, адже повноцінні клієнти Telegram доступні і для десктопних платформ, таких як Linux, MacOS, Windows.

Мовою розробки стала мова Python через її гнучкість і наявність великої кількості бібліотек для реалізації потрібного функціоналу без написання зайвого коду. Логічним вибором IDE став PyCharm Profesional 2022.2 EAP, як один з найкращих серед аналогів для даної мови програмування. Використання професійної версії стало можливим завдяки тому, що компанія JetBrains надає безкоштовний доступ усім студентам вищих навчальних закладів.

Для взаємодії з Telegram було використано одну з найрозповсюдженіших бібліотек-обгорток над API для ботів Telegram, а саме pyTelegramBotApi.

РОЗДІЛ 2. ТЕОРЕТИЧНА ЧАСТИНА

2.1 Попередня обробка зображень

2.1.1 Загальні відомості

В роботі з машинним навчанням попередня обробка вхідних даних або очищення останніх від спотворень є важливим кроком, і більшість інженерів, що працюють в цій області, витрачають значну кількість часу на попередню обробку даних перед створенням самої моделі. Деякі приклади попередньої обробки даних включають виявлення аномалій, обробку пропущених значень і видалення небажаних або шумних даних.

Аналогічно, попередня обробка зображень — це термін для операцій над зображеннями на найнижчому рівні абстракції. Ці операції не збільшують інформаційний вміст зображення.

Метою попередньої обробки є покращення якості зображення, щоб ми могли краще проаналізувати його. За допомогою попередньої обробки ми можемо придушити небажані спотворення та покращити деякі функції, що мають значення для подальшої обробки та аналізу в потрібному контексті в рамках нашого застосунку.

2.1.2 Визначення необхідних підходів в рамках теми

Перед визначенням необхідних підходів до попередньої обробки, нам необхідно розглянути те, як користувач буде застосовувати застосунок, і в якому вигляді зображення будуть поступати до нашої системи.

Слід передбачати, що користувач не завжди зможе зробити зображення відмінної якості, адже фотографії робитимуться на смартфон. Сучасні смартфони хоч і вже дуже сильно розвинулися в питанні якості фотографій відносно перших мобільних телефонів з наявними модулями камер, проте вони все ще мають дуже малі за розмірами матриці у порівнянні з

дзеркальними фотоапаратами. І навіть якщо в умовах хорошого освітлення камери сучасних мобільних флагманів здатні видати подібний до професійних камер результат, то в умовах неідеального освітлення різниця стає знатно помітнішою. Нижче наведено порівняння якості фотографії в нічних умовах між деякими сучасними флагманами і професійною камерою (Рисунок 1). Порівняння було проведено авторитетним виданням DXOMark[11].

Рисунок 1 – Порівняння якості фотографій в умовах поганої освітленості

Також не слід забувати, що далеко не у всіх туристів наявні в користуванні актуальні флагмани преміальних виробників смартфонів. Тож слід розраховувати, що зазвичай якість фотографій буде ще гірша.

Найбільшою проблемою при створенні фотознімків в умовах недостатнього освітлення є наявність великої кількості шумів на результуючому зображенні. Саме з ними і було вирішено боротися на етапі попередньої обробки зображення.

2.1.3 Видалення шумів на зображенні

Для видалення шумів з зображення в ході попередньої обробки вхідних даних було обрано медіанний фільтр.

Медіанний фільтр – це один з варіантів нелінійного рангового фільтра. Він являє собою вікно квадратної форми, що переміщується по зображенню і

охоплює непарне число елементів (пікселів). Вікно проходить по кожній точці зображення, при чому так, що під час обробки чергового пікселя, останній знаходиться в самому центрі вікна фільтра. Центральна точка цього вікна замінюється медіаною елементів, що попали у квадратне вікно фільтру (Рисунок 2). Медіаною дискретної послідовності з N елементів при непарному N називається елемент, для якого існує $(N-1)/2$ елементів, менших або рівних йому по величині, і $(N-1)/2$ елементів, більших або рівних йому за значенням.

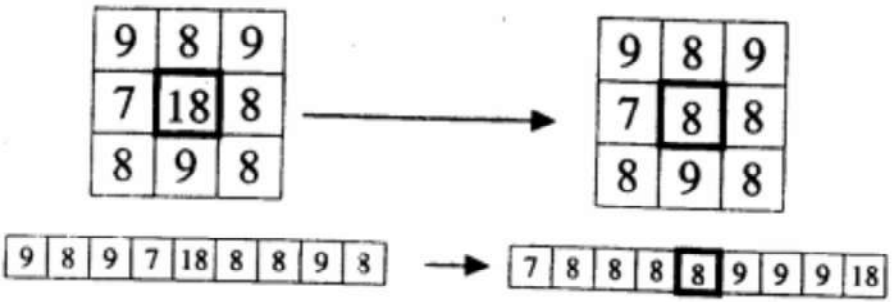

Рисунок 2 – Приклад роботи медіанного фільтру з вікном 3x3

Медіанний фільтр не пропускає імпульсні сигнали, розміри яких складають менше половини ширини вікна фільтра. Варто зазначити, що медіанний фільтр слід вважати евристичним методом обробки зображень.

2.2 Розпізнавання за допомогою Google Cloud Vision

Google Cloud Vision — це хмарне рішення для візуального пошуку, розроблене, щоб допомогти розробникам керувати процесами, пов'язаними з розпізнаванням зображень, за допомогою спеціальних моделей машинного навчання. Завдяки попередньо тренованим моделям машинного навчання та створеному API інженери можуть отримувати результат розпізнавання

рукописного тексту, список об'єктів на зображенні, а також можуть призначати власні мітки для класифікації зображень.

Обрання такого рішення дозволить отримати високу точність розпізнавання, адже моделі тренуються на величезній кількості інформації, що потрапляє в мережу інтернет щоденно.

2.3 Отримання інформації про мистецький об'єкт з WikiArt

WikiArt — це найбільш відома інтернет-енциклопедія, що спеціалізується на об'єктах мистецтва. Ресурс постійно поповнюється та редагується користувачами зі всього світу, що в поєднанні з його популярністю дає змогу стверджувати, що він є одним з найбільш наповнених каталогів з витворами мистецтва.

Також даний ресурс надає власне API для отримання доступу до інформації, що наявна в базах даних цієї інтернет-енциклопедії. Це значно спрощує отримання інформації в порівнянні з парсингом веб-сторінки, окрім того, ми точно знаємо, що ми беремо інформацію, на отримання якої розробники ресурсу надали дозвіл.

2.4 Хостинг на Heroku

Heroku — хмарна PaaS, що підтримує ряд мов програмування. Heroku, одна з перших хмарних платформ, з'явилась в червні 2007 року і спочатку підтримувала тільки мову програмування Ruby, але зараз список підтримуваних мов також містить в собі Java, Node.js, Scala, Golang, Clojure, Python і PHP. На серверах Heroku використовуються операційні системи Debian або Ubuntu.

Heroku було обрано як сервіс, що з часом зарекомендував себе як один з найпопулярніших і простих у застосуванні серед конкурентів, має велику

кількість користувачів, що в свою чергу може значно знизити час вирішення проблем з налаштуванням та роботою сервісу.

РОЗДІЛ 3. ПРАКТИЧНА ЧАСТИНА

3.1 Проектування програмної частини

3.1.1 Набір програмних засобів розробки

При створенні програмного продукту, що відповідає поставленій задачі в рамках даної кваліфікаційної роботи, було використано наступний список програмних засобів розробки:

- Python 3.8 [1]
- pyTelegramBotApi [2]
- OpenCV [4]
- Google Cloud Vision API [3]
- WikiArt API [6]
- PyCharm Profesional 2022.2 EAP
- Heroku [7]

3.1.2 Огляд основних принципів роботи програмного продукту

Даний застосунок надає користувачу простий інтерфейс у вигляді діалогового вікна бота в Telegram. Бот надає можливість ідентифікувати мистецькі витвори з великою точністю, якщо такі витвори раніше були внесені в спеціалізовані електронні репозиторії, доступ до яких є можливість здійснити через мережу інтернет.

Загалом взаємодію користувача з програмним продуктом можна розділити на дві частини: запит на розпізнавання зображення та обрання найбільш схожого витвору серед результатів пошуку.

Роботу першої частини можна розглянути поетапно завдяки діаграмі послідовності (Рисунок 3).

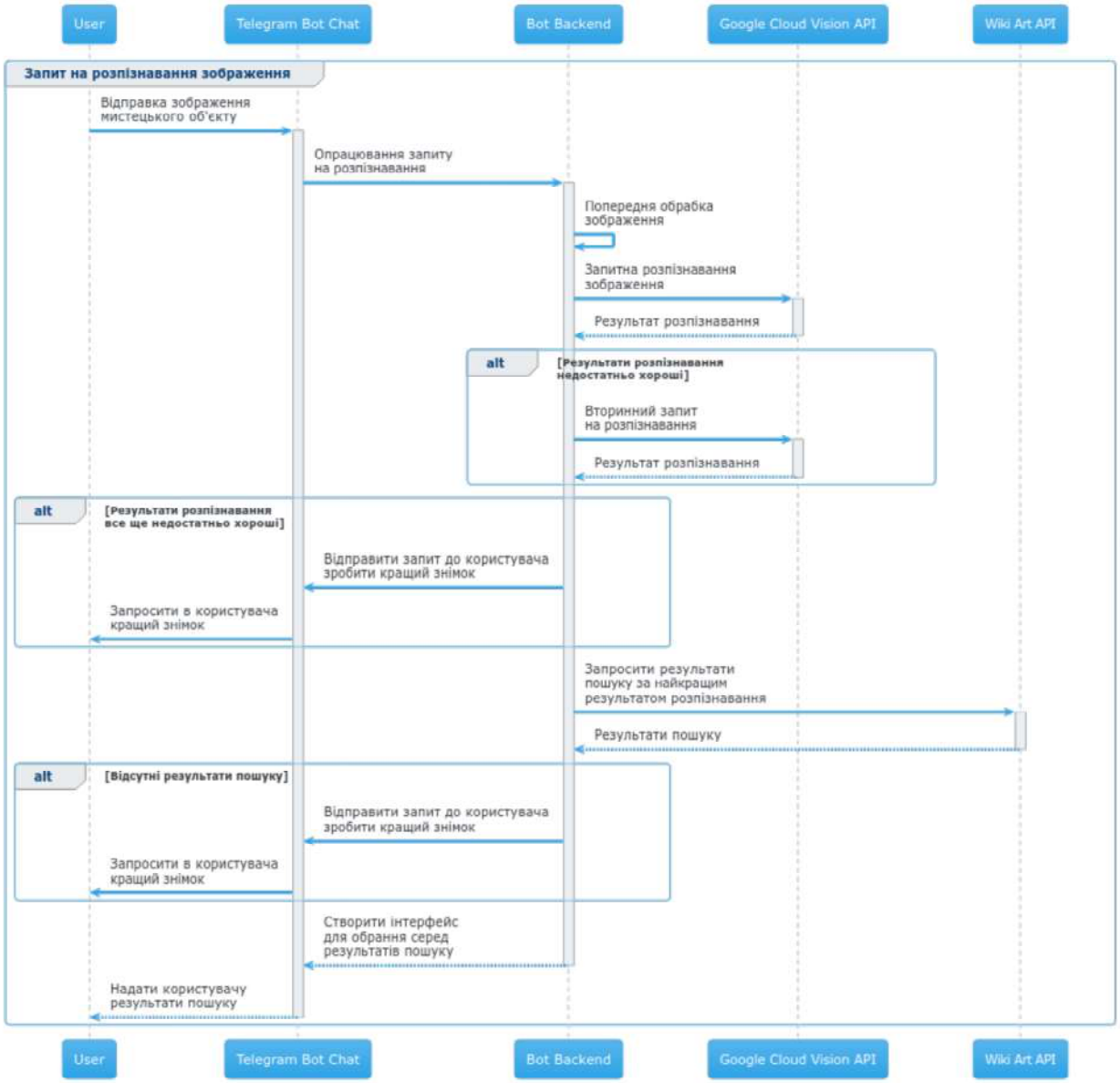

Рисунок 3 — Діаграма послідовності запиту на розпізнавання

Даний запит розбивається на наступні етапи:

– Відправка зображення користувачем. На цьому етапі турист надає системі зображення мистецького об’єкту, інформацію про який він бажає отримати. Користувач може надіслати зображення з будь-якого пристрою, де наявний Telegram. Також зображення може бути будь-якого походження: взяте з галереї та збережених

файлів або зроблене за допомогою камери через застосунок месенджера;

– Попередня обробка зображення. Перед відправкою вхідних даних на розпізнавання, вони проходять етап зниження кількості шумів на зображенні. Зображення проходить медіанну процес медіанної фільтрації (Рисунок 4);

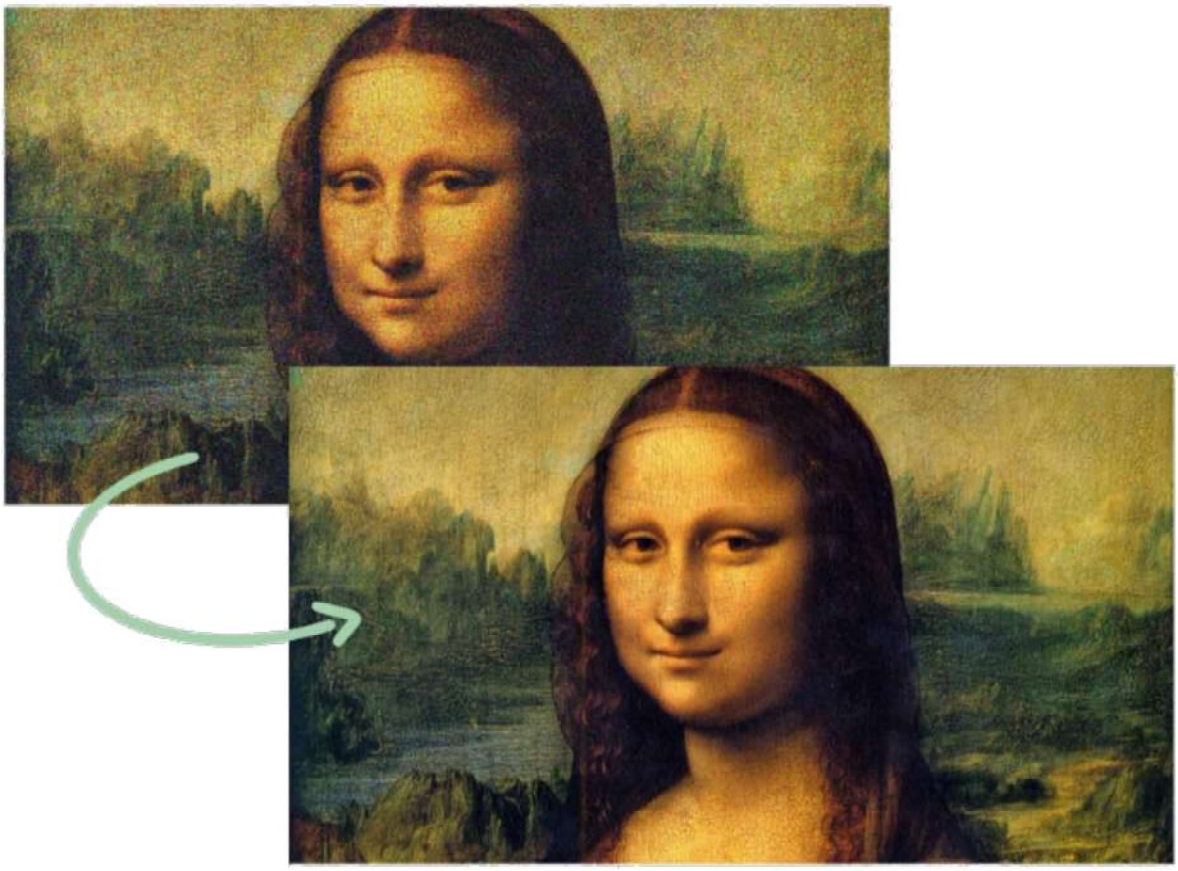

Рисунок 4 — Приклад попередньої обробки зображення

– Запит на розпізнавання до Google Cloud Vision API. Далі відбувається відправка оброблених даних на сервіс для отримання результатів розпізнавання, а також програма очікує на відповідь з результатами;
– Аналіз отриманих результатів. Якщо отримані результати після першої відправки на Google Cloud Vision API не є задовільної

якості (занадто низька впевненість нейронної мережі у результатах розпізнавання), то відбувається повторний запит на сервіс. Для того, щоб це не був абсолютно ідентичний до попереднього запит, змінюються його параметри, а саме активуємо функцію аналізу географічного положення, де було зроблено знімок. Ця інформація береться з мета даних зображення. Якщо після другого запиту результати розпізнавання все ще є незадовільними, то користувач отримує повідомлення, яке просить надіслати зображення мистецького об'єкту, зробленого в кращій якості (менше сторонніх об'єктів, більш сфокусовано саме на витворі культури). Після надання нового фотознімку, дана послудовність почнеться спочатку;

- Запит на отримання результатів пошуку до WikiArt API. На даному етапі відбувається запит до спеціалізованого мистецького каталогу на основі найкращих результатів розпізнавання. Послідовно відбуваються запити до WikiArt у порядку від найкращого задовільного результату розпізнавання до найгіршого серед таких. Запити відбуваються доти, поки не буде отримано непорожні результати пошуку в спеціалізованому каталозі. Якщо не вдається знайти жодного результату пошуку, то користувач отримує повідомлення про необхідність зробити краще зображення об'єкту мистецької творчості. Після повторної відправки, дана послідовність етапів повторюється спочатку.
- Відправка результатів користувачу. Якщо всі згадані вище етапи пройшли успішно, то користувач отримує набір зображень з можливістю обрання потрібного йому.

Після останнього етапу частини з отриманням результатів розпізнавання від системи, користувач має можливість обрати серед наведених зображень витворів мистецтва те, що найбільше походить на той об'єкт мистецтва, що

він бачить перед собою. Частину з отриманням інформації по обранному зображенню теж продемонстровано за допомогою діаграми послідовності (Рисунок 5).

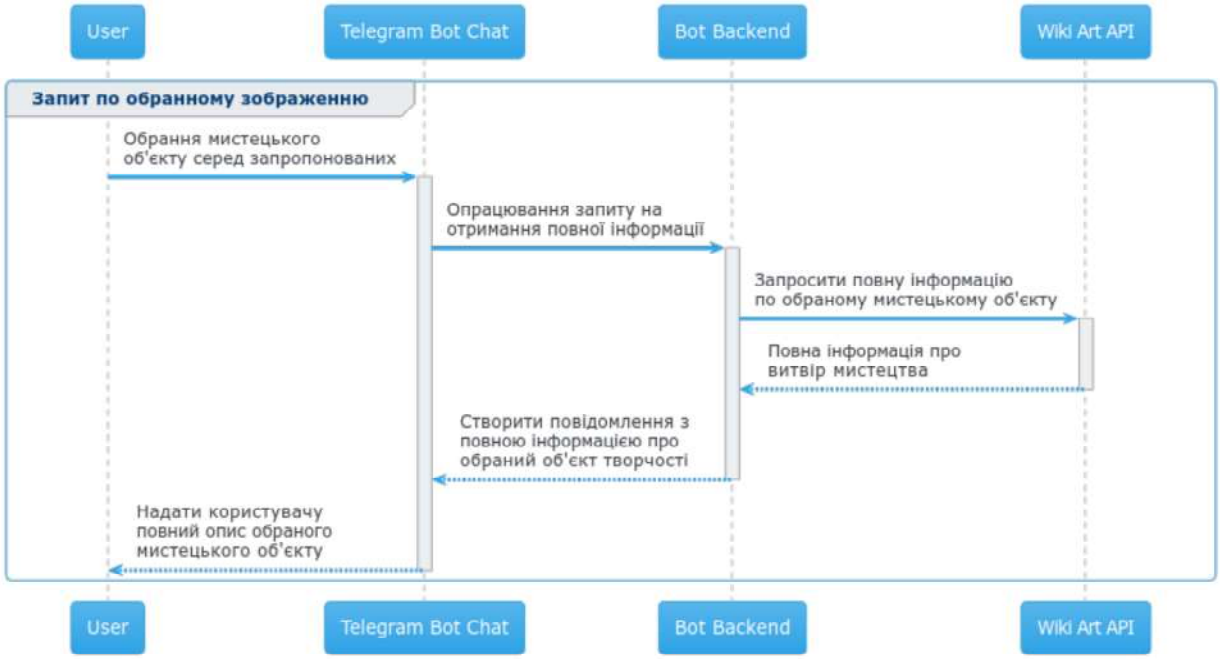

Рисунок 5 — Діаграма послідовності отримання повної інформації

В цій частині користувач обирає бажаний об’єкт творчості через інтерфейс в чаті бота Telegram. Така подія призводить до того, що відбувається запит на сервіс WikiArt на отримання повної інформації про обраний об’єкт художньої творчості за допомогою API. У відповідь приходить повна інформація про мистецький витвір. Далі відбувається обробка отриманої відповіді і оформлення потрібної інформації в зручному вигляді, після чого, результат відправляється користувачу в чат бота Telegram.

3.2 Створення коду

3.2.1 Загальна структура коду

При написанні коду було розділено програмний продукт на наступні модулі:

- processors.image_preprocessor.py
- processors.recognition.py
- processors.description_processor.py
- bot.py

Далі розглянуто наведені модулі більш детально

3.2.2 Модуль processors.image_preprocessor.py

Модуль, що в якості інтерфейсу надає одну функцію filter_noise(image). Параметром функції є вхідне зображення у вигляді масиву байтів, поверненим значенням є такий самий тип. Функція отримує зображення, проводить його попередню обробку, а саме проганяє через медіанний фільтр, та повертає оброблене зображення

3.2.3 Модуль processors.recognition.py

Модуль надає як інтерфейс дві функції:

- get_info(photo). Вхідним параметром є зображення представлене у вигляді масиву байтів. Вихідними даними функції є список результатів розпізнавання, поєднаних з рівнем впевненості нейромережі у кожному з ним, результати відсортовані від кращого до гіршого за цим рівнем. Функція робить запит до Google Cloud Vision API на розпізнавання заданого зображення.
- get_info_with_geo(photo). Вхідні і вихідні дані функції ідентичні до тих що наявні у функції, зазначеної вище. Відмінність у роботі функцій міститься у тому, що ця відправляє запит до сервісу з іншими параметрами, а саме активує функцію врахування геопозиції зображення. Географічне положення береться з метаданих зображення.

3.2.4 Модуль processors.description_processor.py

Інтерфейс модуля складається з двох функцій:

- get_search_results(art_name). Здійснює пошук мистецьких об’єктів в каталозі WikiArt за назвою. Вхідним параметром є назва мистецького об’єкту. Вихідними даними буде список пар елементів: ідентифікаційний номер об’єкту та URL на його зображення.
- get_full_info_by_id(art_id). Здійснює отримання повної інформації про об’єкт за його ідентифікатором в базі WikiArt. В якості вхідних даних маємо ідентифікатор. Функція повертає рядок з форматованою для відправки користувачу інформацією про витвір мистецтва.

3.2.5 Модуль bot.py

Модуль містить три хендлери для роботи з Telegram:

- start(). Реагує на початок роботи з ботом користувачем. Надсилає привітання і базові інструкції.
- process_photo(). Обробляє зображення, що надходить через чат з ботом та керує відправкою результатів розпізнавання чи повідомлень про погану якість фотознімку. Використовує наступні функції з модулів описаних вище:
  - processors.image_preprocessor.filter_noise(image);
  - processors.recognition.get_info(photo);
  - processors.recognition.get_info_with_geo(photo);
  - processors.description_processor.get_search_results(art_name)
- callback_inline(). Обробляє на обрання певного мистецького об’єкту серед результатів пошуку, керує відправкою повної інформації про об’єкт користувачу. Використовує функцію get_full_info_by_id(art_id) з модуля processors.description_processor.

3.3 Інтерфейс користувача

Користувацький інтерфейс являє собою чат з ботом в Telegram.

При початку роботи з ботом користувач отримує повідомлення з вітанням і простим описам, що робити (Рисунок 6).

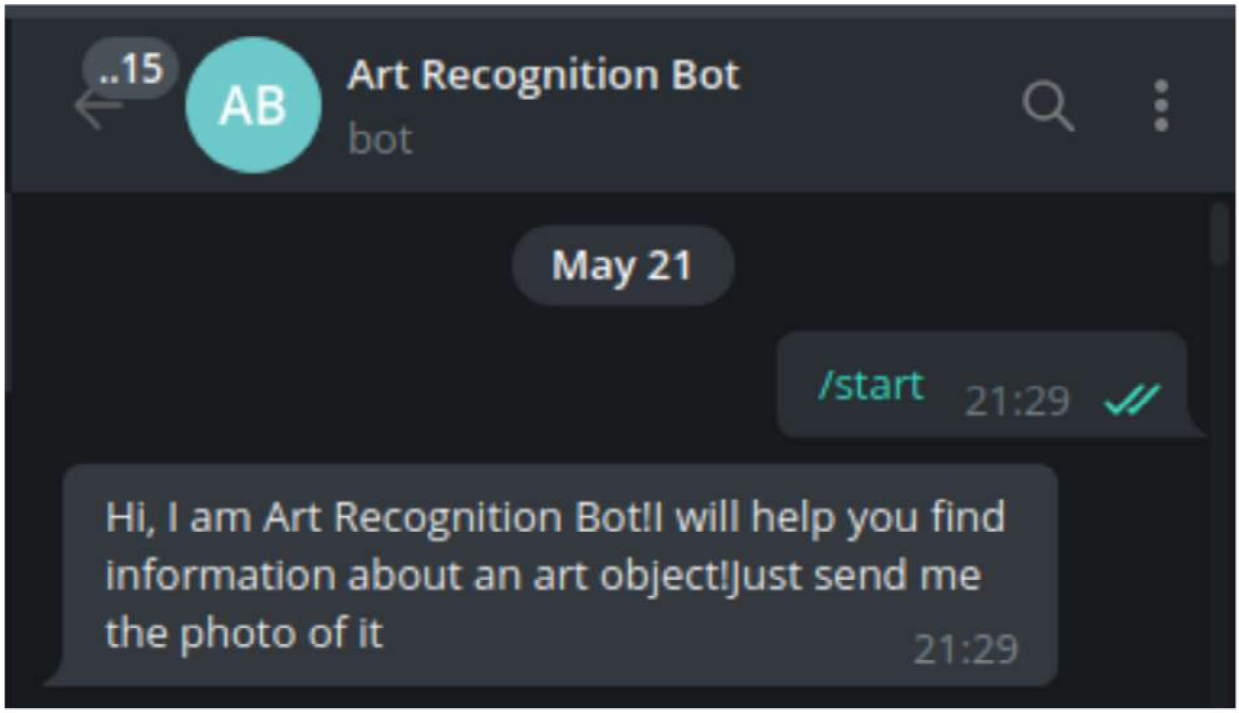

Рисунок 6 — Початкове привітання користувача

Далі користувачу необхідно надіслати зображення мистецького об’єкту, яке він може завантажити з галереї, чи зробити фото за допомогою камери вбудованої в смартфон. Він отримає результати пошуку і набір кнопок для обрання потрібного йому мистецького об’єкту (Рисунок 7).

Далі користувачу пропонується обрати серед наведених варіантів той, що найбільш схожий на те, що турист бачить перед собою при користуванні ботом. Обирання варіанту відбувається за рахунок наданих кнопок, номер кнопки відповідає порядковому номеру зображення у альбомі надісланому на одне повідомлення раніше. Такі кнопки можуть залишатися в чаті необмежену кількість часу, що зручно, якщо користувач випадково натиснув невірну

кнопку, або він захоче через деякий час отримати інформацію про якийсь новий для нього витвір серед тих, що колись пропонував йому бот. Після натискання користувач отримує детальну інформацію про обраний об’єкт мистецької творчості.

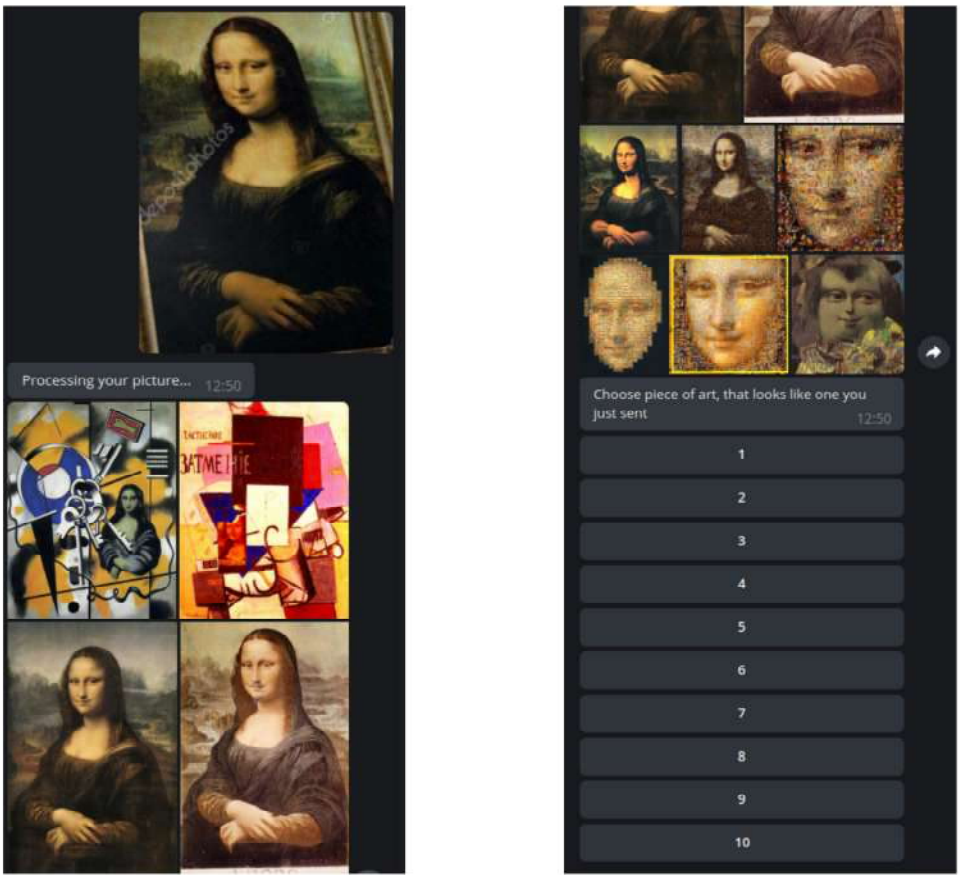

Рисунок 7 — Отримання результатів розпізнавання

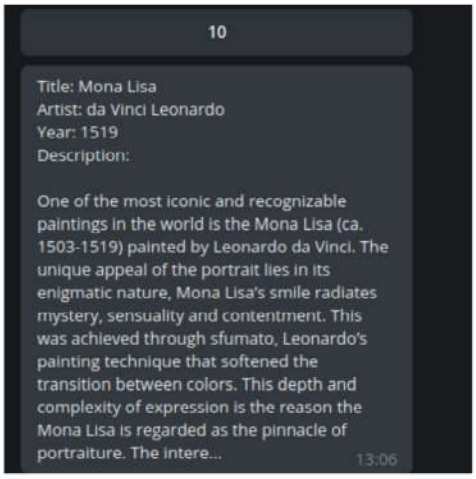

Рисунок 8 — Отримання повної інформації про мистецький об’єкт

3.4 Тестування програмного продукту

Було проведено тестування створеного програмного продукту. Було проведено три різних види тестування:

- Тестування зображень різної якості і різного розширення
- Тестування на зображеннях з зайвою інформацією (таких як рамка картини, оточення мистецького об’єкту тощо), а також на зображеннях, на яких мистецький об’єкт міститься неповністю.
- Тестування на зображеннях обернутих навколо трьох осей обертання

В результаті проведеного тестування додаток показав себе добре в роботі з зображеннями поганої якості. На другому етапі тестування було виявлено, що програмний застосунок добре працює коли мистецький об’єкт складає хоча б 80 відсотків наданого зображення. Якщо зображеня містить мистецький об’єкт неповністю, то стабільну роботу можливо забезпечити, якщо в кадрі наявно щонайменше 85 відсотків мистецького витвору, якщо в кадр попадає менший відсоток, важливо, щоб фрагмент містив основні образи об’єкту мистецької творчості (наприклад, в картині “Мона Ліза” Леонардо да Вінчі таким є жіночий образ в центрі зображення). На третьому етапі тестування було виявлено, що програмний продукт здатний стабільно розпізнавати зображення, що обернуті не більше ніж на 10 градусів по будь-якій з осей обертання.

ВИСНОВКИ

У роботі було розглянуто сучасні методи донесення інформації про об’єкти мистецької творчості туристам. Було виявлено, що у сучасних підходів наявні недоліки, які поки що жоден з існуючих інструментів не здатний виправити. Було виявлено, що таких вад буде позбавлений застосунок, що буде доступний за допомогою мобільного гаджета.

В ході роботи було виконано всі поставлені завдання:

- Виявлено, в якому вигляді користувачу буде найзручніше взаємодіяти з інструментом, що позбавлений наявних недоліків сучасних методів донесення інформації про витвори мистецтва;
- Проаналізовано наявні засоби для розробки програмної частини інструменту, та обрано найзручніші серед наявних альтернатив. Було вирішено створити бота в Telegram з доволі простим інтерфейсом, що складається з двох частин: Запит на розпізнавання зображення та отримання результатів, обрання серед результатів розпізнавання найбільш схожий об’єкт мистецтва та отримання повної інформації про нього;
- Спроектовано та розроблено програмну частину. Програмний продукт отримав планований простий інтерфейс. Система проводить попередню обробку зображення перед відправкою на сервіс з розпізнавання зображень. Система здатна робити запити на сервіс з розпізнавання (Google Cloud Vision) з різними параметрами для підвищення точності розпізнавання. Застосунок отримує інформацію про об’єкти мистецької творчості зі спеціалізованого інтернет-ресурсу WikiArt. Програмний продукт було запущено на сервісі Heroku для постійної роботи;

– Протестовано отриману систему кількома методиками, було отримано задовільні результати. Система показала свою працездатність, а також хороший рівень гнучкості в плані ракурсів та якості фотознімків мистецьких об’єктів.

Розроблений програмний продукт стане гарним супутником для туристів в справі інтерактивного вивчення об’єктів мистецької творчості. Застосунок призначається для особистого використання за допомогою смартфона, планшета або навіть персонального комп’ютера, якщо, наприклад, є бажання отримати інформацію про витвір мистецтва зі старих фотографій.

Є доцільним продовжити просування в розробці даного програмного продукту. Основним, що може бути покращено, є попередня обробка зображення, наприклад додання можливості вирівнювати перспективу може підвищити гнучкість застосунку в питанні розпізнавання об’єктів мистецтва в кадрі з відхиленням по осям обертання відносно самого витвору мистецтва.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Офіційна документація Python 3.8 — Режим доступу: https://docs.python.org/3.8/ (переглянуто 05.06.2022)
2. Офіційна документація pyTelegramBotAPI — Режим доступу: https://pytba.readthedocs.io/en/latest/index.html (переглянуто 05.06.2022)
3. Офіційна документація обгортки на Google Cloud Vision API для мови програмування Python — Режим доступу: https://googleapis.dev/python/vision/latest/index.html (переглянуто 05.06.2022)
4. Офіційна документація OpenCV — Режим доступу: https://docs.opencv.org/4.x/ (переглянуто 05.06.2022)
5. Стаття про медіанний фільтр з ресурсу Medium.com — Режим доступу: https://medium.com/@florestony5454/median-filtering-with-python-and-opencv-2bce390be0d1 (переглянуто 05.06.2022)
6. Офіційна документація WikiArt API — Режим доступу: https://docs.google.com/document/d/1T926unU7mx9Blmx3c8UE0UQTnO3MrDbXTGYVerVQFDU/edit# (переглянуто 05.06.2022)
7. Офіційна документація Heroku — Режим доступу: https://devcenter.heroku.com/ (переглянуто 05.06.2022)
8. Статистика по кількості мобільних телефонів у світі — Режим доступу: https://www.bankmycell.com/blog/how-many-phones-are-in-the-world (переглянуто 05.06.2022)
9. Статистика по часу, що витрачається людьми на соціальні мережі — Режим доступу: https://techjury.net/blog/time-spent-on-social-media/ (переглянуто 05.06.2022)
10. Стаття “Telegram став лідером серед додатків за темпами зростання користувачів” — Режим доступу: https://mind.ua/news/20234354-

telegram-stav-liderom-sered-dodatkiv-za-tempami-zrostannya-koristuvachiv (переглянуто 05.06.2022)

11. Дослідження “Смартфони проти камер: скорочення розриву в якості фото” від ресурсу DXOMark — Режим доступу: https://www.dxomark.com/smartphones-vs-cameras-closing-the-gap-on-image-quality/ (переглянуто 05.06.2022)