

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМ. В. ДАЛЯ  
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ТА ЕЛЕКТРОНІКИ  
КАФЕДРА КОМП'ЮТЕРНИХ НАУК ТА ІНЖЕНЕРІЇ

До захисту допускається  
Т.в.о. завідувача кафедри  
\_\_\_\_\_ Сафонова С.О.  
« \_\_\_\_ » \_\_\_\_\_ 20\_\_ р.

**МАГІСТЕРСЬКА РОБОТА**

НА ТЕМУ:

Дослідження способів побудови доповненої реальності і реалізація системи  
під платформу Android

Освітньо-кваліфікаційний рівень “Магістр”  
Спеціальність 122 “Комп’ютерні науки ”

Науковий керівник роботи:

\_\_\_\_\_

(підпис)

С.О. Сафонова

\_\_\_\_\_

(ініціали, прізвище)

Консультант з охорони праці:

\_\_\_\_\_

(підпис)

Я.О. Критська

\_\_\_\_\_

(ініціали, прізвище)

Студент:

\_\_\_\_\_

(підпис)

Д.В. Гаврилов

\_\_\_\_\_

(ініціали, прізвище)

Група:

КН -18дм

Северодонецьк 2020

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
ІМЕНІ ВОЛОДИМИРА ДАЛЯ

Факультет Інформаційних технологій та електроніки

Кафедра комп'ютерних наук та інженерії

Освітньо-кваліфікаційний рівень “магістр”

Спеціальність 122 – “Комп'ютерні науки”

(шифр і назва)

Спеціалізація \_\_\_\_\_

(шифр і назва)

**ЗАТВЕРДЖУЮ:**

Т.в.о.зав. кафедри КНІ

к.т.н., доц. С.О. Сафонова

« \_\_\_\_\_ » \_\_\_\_\_ 20\_\_ р.

**З А В Д А Н Н Я  
НА МАГІСТЕРСЬКУ РОБОТУ СТУДЕНТУ**

Гаврилов Данило Вадимович

(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження способів побудови доповненої реальності і реалізація системи під платформу Android

керівник проекту (роботи) Сафонова С.О., к.т.н., доцент

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом вищого навчального закладу від «11» 10 2019 р. № \_\_\_\_\_

2. Строк подання студентом роботи 15.01.2020

3. Вихідні дані до роботи Матеріали науково-дослідної практики.

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

1. Огляд технологій доповненої реальності

2. Дослідження алгоритмів побудови доповненої реальності без маркерів.

3. Дослідження алгоритмів побудови доповненої реальності на основі розпізнавання маркерів

4. Дослідження алгоритмів читання QR-коду

5. Розробка програмного додатку, аналіз результатів.

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)  
Електронні плакати

## 6. Консультанти розділів проекту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Охорона праці	<i>Критська Я.О.</i>		

7. Дата видачі завдання 06.09.2019

Керівник

\_\_\_\_\_ (підпис)

Завдання прийняв до виконання

\_\_\_\_\_ (підпис)

## КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів проекту ( роботи )	Примітка
1	Отримання завдання до магістерської роботи	<i>10.09.19-17.09.19</i>	
2	Аналіз завдання, огляд літератури	<i>18.09.19-25.09.19</i>	
3	Аналіз технічних засобів	<i>26.09.19-18.10.19</i>	
4	Розробка методу	<i>19.10.19-06.11.19</i>	
5	Програмна реалізація	<i>17.11.19-08.12.19</i>	
6	Охорона праці	<i>09.12.19-15.12.19</i>	
7	Оформлення пояснювальної записки	<i>16.12.19-29.12.19</i>	
8	Підготовка презентації та доповіді	<i>03.01.20-12.01.20</i>	

Студент

\_\_\_\_\_ ( підпис )

*Гаврилов Д.В.*

\_\_\_\_\_ (прізвище та ініціали)

Науковий керівник

\_\_\_\_\_ ( підпис )

*Сафонова С.О.*

\_\_\_\_\_ (прізвище та ініціали)

## АННОТАЦІЯ

Гаврилов Д.В. Дослідження способів побудови доповненої реальності і реалізація системи під платформу ANDROID.

Метою магістерської роботи є дослідження та розробка способів побудови доповненої реальності і реалізація системи під платформу Android. Результатом роботи є програмна реалізація системи, що складається з двох додатків під платформу Android, які мають спільну архітектуру та частину коду, що дозволить будувати на їх основі інші додатки доповненої реальності.

**Ключові слова:** доповнена реальність, оптичний трекінг, алгоритми побудови AR, метод Віоли-Джонса, алгоритм читання QR-коду, калібрування камери.

## АННОТАЦИЯ

Гаврилов Д.В. Исследование способов построения дополненной реальности и реализация системы под платформу ANDROID.

Целью магистерской работы является исследование и разработка способов построения дополненной реальности и реализация системы под платформу Android. Результатом работы является программная реализация системы, состоящей из двух приложений под платформу Android, которые имеют общую архитектуру и часть кода, что позволит строить на их основе другие приложения дополненной реальности.

**Ключевые слова:** дополненная реальность, оптический трекинг, алгоритмы построения AR, метод Виолы-Джонса, алгоритм чтения QR-кода, калибровки камеры.

## ABSTRACT

Havrylov D.V. Research on ways to build augmented reality and implement the system under the ANDROID platform.

The aim of the work is to develop and explore methods of construction of the augmented reality and to realize system for the Android platform. The result of the work is implemented program implementation of the system, that consist of two Android-based applications, which have a common architecture and part of the code, that will allow them to build on their basis other programs.

**Key words:** augmented reality, optical tracking, AR construction algorithm, method of Viola-Jones, QR code reading algorithm, camera calibration.

## ЗМІСТ

Перелік умовних позначень.....	7
Вступ .....	8
1 Аналіз предметної області та актуальність.....	9
1.1 Історія розвитку технології доповненої реальності.....	9
1.2 Сучасний стан технології AR .....	10
1.3 Сфери застосування AR.....	12
1.4 Сучасні тенденції розвитку AR в Україні і світі.....	15
1.5 Постановка задачі дослідження.....	19
2 Технології доповненої реальності для вдосконалення інтерфейсу користувача.....	20
2.1 Апаратне забезпечення AR.....	20
2.2 Програмне забезпечення та трекінг.....	22
2.2.1 Трекінг на основі датчиків.....	22
2.2.2 Оптичний трекінг .....	23
2.3 Візуалізація доповненої реальності.....	24
2.4 Графічний інтерфейс користувача .....	25
2.5 Засоби розробки програмного забезпечення AR .....	25
2.6 Перспективи розвитку AR .....	26
3 Алгоритми побудови доповненої реальності без маркерів .....	27
3.1 Основна схема побудови доповненої реальності.....	28
3.2 Метод Віюлі-Джонса для розпізнавання облич.....	30
4 Алгоритми побудови доповненої реальності на основі розпізнавання маркерів .....	32
4.1 Основна схема побудови доповненої реальності.....	33
4.2 Задіяні алгоритми.....	38
4.2.1 Алгоритм читання QR-коду.....	39
4.2.2 Вирішення проблеми PnP .....	40
4.2.3 Калібрування камери .....	41
4.3 Побудова доповненої реальності.....	41
5 Програмна реалізація додатку доповненої реальності .....	45
5.1 Загальний дизайн системи .....	45
5.2 Застосовані технології та бібліотеки .....	48
5.3 Деталі реалізації додатку для побудови доповненої реальності без маркерів .....	49

5.4 Деталі реалізації додатку для побудови доповненої реальності на основі розпізнавання маркерів.....	52
6 Охорона праці та безпека в надзвичайних ситуаціях .....	57
6.1 Загальні питання з охорони праці .....	57
6.1.1 Правові та організаційні основи охорони праці .....	39
6.1.2 Організаційно-технічні заходи з безпеки праці.....	59
6.2 Аналіз стану умов праці.....	60
6.2.1 Вимоги до приміщень .....	60
6.2.2 Вимоги до організації місця праці.....	61
6.2.3 Навантаження та напруженість процесу праці .....	62
6.3 Виробнича санітарія.....	63
6.3.1 Аналіз небезпечних та шкідливих факторів при виробництві (експлуатаці) виробу.....	52
6.3.2 Пожежна безпека.....	64
6.4 Гігієнічні вимоги до параметрів виробничого середовища .....	65
6.4.1 Освітлення .....	65
6.5 Вентилювання .....	67
6.6 Розрахунок захисного заземлення (забезпечення електробезпеки будівлі) .....	67
6.7 Екологія .....	69
6.8 Висновки до розділу 6 .....	70
6.9 Перелік корисних посилань до розділу 6 .....	71
Висновки .....	73
Перелік джерел посилань .....	74
Додаток А Лістинг «CvEngineLive2d.java».....	76
Додаток Б Лістинг «GlEngineLive2d.java».....	79
Додаток В Комп'ютерна презентація .....	81

**ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ**

FPS – frames per second, кадрів на секунду

AR – augmented reality, доповнена реальність

XML – extensible markup language, розширювана мова розмітки

SDK – software development kit, інструменти для розробки програмного забезпечення

ПК – персональний комп'ютер

## ВСТУП

Доповнена реальність (AR – augmented reality) – одна з найперспективніших технологій XXI століття. Зараз вона використовується практично скрізь: від ігрової індустрії до медицини.

Історія доповненої реальності почалася ще в 1961 році. З кожним роком технологія удосконалюється і стає звичним і корисним інструментом, а не просто вражаючою іграшкою.

Технологію доповненої реальності часто плутають з віртуальною реальністю (VR – virtual reality). Головна відмінність між ними у тому, що віртуальна реальність складається тільки з нереальних, створених в програмі об'єктів. Надіваючи VR–шолом, повністю опиняємося в штучно створеному світі. Доповнена реальність – це коли нереальні, віртуальні об'єкти в сприйнятті користувача стають частиною реальної навколишньої картини світу.

Тобто, коли людина бачить щось в AR, вона бачить віртуальні об'єкти у реальному світі. Дисплей показує користувачеві фізичний світ з доданими віртуальними об'єктами. Наприклад, маркер карти в інтерфейсі камери на телефоні – щоб показати, в якому напрямі рухатися користувачеві. Вже існують гарнітури, які фактично переносять додаток або гру в реальний світ. Більш просунуті AR–системи дозволяють обробляти стіни у будинку користувача, начебто вони є додатками на екрані комп'ютера.



# 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА АКТУАЛЬНІСТЬ

## 1.1 Історія розвитку технології доповненої реальності

Розробкою технології доповненої реальності дослідники займаються не перший рік. У 1961 році кінооператор Мортон Хайліг представив імерсивний мультисенсорний пристрій, що нагадує своєрідну аркадну гру з вібрацією і відтворенням стереофонічних звуків.

Наступного року Хайліг отримав патент на перший у світі віртуальний симулятор під назвою "Сенсорама". Величезний пристрій, зовні схожий на ігрові автомати 1980-х, дозволяв глядачеві уперше зануритися у віртуальну реальність: наприклад, проїхатися на мотоциклі по вулицях Брукліну. Проте "Сенсорамою" не зацікавилися інвестори і розробки довелося згорнути.

Наступним етапом розвитку технології прийнято вважати 1974 рік, коли комп'ютерний фахівець Майрон Крюгер розробив лабораторію "штучної реальності" під назвою Videoplace.

Вона представляла з себе декілька пов'язаних мережею кімнат, в кожній з яких знаходився великий екран з розташованим позаду нього відеопроєктором. Коли людина заходила до кімнати, вона бачила на екрані своє власне зображення у вигляді примітивного силуету, а також подібні силуети людей в інших кімнатах. В усіх "тіней" можна було міняти колір або розмір, а також приєднувати до них різні візуальні об'єкти.

Можливо, ідеї Крюгера і його друзів спонукали вченого Тома Кодэлла уперше запропонувати термін "доповнена реальність" в 1990 році. Працюючи в компанії Boeing Computer Services в Сітлі, він використав словосполучення для позначення цифрового дисплею на голові, використовуваного електриками літаків, які змішували віртуальну графіку з фізичною реальністю [1].

У 1992 році перша діюча AR-система почала використовуватися військовими ВПС США. Вона дістала назву "Віртуальні світильники" і дозволила створити новий метод навчання пілотів. За допомогою накладення фізично реальних об'єктів на 3d-віртуальні з'явився перший справжній досвід доповненої реальності, що забезпечує зображення, звук і дотик.

Приблизно в той же час в університеті штату Колумбія відбулася презентація системи KARMA ("Помічник в доповненій реальності"), що дозволяє через шолом віртуальної реальності побачити інтерактивну інструкцію з обслуговування принтера.

Але до 1999 року доповнена реальність не використовувалася широко, а багатьма ученими і дослідниками навіть не розумілася. Для її роботи використовувалися складні програмні рішення і громіздке устаткування.

Проте ситуація різко змінилася, коли японський професор Хироказу Като з Інституту науки і технологій Нари винайшов унікальне програмне забезпечення під назвою ARToolKit. Воно дозволило відстежувати відеозахват дій у реальному світі і об'єднати їх з віртуальними об'єктами. Забезпечення могло бути пов'язане з простим кишеньковим пристроєм: наприклад, камерою і підключенням до інтернету. Поява ARToolKit призвела до того, що тепер користувачі бачили безпосередньо сам процес роботи доповненої реальності.

Вже в 2000 році Брюс Томас з лабораторії Wearable Computer розробив першу мобільну гру для відкритого простору з системою доповненої реальності, названу ARQuake. Вона дозволила користувачеві з прикріпленим цифровим дисплеєм на голові повернути голову і побачити інші об'єкти віртуального світу. ARQuake з успіхом презентували на Міжнародному симпозіумі з мобільних комп'ютерів.

Через декілька років в 2008 році перші AR-додатки були створені для смартфонів, і люди по всьому світу змогли уперше скористатися новітньою технологією. Перше застосування призначалося для користувачів Android, і це дозволило їм використати свої камери, щоб побачити на екрані різні об'єкти віртуальної реальності в 3d. Рішення незабаром з'явилося і на iPhone, і запущено в якості навігаційного додатку, названого Wikitude Drive.

У липні 2009 року читачі американського журналу Popular Science отримали незвичайний номер: якщо піднести його до веб-камери, на дисплеї ПК можна було побачити тривимірне анімоване зображення вітряної електростанції, яка виступає прямо з обкладинки. А якщо ще й подути в мікрофон, то пропелери почнуть крутитися швидше.

## 1.2 Сучасний стан технології AR

Найбільш великий інтерес до AR у видатних технологічних гігантів – Google, Apple і Microsoft. Завдяки фінансовим можливостям і штату розробників вони знаходяться ближче за усіх до того, щоб створити по-справжньому масові продукти з використанням доповненої реальності.

Розумні окуляри доповненої реальності Google Glass (рис. 1.1) були представлені відносно нещодавно - в 2012 році. Після презентації були випущені прототипи пристрою

для розробників, і почався довгий процес тестування продукту. Для широкої аудиторії окуляри Google Glass стали доступні в травні 2014 року. Їх ціна на той момент складала 1500 доларів.



Рисунок 1.1 – Окуляри доповненої реальності Google Glass

Незважаючи на інноваційність, через декілька років стало зрозуміло, що проект "Розумних окулярів" від Google провалився і продажі девайса довелося згорнути.

Причин цьому декілька: скандали з конфіденційністю даних (за сформульованими Google умовами використання і політикою конфіденційності, звуки і зображення, які записує облаштування Google Glass, не є власністю людини, що носить окуляри) і цінове позиціонування.

Проте Google продовжує розробки у сфері AR. В першу чергу, це стосується платформи для обчислень доповненої реальності Project Tango. З його допомогою розробники планують дати мобільним пристроям людське розуміння простору і руху.

Заслуги Apple у галузі AR доки скромні, проте компанія продовжує розробки. Головне досягнення компанії – технологія доповненої реальності ARKit, яка дозволяє розпізнавати габарити реальних об'єктів і враховувати умови освітлення, щоб максимально достовірно інтегрувати віртуальні об'єкти в реальне життя. Сумісна з iOS технологія може стати наймасовішою платформою доповненої реальності у світі, враховуючи інтерес до продуктів компанії.

У відповідь на поширення ARKit експерти чекають подальше просування окулярів змішаної реальності Microsoft HoloLens і технології доповненої реальності Google Tango [2].

Microsoft також щосили прагне до створення доповненої реальності в якості платформи для майбутнього мобільних технологій. У компанії об'єднали віртуальні і доповнені світи для користувачів, створивши, можливо, перше уявлення про змішану

реальність завдяки гарнітурі Microsoft HoloLens (рис. 1.2). Від інших облаштувань доповненої реальності вони відрізняються тим, що перейняли у VR-шоломів можливість відстежувати найдрібніші рухи голови користувача. Процес відбувається за допомогою звичайного гіроскопа і акселерометра і дозволяє не лише прискорити обробку даних в особливих випадках, але і доповнити управління жестами.



Рисунок 1.2 – AR-окуляри Microsoft HoloLens

Крім того, Microsoft вже декілька років продовжує розробляти платформу змішаної реальності Windows Mixed Reality, заявлену як частину операційної системи Windows 10. За даними розробників, вона забезпечує "голографічний досвід" в змішаній реальності з сумісними шоломами. Ймовірно, в компанії розраховують, що з розвитком AR користувачам знадобиться уніфікуючий інтерфейс і спільна платформа.

### 1.3 Сфери застосування AR

У освіті AR може бути використана для відтворення історичних подій або читання звичайних книг в 3d-проекціях. Доповнена реальність надзвичайно корисна для педагогів в умовах занять в класі або під час презентацій, і дозволяє учням глибше зрозуміти певну тему.

Приклад вже впроваджених технологій – японський додаток New Horizon, який за допомогою вбудованих камер смартфона показує прямо в учбових книгах анімованих персонажів на потрібних сторінках (рис. 1.3).

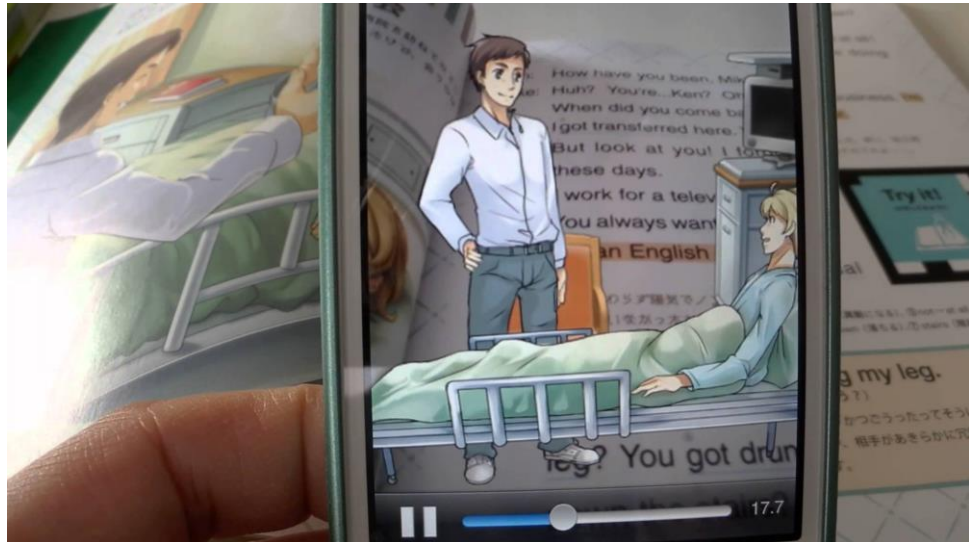


Рисунок 1.3 – Зображення роботи додатку New Horizon

Ще один освітній проект LearnAR, підходить для використання в школі. Використовуючи мітки, можна давати відповіді, а також проводити візуальне відображення будь-яких реальних об'єктів. Наприклад, у біології – кісток людини або його внутрішніх органів (рис. 1.4), в геології – мінералів.

Ще одна сфера застосування AR – охорона здоров'я.

Наприклад, сенсорна програма Palpsim з використанням технології доповненої реальності дозволяє навчатися пальпації стегнової артерії і введенню голки. Ця розробка створює віртуальне середовище, в якому навчаються майбутні лікарі. За допомогою Palpsim можна продіагностувати віртуального пацієнта за допомогою складної сенсорної системи, зробленої на базі гідравлічного механізму. На екран виводиться зображення, але робота, проте, ведеться із справжніми інструментами. Під час проведення різних маніпуляцій студент навіть може відчувати пульс пацієнта і бачити кров, хоч і віртуальну (рис. 1.5).

Додаток ARnatomy вже допомагає майбутнім лікарям вивчити реальну модель скелета, а зорове пристосування VA - ST використовується людьми зі значною втратою зору. Воно створює нариси контурів обличчя співрозмовника [8].

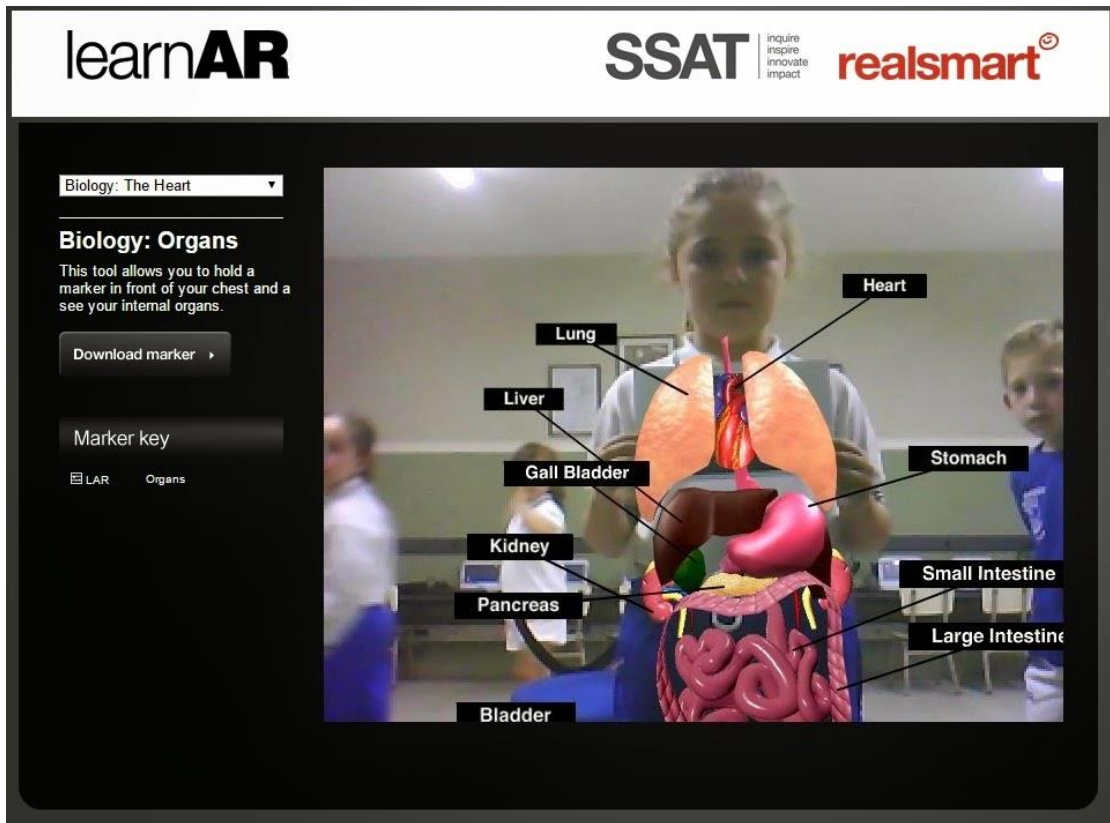


Рисунок 1.4 – Додаток LearnAR



Рисунок 1.5 – Робота додатку Palpsim

Військові теж цікавляться технологією. Американська компанія BAE Systems розробила шолом під назвою Striker II (рис. 1.6), в якому замість окулярів використовується своєрідний козирок з дисплеєм [7]. На нього проектується зображення з камери нічного бачення, а апарат здатен відстежувати рухи голови оператора. Так дані завжди розташовуються у напрямку погляду користувача.



Рисунок 1.6 – Військовий у шоломі Striker II

Apple, Google і Microsoft активно інвестують кошти в AR, щоб зробити технологію ефективнішою і доступнішою для мільярдів користувачів смартфонів.

У одному з інтерв'ю головний виконавчий директор Apple Тім Кук заявив: "AR буде так само важливий, як їжа три рази на день". А нещодавно і Facebook оприлюднив плани роботи з цифровим моделюванням 3d-об'єктів в соцмережах, щоб користувачі по-справжньому занурювалися в серфінг новинної стрічки і спілкуванню з друзями.

#### **1.4 Сучасні тенденції розвитку AR в Україні і світі**

Побачити кожне з семи чудес світу, подивитися нову квартиру, коли забудовник ще навіть не заклав фундамент, зануритися в гру, відчувати себе головним героєм. Усе це, і набагато більше, сьогодні надає технологія доповненої реальності.

Потенціал застосування AR великий. Впровадження елементів доповненої реальності в повсякденне життя здійснюється через багато каналів. За оцінками ряду

аналітичних видань, впродовж наступних п'яти-семи років додатки AR трансформують деякі галузі промисловості. Впровадження елементів доповненої реальності очікується в розважальній індустрії, e-commerce, охороні здоров'я, туризмі і навіть у видавничій справі. Передусім, варто розуміти, що AR зробить революцію в конкретних галузях, повністю їх трансформуючи або частково доповнивши.

Україна не є аутсайдером на арені розвитку технологій доповненої реальності. Кількість таких проектів вже більше десяти. Серед гравців ринку можна виділити AR/VR ательє, які займаються створенням контенту (ByOwls, Lemon Group), компанії, що створюють програмне забезпечення (Lookinar, Program - Ace), і компанії, що створюють власний продукт в одному з напрямів (VRNET, Fablab Fabricator, Sensorama, ARbrowser).

В основному, інвестиції в галузь в Україні є приватними, тому інформація про них фактично відсутня. Для цієї індустрії в Україні 2018 рік можна назвати більш ніж успішним.

Існує проблема у визначенні драйверів зростання. Хто в першу чергу рухатиме ринок? Виробники софту своїми застосовними програмами або ж виробники устаткування, які дозволяють технічно поліпшити відчуття доповненої реальності?

Аналітики зійшлися на тому, що компанії-виробники устаткування на даний момент є локомотивом впровадження ініціативи AR шляхом створення пристроїв для користувачів. Проте індустрія не може добитися успіху без якісного програмного забезпечення, а апаратні компанії працюють над стимулюванням його розробки і зростанням екосистеми.

Додатки все ще розвиваються, примушуючи творців проявляти обережність при виділенні фінансових ресурсів на розробку і визначення того, які апаратні платформи підтримувати. Деякі аналітики стверджують, що ціни на продукцію з AR залишаться відносно високими через її новизну. У подальші роки ціни знизяться, але через масштабування продажів валовий прибуток виробників вийде на органічний рівень [3].

Очікується, що ринок устаткування буде рости повільніше, ніж ринок програмного забезпечення найближчими роками, оскільки споживачі будуть консервативними і обережними в придбанні устаткування AR через його високу вартість в порівнянні з додатками. Проте у міру розвитку, співвідношення об'ємів ринку виробників апаратного забезпечення і розробників софту вирівнюватиметься. За оцінками Goldman Sachs, до 2025 року загальний потенційний об'єм ринку складе \$80 млрд (\$45 млрд hardware- ринок і \$35 млрд. - на software- ринок). Останній продемонструє найбільшу експансію AR/VR технологій в секторі геймінга (виручка складе \$11, 6 млрд), охорони здоров'я (\$5, 1 млрд), інженерії (\$4, 7 млрд) і трансляції різних подій (\$4, 1 млрд) (рис. 1.7).



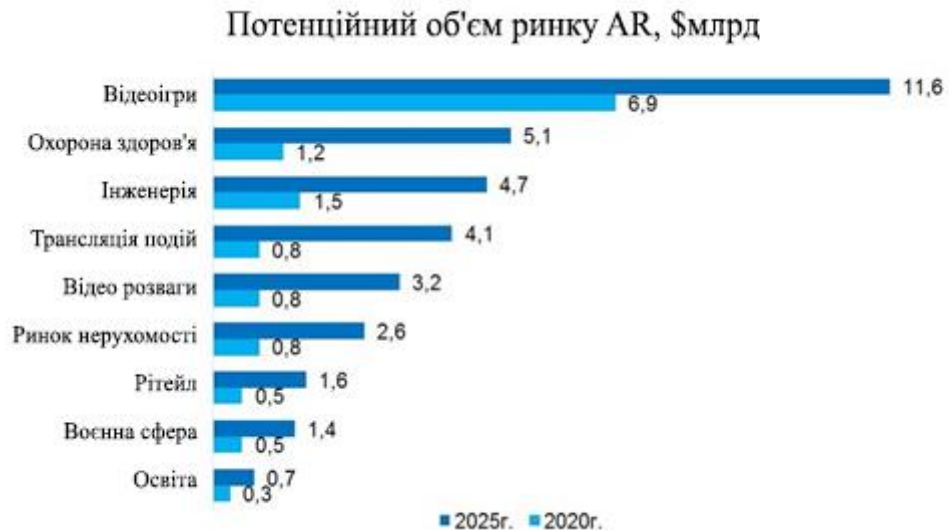


Рисунок 1.7 – Потенційний об'єм ринку AR у 2020 та 2025 році

Результати нового опитування, проведеного міжнародною фірмою Perkins Coie LLP, показують глобально зростаючий тренд розвитку галузі, в якій більше уваги приділяється більш практичним додаткам, і яка в той же час стикається з проблемами клієнтської бази, що розширюється.

Розвиток AR технологій припускає взаємодію між споживачем і виробником. Готовність споживачів абсорбувати пропоновані технології також є важливим драйвером розвитку галузі. Стримуючі ж чинники, згідно з опитуванням компанії Perkins Coie, представлені на рисунку 1.8.

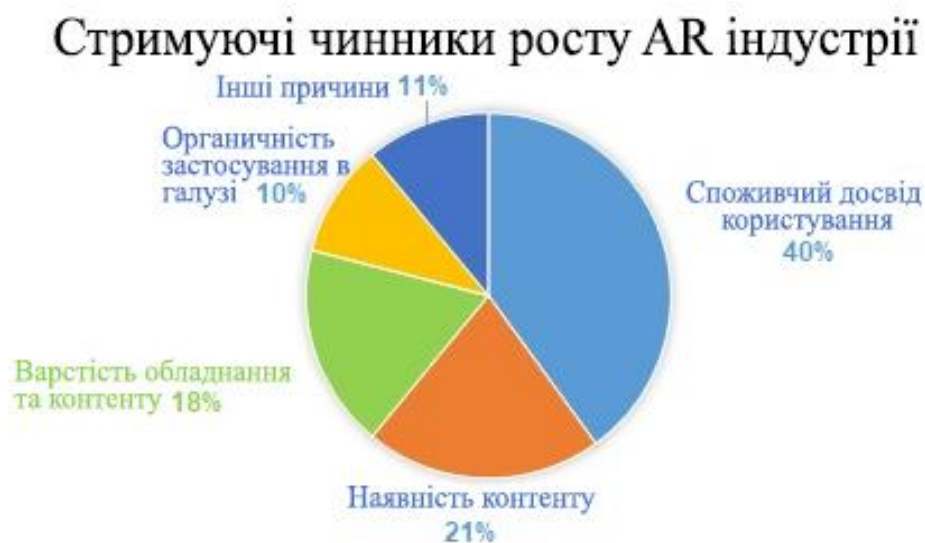


Рисунок 1.8. – Стримуючі чинники росту AR індустрії

Великі технологічні компанії, які хочуть вийти на ринок AR, вважають за краще не витрачати внутрішні ресурси на створення окремого підрозділу, а купити вже існуючий стартап, що займається подібною активністю. Наведена вище теза підтверджується тим, що в 2014 році компанія Facebook купила Oculus, а Google інвестувала \$542 млн в Magic Leap.

Змінився вектор інтересу інвестиційних компаній до AR - розробників. Якщо на початку розвитку галузі спостерігалось інвестування переважно в компанії на ранніх стадіях зростання, яке досягло піку в 2014 році з сумою порядку \$650 млн (процес, в цілому, органічний через молодість самої галузі), то зараз такий вид інвестування знизився до \$250 млн, поступившись інвестуванню в компанії на пізній стадії розвитку (прирівнюється до суми порядку \$850 мільйонів) (рис. 1.9).

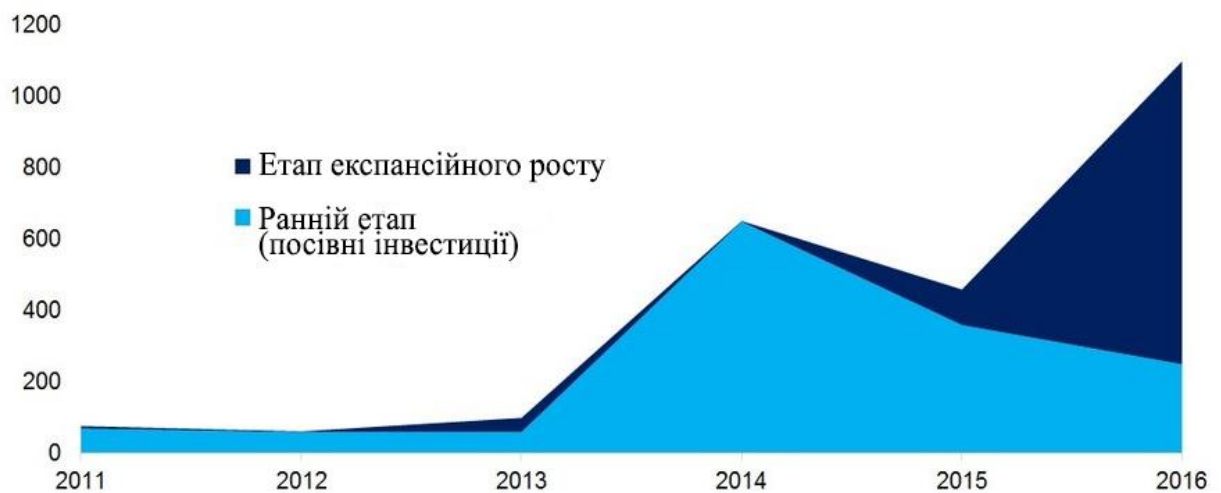


Рисунок 1.9 – Етапи фінансування AR-компаній, \$млн

До цього стратегічні покупці часто вкладали гроші в невелику компанію на ранньому етапі з наміром її придбати, якщо зростання або розвиток компанії перевищить пороговий рівень. На даний момент інвестори змінили свої переваги, схилившись у бік компаній, які вже мають готовий продукт і певну прибуткову структуру (здатність генерації грошового потоку). В майбутньому нас чекають ще більш проривні проекти у сфері доповненої реальності, які фінансуватимуться технологічними гігантами. Проте варто чітко визначити, що треба споживачеві, і як це технологічно утілювати. Баланс в розробці апаратного і програмного забезпечення зможе забезпечити стійкий розвиток галузі, що ще більше наблизить нас до епохи змішаної реальності [4].

## 1.5 Постановка задачі дослідження

Аналіз існуючих систем показав, що в багатьох областях використання доповнена реальність може збільшити ефективність роботи та прибуток. На даний час вона використовується недостатньо широко, в деяких областях тільки починається впровадження цієї технології.

Мета роботи – створити систему, яка покаже потенціал технології доповненої реальності на мобільних пристроях.

Вона буде складатися з двох додатків під платформу Android, які мають спільну архітектуру та частину коду, що дозволить будувати на ній інші додатки доповненої реальності.

Код, що забезпечує отримання зображення з камери, вивід віртуальної частини доповненої реальності на екран та передачу даних між модулями системи, має розташовуватися у спільній між двома проектами частині архітектури. Ця частина є універсальною, тобто не прив'язана до конкретної задачі, та може бути змінена під інші потреби.

Перший додаток буде виводити на екран мультиплікаційного персонажа та анімувати його залежно від знайдених обличчя у кадрі. Він буде мати два режими – коли персонаж залишається нерухомим, лише слідкуючи поглядом за обличчями, та коли копіює переміщення обличчя відносно камери та деякі його риси.

Другий додаток буде відображати віртуальні візитні картки, використовуючи реальну картку з зображенням на ній маркером, як точку відліку. Дані для відображення на віртуальній картці завантажуються з бази даних, щоб відображати більше корисної інформації та дозволити швидко оперувати нею.

Додатки мають підтримуватися багатьма мобільними пристроями на системі Android версії від 4.0, тому важливу роль грає оптимізація компонентів, чутливих до потужності системи.

## 2 ТЕХНОЛОГІЇ ДОПОВНЕНОЇ РЕАЛЬНОСТІ ДЛЯ ВДОСКОНАЛЕННЯ ІНТЕРФЕЙСУ КОРИСТУВАЧА

Технологія доповненої реальності може бути застосована для вдосконалення інтерфейсу користувача при рішенні різних типів прикладних завдань, серед яких:

- навігація – усередині приміщення та на відкритій місцевості;
- візуальні вказівки (ремонт, складання, медична операція, прицілювання і інше);
- контекстне отримання інформації (свідчення по відомому приладу, пристрою, одиниці техніки, ділянці місцевості);
- візуалізація тривимірного об'єкту (меблів, приладу, будівлі).

При цьому безпосередньо способи реалізації можуть відрізнятися залежно від типу і особливостей конкретного завдання. Так, наприклад, можна відмітити застосування різних видів трекінгу, відображення двомірного або тривимірного контенту, використання різного апаратного забезпечення, різної логіки організації тривимірного простору в ПЗ, підготовлене або непідготовлене приміщення, наявність заздалегідь підготовлених тривимірних моделей об'єктів навколишнього оточення.

Існують завдання, для вирішення яких використання технології доповненої реальності несе найбільш суттєві переваги перед іншими методами.

Одним з таких завдань є візуалізація тривимірних об'єктів. Концепція доповненої реальності пропонує досконаліший призначений для користувача інтерфейс за рахунок сукупності способів управління і візуалізації. Керування ракурсом об'єкту, як правило, здійснюється рухами голови користувача або пристрою і є природним, зрозумілим і ефективним. Спосіб візуалізації тривимірного об'єкту шляхом поєднання його зображення з навколишнім оточенням у відповідному ракурсі дозволяє краще сприймати об'єкт, його розміри (за умови відповідності масштабу) і, в деяких випадках, властивості матеріалів [5].

### 2.1 Апаратне забезпечення AR

Пристрої для запуску програм із застосуванням технології доповненої реальності можуть бути класифіковані за їх положенням між користувачем і реальною обстановкою

на головні (наприклад, окуляри, шоломи), ручні (наприклад, смартфони, КПК, планшетні комп'ютери) і просторові (рис. 2.1).

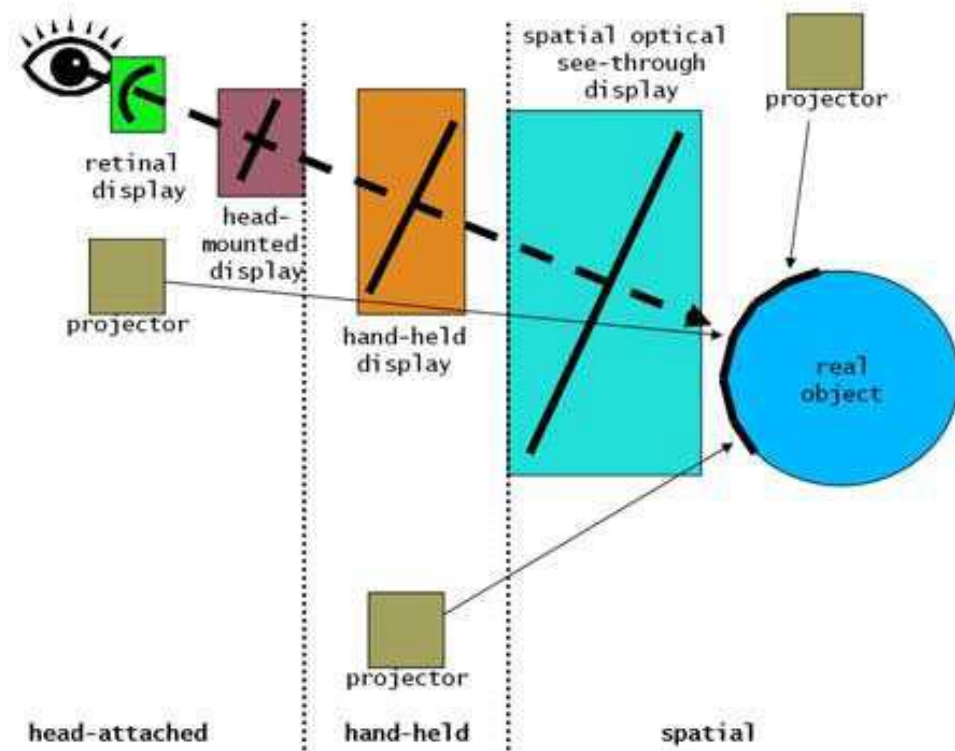


Рисунок 2.1 – Класифікація пристроїв AR

Спочатку, застосування доповненої реальності зв'язувалося дослідниками з використанням різних головних дисплеїв, спеціальних окулярів і шоломів. Але рівень розвитку технологій не дозволяв говорити про високу мобільність рішень і, тим більше, розраховувати на їх масове впровадження.

Розвиток мобільних пристроїв запропонував широкі можливості для дослідників і розробників. Такі пристрої мають камеру, дисплей і достатні обчислювальні можливості для запуску додатків з доповненою реальністю.

Форм-фактор пристроїв надає мобільність, можливість використання в ширшому колі завдань, у тому числі поза приміщенням. Деякі дослідники вважають, що мобільний телефон – це найкраща платформа для доповненої реальності.

Варто відмітити, що використання мобільного пристрою є зараз соціально прийнятнішим в порівнянні з головними пристроями. Проте, останнім часом спостерігаються маркетингові зусилля, спрямовані на популяризацію таких пристроїв, як окуляри доповненої реальності [6].

Рівень технологій виробництва дисплеїв, елементів живлення і інших апаратних компонентів дозволяє говорити про передумови масового виробництва нових головних

пристроїв. Їх головною перевагою перед ручними є краще сприйняття тривимірного простору за рахунок стереоскопічного відображення і природної навігації рухами голови.

Проте питання повної відмови від ручних пристроїв в найближчому майбутньому далеко неоднозначне. Робота з повністю віртуальним графічним інтерфейсом користувача, хай і в тривимірному просторі, є не такою ергономічною, як, наприклад, робота з сенсорним екраном і тим більше з апаратними елементами управління. У користувача повністю відсутній тактильний зв'язок з інтерфейсом.

Таким чином, видається більш обґрунтованим паралельний розвиток головних і ручних пристроїв для використання з метою візуалізації тривимірних об'єктів в доповненій реальності. До того ж сучасні головні пристрої базуються, в основному, на тому ж програмному забезпеченні, що і ручні.

## **2.2 Програмне забезпечення та трекінг**

Для реалізації технології доповненої реальності потрібні два основні програмні компоненти: трекінг і візуалізація. Під трекінгом розуміється відстежування положення камери пристрою відносно системи координат. Віртуальна камера в тривимірному просторі "синхронізується" з фізичною камерою, і за рахунок цього досягається коректне відображення тривимірної моделі в навколишньому оточенні.

### **2.2.1 Трекінг на основі датчиків**

Трекінг в сучасних мобільних пристроях частково може бути здійснений за рахунок вбудовуваних в них датчиків прискорення і положення пристрою. Таким чином можливо визначити нахил пристрою по трьох осях. При цьому зміщення залишиться невідомим. Безперечною перевагою цього способу є відсутність необхідності в якому-небудь калібрувальному зображенні. Проте до його мінусів, окрім нефункціонуючого зміщення, також відноситься неточність, пов'язана з висотою, на якій знаходиться пристрій: вона визначається або приблизно, виходячи з людського зросту, або жорстко задається вручну. Також існує проблема коректного масштабування віртуальних об'єктів відносно реальних.

Трекінг на основі датчиків потенційно може використовуватися у поєднанні з додатковими системами позиціонування (наприклад, ГЛОНАСС, GPS або системи для

позиціонування усередині приміщень). Проте мала точність поточних систем може бути використана тільки для перегляду дуже великих об'єктів.

Також використання даних з датчиків прискорення і положення є одним з прийомів по поліпшенню роботи оптичного трекінгу. В деяких випадках це дозволяє компенсувати недоліки оточення з патернами, що повторюються, такими як однакові вікна будівель.

### 2.2.2 Оптичний трекінг

Найбільш поширеним зараз є оптичний трекінг. За даними досліджень більше 80% робіт, прийнятих на найбільшу наукову подію по доповненій реальності, – Міжнародний симпозиум по змішаній і доповненій реальності (International Symposium on Mixed and Augmented Reality – ISMAR), присвячені оптичному трекінгу. Багато дослідників ведуть роботу над вдосконаленням трекінгу. І хоча можливо добитися практично бездоганних результатів в повністю контрольованому приміщенні з множиною допоміжних пристроїв і пристосувань, на практиці розробникам доводиться стикатися з серйозними перешкодами для коректної роботи системи: недостатнє освітлення, відбиваючі і напівпрозорі матеріали, і багато що інше.

Для широкого впровадження розробок необхідна коректна робота на поширених споживчих пристроях. На сьогодні для таких пристроїв характерний обмежений обчислювальний ресурс, економія енергії і наявність найчастіше тільки однієї RGB-камери. Це обмежує круг можливих підходів, застосованих для реалізації трекінгу.

Зараз актуальним є використання оптичного трекінгу з розпізнаванням в зображенні так званих точок, що поступають з камери, інтересу (ключових точок).

Поширеним є спосіб, при якому використовується задане заздалегідь зображення для відстежування, – маркер. У маркері виявляються ділянки, які за певними параметрами можливо буде з деякою високою вірогідністю визначити на зображенні навколишнього оточення. Безпосередньо в процесі трекінгу при успішному розпізнаванні достатньої кількості точок в зображенні навколишнього оточення обчислюється положення камери відносно маркера. Основною важливою перевагою такого підходу є те, що коректне положення камери може бути відновлене навіть при розпізнаванні тільки частини точок маркера у випадках, коли деякі ділянки маркера перекриті якими-небудь перешкодами або не розпізналися з інших причин. Також варто відмітити, що в якості маркера можна використати деяке зображення, наприклад фотографію, в цілях естетики або для інформування користувача.

Значно досконалішим підходом є використання методу одночасної навігації і складання карти (Simultaneous localization and mapping – SLAM). Цей метод застосовується в роботі автономних транспортних засобів і роботів для побудови карти невідомого простору або для оновлення наявної карти при одночасному відстежуванні їх поточного місця розташування. Ще в 2009 році Вагнер (Wagner) і Шмальштиг (Schmalstieg) писали, що метод був застосований в доповненій реальності, але немає реалізацій для мобільних телефонів у зв'язку з їх низькою продуктивністю. Проте на сьогодні метод був успішно застосований у ряді розробок.

При використанні методу одночасної навігації і складання карти безпосередньо в процесі трекінгу зазвичай створюється так звана хмара точок. Це дозволяє використати точки усього зображення навколишнього оточення, не обмежуючись розпізнаванням точок тільки в маркері, а ніби розширюючи його. При цьому значно підвищується зручність використання систем на основі подібних рішень.

Роботи по вдосконаленню алгоритмів SLAM ведуться і нині. У деяких реалізаціях користувачеві при роботі з ПЗ була потрібна попередня побудова карти простору. Цей підхід не знайде застосування в масових продуктах. Велику цінність представляють системи з мінімальними підготовчими діями для користувача. Тому має сенс поєднувати підходи з використанням маркера і SLAM. При такому поєднанні точки з маркера виступають первинними точками хмари. Після первинного розпізнавання маркера хмара розширюється точками оточення. Важливим є те, що при відомих розмірах маркера є можливість задати масштаб хмари і, відповідно, відобразити віртуальні об'єкти в реальному розмірі. Цей підхід є найбільш актуальним зараз при використанні пристроїв з одною RGB-камерою.

### **2.3 Візуалізація доповненої реальності**

Варто відмітити, що візуалізація є одним з аспектів, що найбільш пропрацьовані, в реалізації технології доповненої реальності. Відносно мобільних пристроїв зараз в чомусь повторюється шлях розвитку тривимірної графіки на настільних комп'ютерах: розробники стикаються з тими ж обмеженнями в ресурсах і для вирішення застосовують добре вивчені методи оптимізації, імітації освітлення і відображень і багато що іншого.



## 2.4 Графічний інтерфейс користувача

В графічному інтерфейсі користувача успішно поєднуються двомірні і тривимірні елементи. Тексти і меню оптимально відображати в двомірному виді. Тоді як самі об'єкти і деякі допоміжні елементи візуалізуються в тривимірному просторі.

Графічний інтерфейс повинен передбачати логіку вибору одного або декількох об'єктів для візуалізації. Список об'єктів може бути реалізований як окремий екран каталогу. Але може відображатися і поверх відеопотоку з камери. В цьому випадку користувач має можливість швидше змінити об'єкт, а також контекст використання стає зрозумілішим. До мінусів відноситься ресурсоємність підходу. Не можна також не відмітити, що картинка з камери може в деякій мірі відволікати увагу користувача.

У разі, якщо передбачена зміна параметрів об'єктів, таких як варіанти розташування модулів, розміри і інші опції, реалізується також інтерфейс зміни параметрів об'єктів. Деякі параметри можна задати до входу в режим візуалізації. Але безпосередньо в процесі візуалізації зазвичай також потрібний інтерфейс для зміни таких параметрів, як, наприклад, матеріали об'єкту і колірні рішення.

## 2.5 Засоби розробки програмного забезпечення AR

Важливо відмітити серйозний розвиток засобів розробки програмного забезпечення. Те, що раніше вимагало великих ресурсів, зараз на основі напрацювань реалізується істотно швидше і простіше.

Зокрема можна виділити інструмент для розробки дво- і тривимірних застосувань і іgr-Unity, що підтримує провідні платформи для мобільних пристроїв. Примітне його використання в контексті реалізації додатків для візуалізації тривимірних об'єктів із застосуванням технології доповненої реальності. Цей засіб підтримує більшість ручних, а також і головних пристроїв для запуску додатків з доповненою реальністю. За допомогою різних програмних розширень, таких як Qualcomm Vuforia, підтримується функціонал трекінгу і інші подальші можливі розширення (для підтримки нових алгоритмів і апаратури). Також вбудованими засобами Unity можливо реалізувати необхідний графічний інтерфейс користувача, і існує можливість сумісності із сторонніми засобами реалізації інтерфейсу.

## 2.6 Перспективи розвитку AR

На даний момент актуальним залишається розвиток ручних і головних мобільних пристроїв. Потрібне підвищення обчислювальних потужностей, роздільної здатності дисплеїв і інших характеристик. Істотним кроком видається впровадження в ці облаштування структурних сенсорів і додаткових RGB- камер.

Переваги, які надає структурний сенсор, включають:

- відсутність необхідності в маркері або калібрувальному зображенні;
- миттєве обчислення розмірів предметів і приміщення;
- реалістична фізика (віртуальні об'єкти не проникають крізь стіни і крізь інші предмети);
- коректні оклюзії (правильне накладення реальних об'єктів на віртуальні).

Відповідно програмне забезпечення повинне повною мірою використати переваги цих сенсорів. Зокрема потрібна їх інтеграція при реалізації методу одночасної навігації і складання карти.

Отже, технологія доповненої реальності може бути успішно застосована для вдосконалення призначеного для користувача інтерфейсу візуалізації тривимірних об'єктів.

### 3 АЛГОРИТМИ ПОБУДОВИ ДОПОВНЕНОЇ РЕАЛЬНОСТІ БЕЗ МАРКЕРІВ

Існує декілька підходів до побудови доповненої реальності, які відрізняються за способом отримання програмою інформації про положення в просторі.

За відсутності простих маркерів для побудови доповненої реальності програма може слідувати за будь-якими об'єктами, наприклад: фотографіями, обличчями, обкладинками журналу. Зазвичай для слідування за об'єктами використовуються наступні кроки: виділення ознак або ключових точок (англ. feature detection), виділення дескриптора на основі знайдених ознак, порівняння дескриптора з еталонним.

Не обов'язково повинен бути об'єкт – точка відліку, програма може отримати дані про положення у просторі на основі слідування за ознаками, які знайдені в навколишньому просторі. Прикладом такої системи є РТАМ (Parallel Tracking and Mapping), результат роботи якої зображений на рисунку 3.1 [9]. Окрім розпізнавання зображення з камери дані можуть братися з фізичних датчиків, наприклад: акселерометр, гіроскоп, датчик освітлення, вимірник відстані. Системи, в яких деяка інформація накладається на існуюче зображення, наприклад скло автомобіля або літака, але є нерухомою відносно нього, також може вважатися системою доповненої реальності.

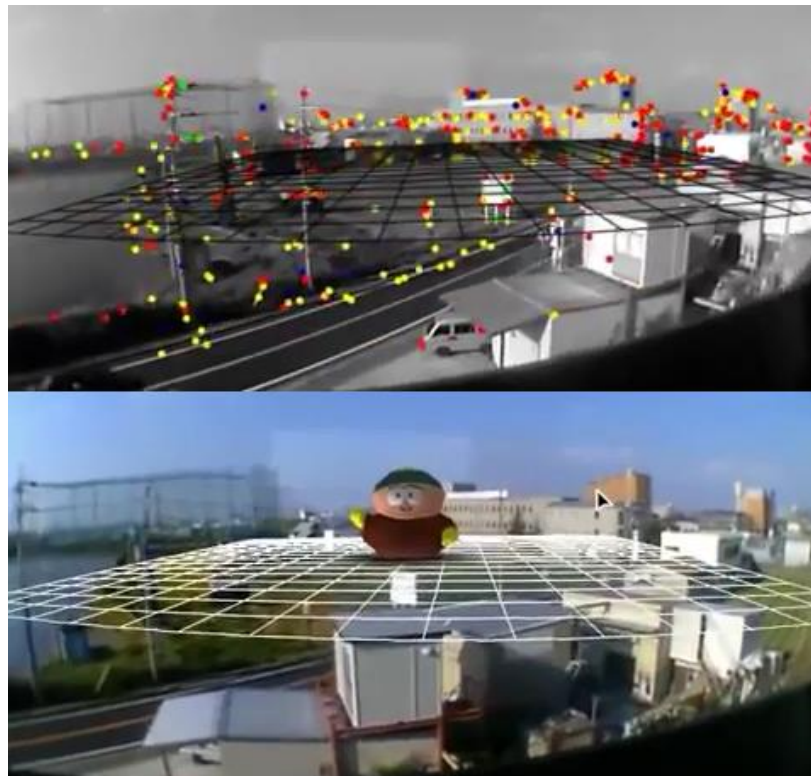


Рисунок 3.1 – Доповнена реальність на основі слідування за ознаками

### 3.1 Основна схема побудови доповненої реальності

Система доповненої реальності будується з двох частин:

- перша відповідає за обробку зображення з камери мобільного пристрою;
- друга відповідає за вивід мультиплікаційного персонажу.

До модуля обробки зображення спочатку завантажуються XML файли з даними машинного навчання та створюються на їх основі три класифікатори Віюлі-Джонса для пошуку очей, усмішки та обличчя. Кожен кадр отриманий з камери обробляється методами цього класу за алгоритмом представленим на рисунку 3.2. Інформація щодо властивостей знайдених облич – координати, наявність усмішки, кут нахилу, передається у текстовому вигляді до наступного модулю для анімації персонажу та виводу його на екран.

Друга частина відповідає за вивід мультиплікаційного персонажу на OpenGL площину поверх відеокadresу та контроль за його анімацією на основі даних, отриманих у результаті обробки відеокadresу (префікс класів G1).

Створюється OpenGL ES площина у вікні додатку поверх зображення з камери. Площині задається прозорий фоновий колір. Створюється об'єкт з бібліотеки Live2D на основі файлу моделі, до відеопам'яті завантажуються текстури з файлів, необхідні для відображення моделі.

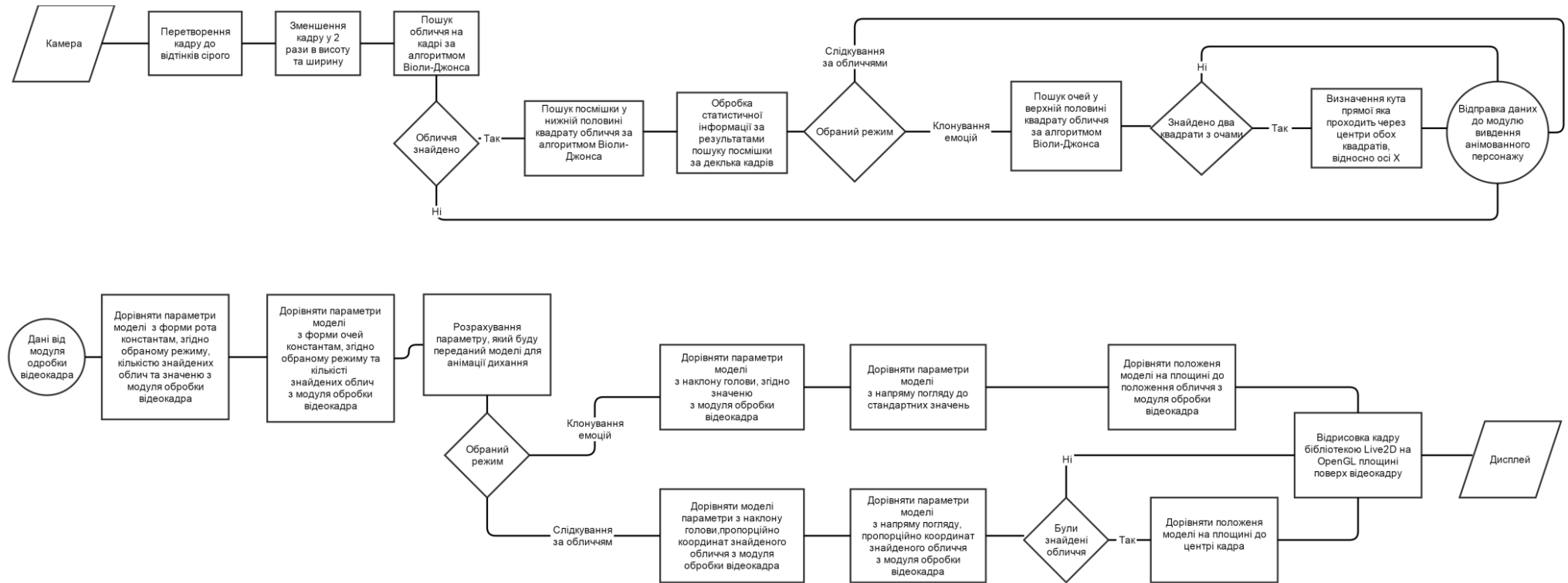


Рисунок 3.2 – Схема алгоритму створення доповненої реальності на основі розпізнавання обличчя

### 3.2 Метод Віюли-Джонса для розпізнавання облич

В 2001 році Віюла та Джонс запропонували алгоритм для розпізнавання облич, який став проривом в області розпізнавання облич і завдяки своїй швидкодії швидко став популярним. Базовою ідеєю при створенні алгоритму Віюли-Джонса для розпізнавання облич є виділення локальних особливостей (ознак) зображення і подальше навчання алгоритму на них. Кожна маска характеризується розміром світлої та темної областей, пропорціями, а також мінімальним розміром [10].

Для ефективного обчислення фільтрів Хаара використовується інтегральне представлення зображення. Таке представлення використовується в вейвлет-перетвореннях, SURF та багатьох інших методах. Якщо коротко, то інтегральне представлення є матрицею, розмірність якої збігається з розмірністю вихідного зображення, а елементи розраховуються за формулою:

$$I(x, y) = \sum_{i=0, j=0}^{i \leq x, j \leq y} I(i, j), \quad (3.1)$$

де  $I(i, j)$  – яскравість пікселів вихідного зображення.

Маючи інтегральну матрицю можна дуже швидко обчислювати суму яскравості пікселів довільних прямокутних областей зображення за формулою:

$$S(ABCD) = I(A) + I(C) - I(B) - I(D), \quad (3.2)$$

де ABCD – заданий прямокутник [11].

В цілому метод використовує технологію ковзаючого вікна. Так як об'єкт на зображенні може мати різний масштаб, необхідно провести пошук цього об'єкту з різним масштабом ознак. Рамка розміром меншим, ніж вхідне зображення, рухається з деяким кроком по зображенню та за допомогою каскаду слабких класифікаторів виявляє, чи є обличчя в розглянутому вікні. Метод ковзаючого вікна ефективно використовується в різноманітних задачах комп'ютерного зору та розпізнавання об'єктів.

Метод складається з двох частин: алгоритм навчання і алгоритм розпізнавання. На практиці швидкість алгоритму навчання не важлива. Вкрай важлива швидкість алгоритму розпізнавання. Метод можна віднести до структурних, статистичних і нейронних методів.

Віюла і Джонс адаптували ідею використання вейвлетів Хаара і розробили те, що було названо ознаками Хаара. Ознака Хаара складається з суміжних прямокутних

областей. Вони позиціонуються на зображенні, далі сумуються інтенсивності пікселів в областях, після чого обчислюється різниця між сумами. Ця різниця і буде значенням певної ознаки, певного розміру, певним образом спозиціонованої на зображенні.

При розпізнаванні чоловічих облич спільним для двох зображень звичайно є те, що область в області очей темніше за область в області щік. Тобто спільною ознакою Хаара для облич є 2 суміжних прямокутних регіони, що лежат на очах і щоках.

На етапі виявлення в методі Віюлі-Джонса вікно заданого розміру рухається по зображенню та для кожної області зображення, де воно проходить, розраховується ознака Хаара. Присутність або відсутність предмету у вікні визначається різницею між значенням ознаки та навчаним порогом. Оскільки ознаки Хаара мало підходять для навчання або класифікації, для опису об'єкта з достатньою точністю необхідна більша кількість ознак. Тому в методі Віюлі-Джонса ознаки Хаара організовані в каскадний класифікатор. Ключовою особливістю ознак Хаара є найбільша серед усіх ознак - швидкодія. Ознаки Хаара зображені на рисунку 3.3.

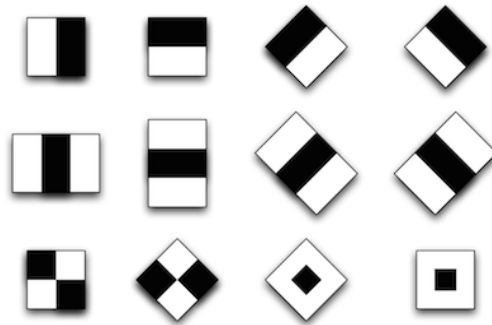


Рисунок 3.3 – Ознаки Хаара

Можливо використання алгоритму бустинга (англ. boosting) для обрання найбільш відповідних до шуканого об'єкту на даній частині зображення, процедури послідовної побудови композиції алгоритмів машинного навчання, коли кожен наступний алгоритм прагне компенсувати недоліки композиції всіх попередніх алгоритмів [12].

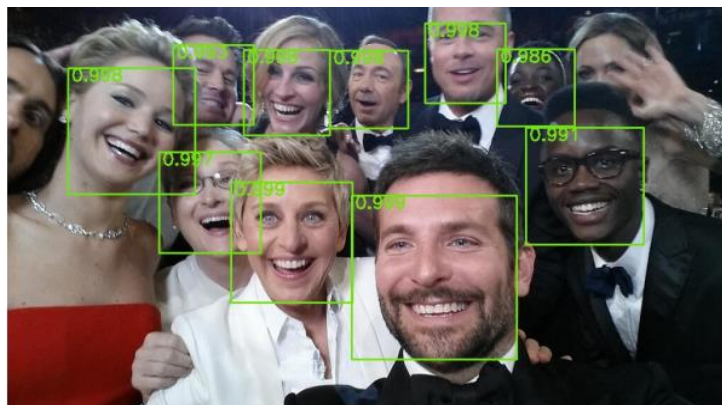


Рисунок 3.4 – Приклад розпізнавання облич за методом Віюлі-Джонса

#### 4 АЛГОРИТМИ ПОБУДОВИ ДОПОВНЕНОЇ РЕАЛЬНОСТІ НА ОСНОВІ РОЗПІЗНАВАННЯ МАРКЕРІВ

Маркери – реальні об’єкти, які поміщені до поля зору камери. Коли програма виявляє присутність одного або декількох маркерів, вони використовуються як точка відліку для побудови доповненої частини. Бажано, маркер повинен бути чорно-білим та мати просту геометричну форму – прямокутник або квадрат. При таких умовах можна виявити його за допомогою швидких алгоритмів пошуку кутів у чорно-білому режимі. Це зменшить час на його розпізнавання, що особливо важливо на мобільних пристроях, де обчислювальна потужність обмежена, а обробляти відеокадр потрібно багато раз на секунду.

QR-код (англ. quick response) може непогано виконувати роль маркеру (рис. 4.1). Засобами комп’ютерного зору легко виявити код та його орієнтацію завдяки несиметричності. Основний принцип QR-коду в тому, що він може працювати як гіперпосилання. Це особливо зручно, коли необхідно повідомити велику кількість інформації або спростити її використання.

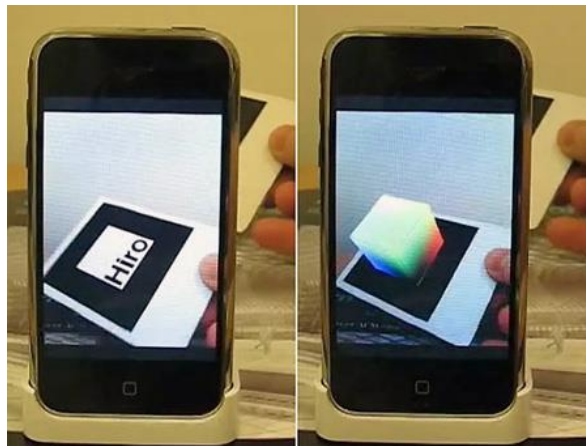


Рисунок 4.1 – Простіший приклад доповненої реальності на основі маркерів

QR-код картинки здатний полегшити процес вибору продукту або послуги, адже в невеликому квадратику на упаковці або рекламному плакаті може міститися до 4296 символів, що вміщують відомості, які незручно або дорого розміщувати в текстовому виді.

QR-код здатний оптимізувати зображення і комунікаційний процес. Наприклад, в значку на візитці часто зашифрований телефонний номер, який може бути доданий в список контактів одним натисненням кнопки. Також за допомогою QR-кодів можна



перейти по посиланню, відправити електронний лист, sms, упізнати координати місця і багато що іншого.

Іноді зустрічаються оригінальні варіанти використання двомірного коду – у Львові їх можна знайти на більшій частині місцевих пам'яток, а в Японії вони широко використовуються в меморіальних цілях – розміщені на надгробках японських кладовищ. Вони легко дозволяють будь-якому перехожому, у якого є скануючий пристрій, дізнатися інформацію про померлих.

#### 4.1 Основна схема побудови доповненої реальності

Перший етап обробки зображення з камери – пошук маркерів. Маркер має вигляд візитної картки розміром 90x50мм. Структура маркеру вказана на рисунку 4.2. Вона складається з квадрата з чорними контурами розміром 27x27мм, який розташований на відстані 6 мм від правого краю та відцентрований по вертикалі, всередині нього розташований QR-код. Між QR-кодом та контуром повинен бути відступ – це вимога стандарту QR-коду для надійного його визначення.

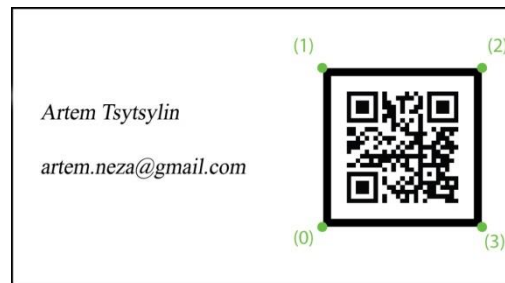


Рисунок 4.2 – Структура маркеру у вигляді візитної картки

Термін QR-code на даний момент є зареєстрованою торговою маркою DENSO Corporation. Використання кодів не обкладається жодними ліцензійними відрахуваннями, а самі вони описані і опубліковані в якості стандартів JIS-X-0510 та ISO / IEC 18004: 200. У даному стандарті описуються версії. В даному випадку поняття «версія» передбачає розмір символу (квадрата) QR-коду [13].

QR-код обраний частиною маркеру з наступних причин:

– він розроблений для швидкого визначення скануючим обладнанням в різних умовах, тому визначення трьох пошукових візерунків в нашому випадку також буде надійним;

– три пошукових візерунки дозволяють легко визначити орієнтацію маркера (рис. 4.3);

– QR-код – це перевірений роками стандарт кодування для машинного розпізнавання (в вільному доступі знаходяться алгоритми та бібліотеки для його читання);

– закодована інтернет-адреса може виконувати три функції відразу: зберігати ідентифікатор для відображення різної інформації у доповненій реальності з різними маркерами, відправляти користувача на сторінку встановлення мобільного додатка для перегляду доповненої реальності, або на сторінку з аналогічною інформацією у разі неможливості використання додатку.



Рисунок 4.3 – Приклад визначення орієнтації QR-коду за трьома точками

До QR-кода закодований текст посилання на мобільний додаток в інтернеті, скорочений сервісом «g.ua» до «<http://g.ua/kBPs>» з додаванням після посилання символу «#», який буде ігноруватися при використанні коду для переходу до веб-сторінки, та коду маркера, який буде прочитаний додатком.

Алгоритм для побудови доповненої реальності на основі маркерів є наступним: зображення приводиться з кольорового до відтінків сірого, далі для отримання бінарного зображення розраховується порогове значення за методом Оцу. На бінарному зображенні виконується пошук контурів за алгоритмом Suzuki85. Далі йде апроксимація контурів до простіших полігонів за алгоритмом Дугласа-Пекера, серед яких до подальшої обробки допускаються лише відповідні до наступних вимог: мати чотири кути, опуклу форму та площу більше деякої відносної величини.

Після перевірки, що чотирикутник не знаходиться в області, де вже був успішно знайдений маркер, виконується пошук трьох або більше вкладених чотирикутників, площа яких знаходиться у деяких межах відносно площі маркера цілком. Після визначення найбільш віддаленого від всіх трьох вкладених чотирикутників маємо

координати чотирьох кутів маркера в певній послідовності, яка вказана зеленим кольором на рисунку 4.2.

Для подальшого розпізнавання окремих маркерів, на основі визначеного чотирикутника, з кадру витягується фрагмент, з якого видаляються геометричні деформації. Фрагмент має вигляд бінарного зображення невеликого розміру для підвищення швидкодії. Якщо є збережені маркери, на яких QR-код вже був розпізнаний, фрагмент порівнюється із усіма з них. Якщо пара маркерів, які збігаються, знайдена, то маркеру на цьому кадрі присвоюється збережений код, згідно з яким буде малюватися доповнена реальність. Якщо збігів немає, фрагмент відправляється до детектору QR-кодів та при успішному розпізнанні його фрагмент буде збережений для прискорення подальших розпізнань.

Далі потрібно визначити положення камери у тривимірному просторі – вирішити проблему PnP (perspective-n-points problem), на основі яких будується матриця виду (англ. view matrix) для правильного подальшого відображення тривимірних об'єктів доповненої реальності. Тепер код маркеру та матриці виду відносно його можуть бути передані до модулю відображення 3D об'єктів.

Оптика будь-якої камери вносить викривлення до кадру та має різну фокусну відстань. Для отримання матриць перетворень маркеру, відповідних до реальності, має бути проведене калібрування камери – серія знімків візерунку під різними кутами з подальшою їх обробкою. Використовуємо візерунок з 44 чорних кіл на білому фоні, як на рисунку 4.4, який буде роздрукований на зворотній стороні картки. Отримані параметри зберігаються та потім використовуються у вирішенні задачі PnP.

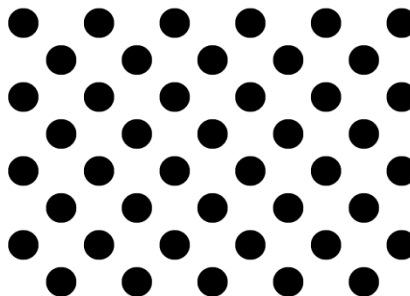


Рисунок 4.4 – Узор для калібрування

У модулі відображення 3D об'єктів створюється OpenGL ES площина у вікні додатку поверх зображення з камери. Площині задається прозорий фоновий колір. Після отримання повідомлення від модулю обробки зображення з матрицею виду та кодом маркеру виконується пошук даних візитної картки за кодом. Знайдені дані відправляються

до інтерфейсу програми у текстовому вигляді для можливості швидко відкрити їх у системі Android. Також за ними обчислюються та завантажуються до буфера вершини 3D-моделі, яка буде відображена. Малюється текстура з фоновим кольором, текстом та відступами, зазначеними у даних картки. Вона також завантажується до буфера. Якщо дані за кодом не знайдені, завантажуються стандартні вершини та текстура.

Перемножуюся матриці моделі (англ. *model matrix*), яка відповідає за трансформації об'єкту, який буде відображено, матриця виду (англ. *view matrix*), інформацію в якій можна представити як параметри положення та повороту камери, з вигляду через яку буде малюватися сцена. Ця матриця отримується з модулю обробки зображення. Остання матриця – матриця проєкції (англ. *projection matrix*) з параметрами щодо розміру екрану та мінімальної з максимальною відстаней малювання, інформацію з матриці проєкції можна представити як параметри зрізаної піраміди (англ. *viewing frustum*), зображеної на рисунку 4.5, об'єкти лише всередині якої будуть малюватися.

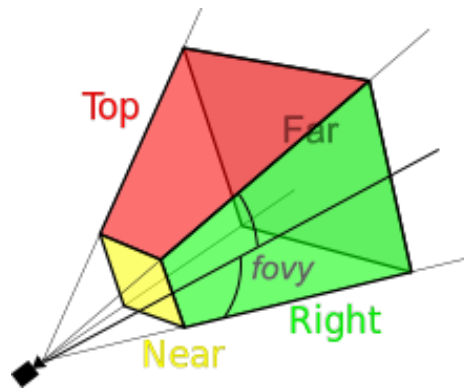


Рисунок 4.5 – Піраміда виду, яка пояснює матрицю проєкції

Отримана матриця, назва якої – MVP-матриця, що є аббревіатурою від назв трьох матриць, завантажується до OpenGL-підсистеми.

$$\begin{bmatrix} Right_x & Up_x & Look_x & 0 \\ Right_y & Up_y & Look_y & 0 \\ Right_z & Up_z & Look_z & 0 \\ -(Position \cdot Right) & -(Position \cdot Up) & -(Position \cdot Look) & 1 \end{bmatrix} \quad (4.1)$$

Після цього відображається чотирикутник із завантажених вершин з накладанням на нього завантаженої текстури, також на цьому етапі можуть бути відображені інші тривимірні об'єкти, які можуть створюватися індивідуально для маркерів з різними кодами. Якщо маркерів на поточному кадрі знайдено не буде, відображення ніяких тривимірних об'єктів не відбувається. Загалом алгоритм, від отримання відеокадру з камери до виводу тривимірних об'єктів, зображений на рисунку 4.6.

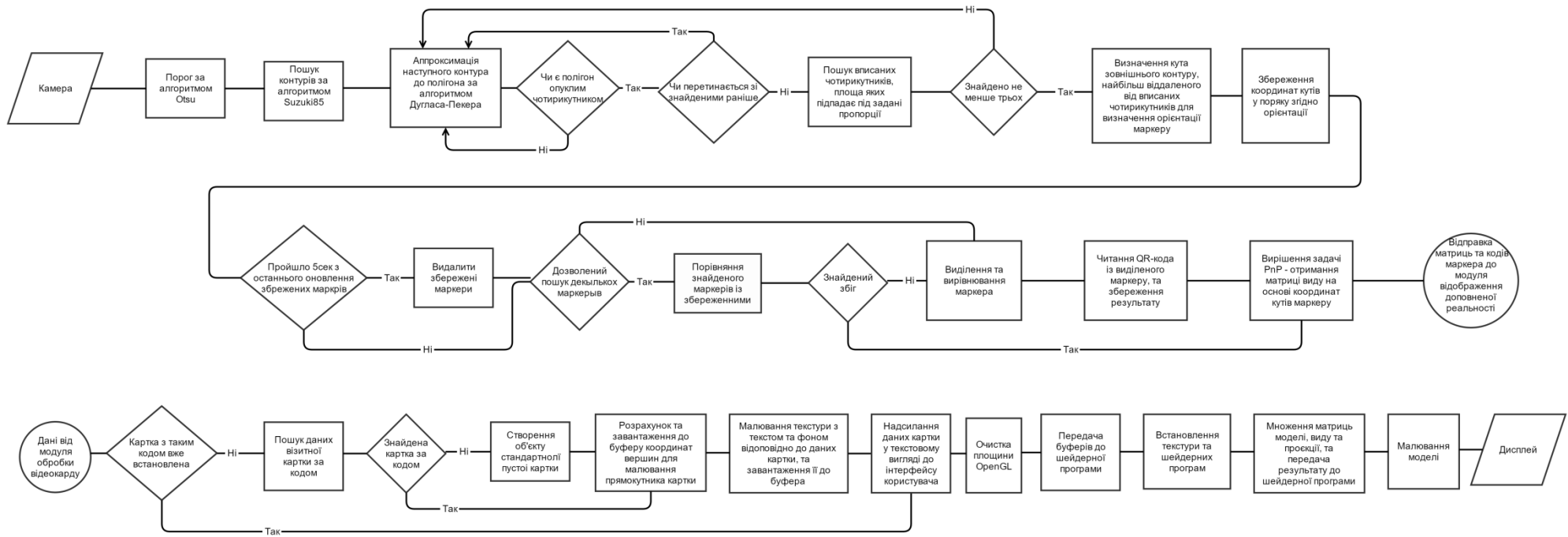


Рисунок 4.6 – Схема алгоритму створення доповненої реальності на основі розпізнавання маркерів

#### 4.1 Задіяні алгоритми

Метод Оцу – алгоритм обчислення порога бінарізації для напівтонового зображення, який використовується в області комп'ютерного розпізнавання образів та обробці зображень, винайдений Нобуюкі Оцу в 1979 році. За методом піксели розподіляються на два класи – корисні та фонові, розраховується такий поріг, щоб мінімізувати внутрікласову дисперсію, яка розраховується як зважена сума двох дисперсій за формулою

$$\sigma_{\omega}^2 = \omega_1(t)\sigma_1^2(t) + \omega_2(t)\sigma_2^2(t), \quad (4.2)$$

де ваги  $\omega_i$  — це вірогідності двох класів, розділених порогом  $t$ ;

$\sigma_i^2$  — дисперсія цих класів [14].

Алгоритм Suzuki85 – алгоритм топологічного структурного аналізу бінарного зображення та пошуку контурів, представлений Сатоши Сузукі та Кейічі Абе в 1985 році. Входить до пакету інструментів для розробки програм комп'ютерного зору OpenCV.

Алгоритм поділяє зображення на 1-пікселі (компоненти контуру), 0-пікселі (фон або отвір) та визначає відношення оточування контурів на бінарному зображенні. На виході отримуємо координати точок контуру у ланцюжковому коді Фрімена. Можлива апроксимація до вертикальних, горизонтальних та діагональних ліній із збереженням тільки їх кінцевих точок [15].

Алгоритм Дугласа-Пекера — алгоритм, дозволяючий знизити кількість точок кривої, апроксимованої великою серією точок. Був відкрит Урсом Рамером в 1972, Давидом Дугласом і Томасом Пекером в 1973. Відомий під назвами алгоритм ітеративної найближчої точки і алгоритм розбиття та злиття.

Суть алгоритму в тому, щоб по ломаній, апроксимуючій криву, побудувати ломану с меншою кількістю точок. Алгоритм виявляє розбіжність, яка обчислюється по максимальній відстані між вхідною та спрощеною кривими. Спрощена крива складається з підмножини точок, які визначаються з вхідної кривої.

#### 4.1.1 Алгоритм читання QR-коду

QR-код представляє собою двовимірний матричний код, який розроблений для збереження текстової інформації та швидкого машинного перетворення. Представляє собою матрицю з чорних квадратів на білому фоні та складається з структурних елементів, позначених на рисунку 4.7.

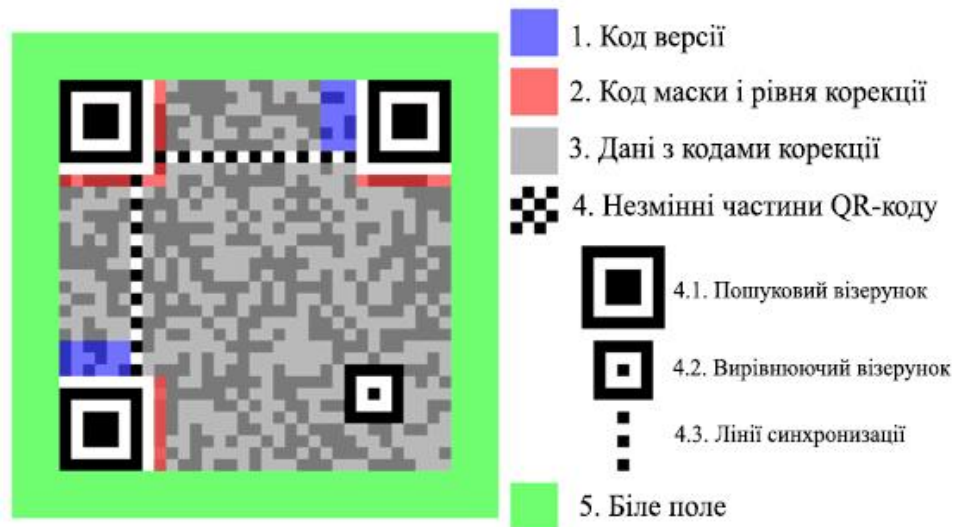


Рисунок 4.7 – Структура QR-коду

Дані, які необхідно закодувати, розбиваються на блоки. До розбитих блоками даних додається заголовок, який вказує режим і кількість блоків. На випадок некоректного читання даних застосовуються спеціальні коди, які здатні виправити недоліки при читанні. Це так звані коди Ріда-Соломона (RS) записуються після всіх інформаційних даних [16].

Дані з RS-кодами перед записом у QR-код «перемішуються». Для цих цілей використовуються маски. Існує 8 алгоритмів, серед яких вибирається найкращий. Кодувальник являє собою сукупність зсувних регістрів (shift registers), які об'єднані за допомогою суматорів і множників. Дані операції проводяться на основі арифметики Галуа. Зсувний регістр являє собою послідовність елементів пам'яті, званих розрядами, кожен з яких містить один елемент поля Галуа  $GF(q)$ . Він міститься в розряді символів та, покидаючи цей розряд, «вистрілюється» на вихідну лінію. Одночасно з цим розряд «засмоктує» символ, що знаходиться на вході. Заміщення символів відбувається дискретно в певні проміжки часу, звані тактами [17].

Читання складається з наступних кроків:

- знаходження пошукових зорів;
- визначення версії коду;
- пошук зорів для вирівнювання;
- читання системної інформації щодо маски даних та рівня корекції помилок;
- читання заголовку типу даних;
- застосування маски до заголовку;
- читання даних;
- застосування маски до даних.

#### 4.1.2 Вирішення проблеми PnP

Об'єкт має деякі тривимірні точки у системі координат, також відомі координати точок об'єкту на зображенні, та відомо, яка точка на зображенні відповідає певній тривимірній точці. Потрібно визначити положення об'єкту – знайти матриці R та T, які характеризують поворот та переміщення відповідно.

Цю задачу можна розглянути як задачу оптимізації. Маємо тривимірні точки та знаємо, що для проекційної матриці P точки  $(X, Y, Z)$  проєцуються у деякі  $(u, v)$  при деякому фіксованому положенні. Потрібно взяти такі R і T, щоб проєкції співпадали з тими координатами, які спостерігаємо насправді. Таким чином отримуємо задачу оптимізації – необхідно знайти мінімум функції помилки:

$$error(P) = \sum_i \left| \begin{pmatrix} u_i \\ v_i \end{pmatrix} - \begin{pmatrix} u_i^p \\ v_i^p \end{pmatrix} \right|, \quad (4.3)$$

варіюючи R і T, при умовах

$$w_i \begin{pmatrix} u_i^p \\ v_i^p \\ 1 \end{pmatrix} = P \begin{pmatrix} x_i \\ y_i \\ x_i \end{pmatrix}. \quad (4.4)$$

Зазвичай використовується метод оптимізації Левенберга-Марквардта, який комбінує методи найшвидшого спуску та метод Ньютонa [18].



### 4.1.3 Калібрування камери

Потрібно визначити внутрішні параметри камери – коефіцієнти дисторсії та масштабуючі коефіцієнти  $f_x$ ,  $f_y$  з коефіцієнтами зсуву  $c_x$ ,  $c_y$ , які утворюють матрицю  $K$ . Їх можна визначити, якщо декілька раз під різними кутами показати програмі об'єкт, параметри якого відомі. Зазвичай при калібруванні використовують шахівницю, де відомо, що між усіма кутами клітинок однакові відстані або інші дошки з колами (рис. 4.4) або прямокутниками з однаковими відстанями між ними.

## 4.2 Побудова доповненої реальності

Для вирішення цього завдання скористаємося описаною вище бібліотекою OpenCV. З її допомогою здійснюватиметься пошук маркера на відеопотоці з подальшим відображенням віртуального об'єкту. Безпосередньо сам процес пошуку описувати не будемо, оскільки для цього використовуються алгоритми feature point detection, що описаний вище.

Передусім, важливо питання побудови 3d просторів по знайдений 2d гомографії. Під гомографією розуміється матриця побудови взаємно-однозначного відображення проектного простору на себе, що переводить точки в точки, прямі в прямі та зберігає відношення інцидентності точок і прямих, а також подвійне відношення будь-якої четвірки колінеарних точок. Вона дозволяє приводити зображення до єдиної перспективи і геометрії.

Для побудови 3d просторів необхідно знати дві матриці: внутрішню (intrinsic matrix) і зовнішню (extrinsic matrix). Потім, скориставшись методами OpenGL, можна без яких-небудь проблем намалювати віртуальний об'єкт поверх маркера.

Внутрішня матриця (чи матриця проєкції) складається з параметрів використовуваної камери:

$$\begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}, \quad (4.5)$$

де  $f_x$ ,  $f_y$  – фокальна відстань по двох осях;

$c_x$ ,  $c_y$  – координати центру фокусу.

Процес знаходження параметрів камери називається її калібрацією. Для її проведення необхідно зробити знімки калібрувального шаблону (шахівниці) за допомогою використовуваної камери, перенести фотографії на комп'ютер і зробити підрахунок параметрів матриці.

З метою отримання як можна точніших параметрів необхідне дотримання наступних умов:

- шаблон шахівниці роздруковується на чистому аркуші формату А4;
- лист шаблону повинен лежати рівно на поверхні, краї не мають бути загнутими, бажана відсутність вигинів;
- розмір знімків з камери має бути приведений до розміру кадру відеопотоку. В деяких випадках можлива ситуація, коли роздільна здатність фотографій вища за роздільну здатність відео. Тому, перед калібруванням необхідно зменшити знімки до потрібного розміру;
- кількість знімків шаблону має бути не менше 10. Виконати їх треба з різних ракурсів. Чим більше буде знімків, тим точніші будуть отримані параметри матриці проєкції, а ця точність далі впливатиме на наявність/відсутність зрушень при побудові 3d об'єктів.

Зовнішня матриця (чи матриця моделі) – це матриця перетворень моделі за допомогою розтягування, повороту і перенесення. Вона дозволяє однозначно задати положення об'єкту в просторі.

$$\begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix}, \quad (4.6)$$

де діагональні елементи відповідають за розтягування моделі;

інші елементи  $r$  – за поворот об'єкту в просторі;

елементи  $t$  відповідають за перенесення.

В цілому, структура цієї матриці може варіюватися залежно від рівнянь перетворень координат, але описані ключові елементи в ній присутні завжди.

Обчислення робиться шляхом визначення чотирьох пар точок об'єкту і його відповідного положення в кадрі. Точки положення в кадрі – це вершини чотирикутника, що описує (обмежує) об'єкт в кадрі. Отримати ці точки можна, якщо подіяти на крайні точки шаблону гомографією.

Відносно точок самого об'єкту варто відмітити декілька моментів:

– точки об'єкту задаються в  $3d$ , а точки на кадрі в  $2d$ . Відповідно, якщо задати їм ненульове значення  $z$ , то початок координат на  $z$  буде зрушений відносно площини об'єкту на кадрі;

– точки об'єкту задаються так, щоб було зручно працювати в  $3d$  просторі, а саме: початок координат знаходиться у центрі шаблону, а одиниця довжини дорівнює половині меншої сторони. В цьому випадку виключаємо залежність від конкретних розмірів шаблону в пікселях, а  $3d$  простір буде відмасштабовано по меншій стороні.



Рисунок 4.8 – Розміщення об'єкту на маркері з напрямками осі координат

Сконструйовані матриці потім використовуються для обчислення вектору повороту і перенесення з подальшим занесенням в матрицю моделі. Варто відмітити, що процес обчислення матриці моделі займає декілька мілісекунд і істотно не впливає на швидкість роботи програми. Основне навантаження йде в області аналізу відеопотоку і пошуку на ньому шаблонного зображення.

Описаний вище підхід можна застосувати до усіх бібліотек комп'ютерного зору в плані побудови доповненої реальності. З практичної точки зору він виглядає дещо складним, зважаючи на необхідність проведення калібрування камери для отримання коректної матриці проектування.

Одним із способів вирішення цієї проблеми є створення набору параметрів різних камер, а залежно від використовуваної камери робитиметься завантаження необхідних параметрів.

Розвиваючи цей напрям, можна цілком побудувати свій власний фреймворк доповненої реальності. Розробляючи різні способи фільтрації точок, можна добитися того,

що маркер знаходитиметься навіть при сильному куті нахилу і великому віддаленні. Це дозволить отримати реалістичнішу картинку і дасть можливість користувачам розглянути модель з усіх боків на різній відстані.

## 5 ПРОГРАМНА РЕАЛІЗАЦІЯ ДОДАТКУ ДОПОВНЕНОЇ РЕАЛЬНОСТІ

Для розробки мобільного додатку була обрана платформа Android, як одна з найпоширеніших, також тому, що вона не потребує особливого обладнання для розробки та дешевше для публікації.

Середою обробки обрано Eclipse з плагінами для розробки під Android.

### 5.1 Загальний дизайн системи

Система поділяється на дві частини: перша – для обробки зображення з камери, складається з двох класів CvEngineBase та CvProcessorBase для більш простішої реалізації багатопоточності та інших модифікацій в майбутньому; друга – для виводу тривимірних об'єктів на екран. Поділ її на два класи є необхідністю, тому що OpenGL в системі Android потребує створення окремого класу для рендеру (GIRendererBase). Обмін даними між двома частинами та їх ініціалізацію забезпечує клас AREngineBase.

Клас CvEngineBase відповідає за отримання зображення з камери, підготовку його до обробки та передачі результатів обробки до інших модулів. При його ініціалізації відбувається завантаження бібліотеки OpenCV, та він реалізує інтерфейс CvCameraViewListener2, тому отримує кожен новий кадр до методу OnCameraFrame. Спочатку обробки кадру він перегортається та віддзеркалюється у методі getMatFromFrameWithRotation відповідно до орієнтації пристрою в просторі та вимогам, переданим у параметрах. Після цієї підготовки зображення передається до класу CvProcessorBase методом pushFrame та отримується методом pullFrame для виводу на екран. При обробці в одному потоці зображення буде отримане лише після завершення обробки, при асинхронній обробці це може відбутися раніше. Підраховується середня кількість кадрів на секунду, яку можна отримати методом getFPS.

Метод pushFrame класу CvProcessorBase збережує зображення до змінної frame, яку можна отримати методом pullFrame. Також для отримання результатів обробки окрім вихідного зображення, клас має змінну stringData та метод pullStringData для доступу до неї.

Клас GLEngineBase відповідає за ініціалізацію OpenGL отримання та підготовку даних для відображення на OpenGL площині. При його ініціалізації OpenGL площині

назначаються основні параметри та екземпляр класу `GLRenderBase`, до методів якого буде звертатися система для малювання сцени.

Клас `AREngineBase` є оболонкою для усієї системи створення доповненої реальності, дозволяє ініціалізувати її та передавати сигнали для контролю роботи всіх компонентів системи.

Щоб мати можливість повторного використання системи в інших додатках, ці базові класи повинні бути незмінними, для реалізації програми необхідно перевантажити їх усі. Діаграма класів зображена на рисунку 5.1. До проекту підключена бібліотека `OpenCV`.

На головному екрані додатку в повноекранному режимі (клас `MainActivity`) створюються наступні об'єкти: вікно перегляду зображення з камери `CameraBridgeViewBase`, поверхня `GLSurfaceView`, яка перекриває попереднє вікно. Обидва вікна передаються до об'єкту класу `AREngineBase` при його ініціалізації, які будуть використані в класах `CvEngineBase` та `GLEngineBase`. Також створюється текстове поле, яке розташоване поверх обох попередніх вікон. До нього буде асинхронно друкуватися інформація щодо швидкодії додатку, отримана з методу `getFPSString` класу `AREngineBase`. При натисканні на будь-яку точку екрану викликається меню (рис. 5.2), при виборі пунктів якого будуть передані відповідні текстові повідомлення через метод `sendMessage` класу `AREngine`. Меню дозволяє виконати наступні дії: викликати вікно з інформацією про додаток, вимкнути показ змін, внесених до відео під час його обробки, заборонити `OpenGL`-малювання поверх відео, сховати інформацію про швидкодію, замінити відеокадр на чорний, змінити режим віддзеркалення кадру, обрати одну з камер пристрою для використання.

Система передачі повідомлень між модулями системи працює наступним чином: до методу `sendMessage` передається ідентифікатор модулю-отримувача з перерахування `MSG_ENUM`, яке має три значення – `TO_CV`, `TO_GL` та `TO_VIEW`, код повідомлення та його тіло. Залежно від ідентифікатора з `AREngineBase` викликається метод `onMessage` відповідному класі з такими самими кодом та тілом повідомлення.

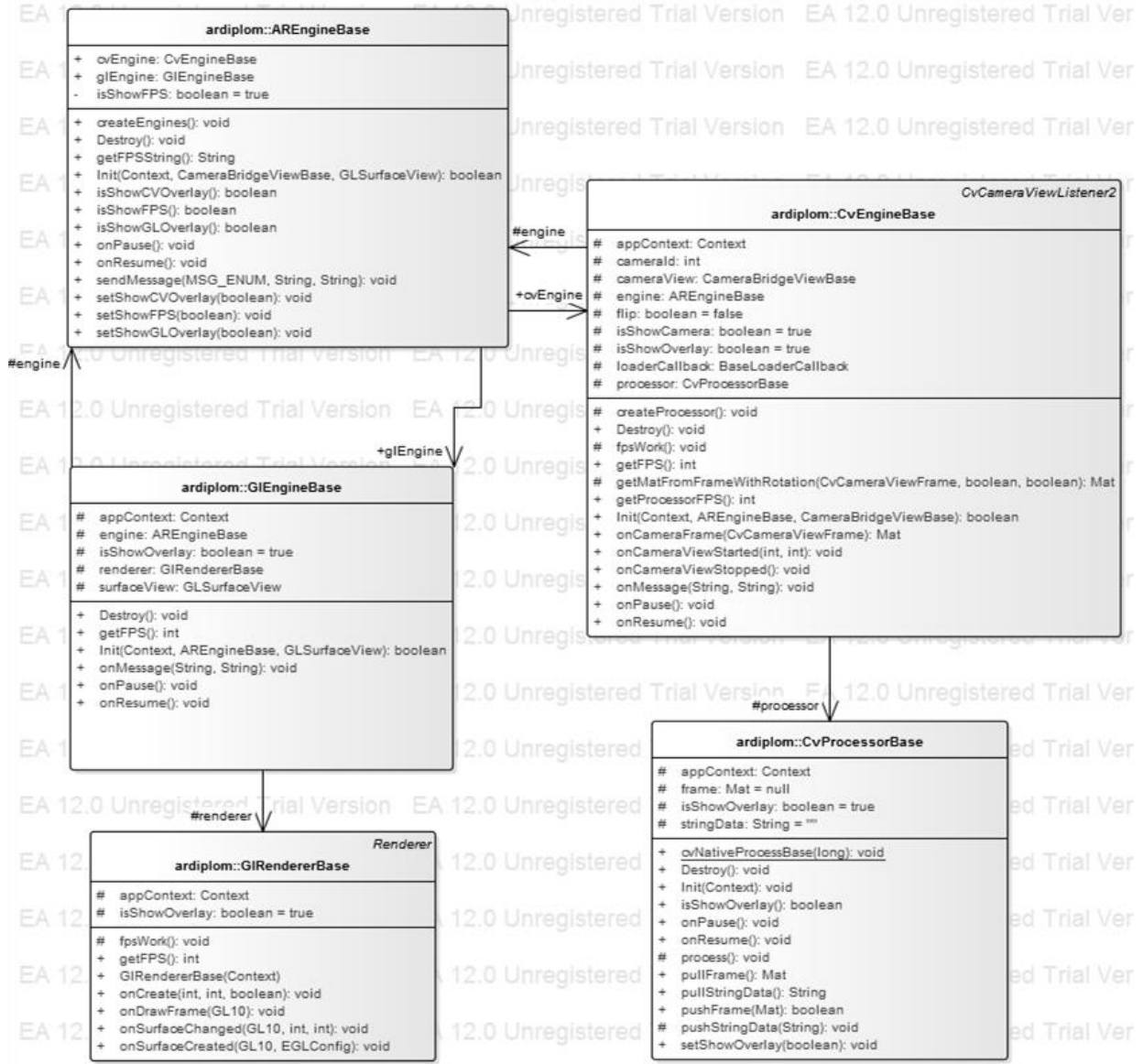


Рисунок 5.1 – Загальний дизайн системи

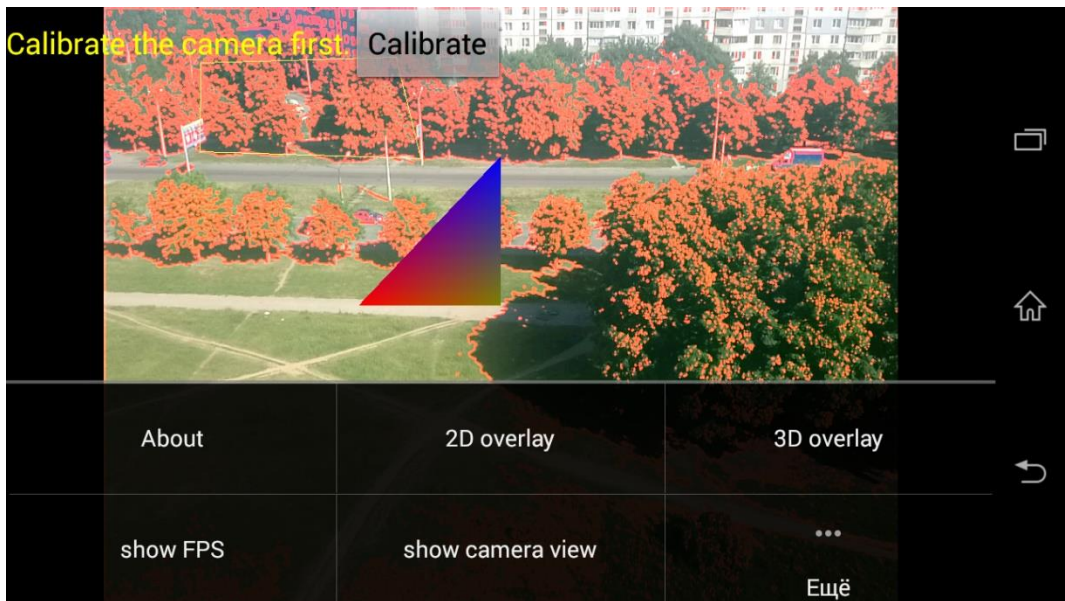


Рисунок 5.2 – Інтерфейс користувача

## 5.2 Застосовані технології та бібліотеки

Інтегрована середовище розробки Eclipse – вільне модульне інтегроване середовище розробки програмного забезпечення. Включає набір інструментів для розробників на мові Java, засоби для управління сирцевими кодами, візуальні побудовники GUI тощо. Може бути використаний для розробки додатків на Java та за допомогою різних плагінів на багатьох інших мовах програмування, включаючи Ada, C, C++, COBOL, Fortran, Perl, PHP, Python, R, Ruby (включно з каркасом Ruby on Rails), Scala, Clojure та Scheme. Середовища розробки зокрема включають ADT (Android Development Toolkit) для платформи Android на Java, Eclipse CDT для C/C++, також підтримується Android NDK (Native Development Kit), який дозволяє писати окремі частини коду на мові C/C++ та виконувати його поза віртуальною машиною Java, з'єднуючи їх за механізмом JNI (Java Native Interface).

OpenCV — бібліотека функцій та алгоритмів комп'ютерного зору, обробки зображень і чисельних алгоритмів загального призначення з відкритим кодом. Реалізована на C/C++, але має інтерфейси для Python, Java, Matlab та інших мов. Бібліотека багатоплатформна, підтримуються такі платформи, як: Windows, MacOS, iOS, Android, Linux. Аналіз і обробка зображень, розпізнавання жестів, облич і об'єктів, ідентифікація, відстеження об'єктів, побудова їх тривимірної моделі – є сферами застосування бібліотеки.

OpenGL ES — підмножина графічного інтерфейсу OpenGL, розроблена спеціально для вбудованих систем — мобільних телефонів, кишенькових комп'ютерів, гральних консолей та є основним графічним інтерфейсом у системі Android. В свою чергу OpenGL – специфікація до графічних бібліотек, що визначає незалежний від мови програмування багатоплатформний програмний інтерфейс (API) для написання додатків, що використовують 2D та 3D комп'ютерну графіку. Цей інтерфейс містить понад 250 функцій, які можуть використовуватися для малювання складних тривимірних сцен з простих примітивів. Широко застосовується індустрією комп'ютерних ігор і віртуальної реальності, у графічних інтерфейсах, при візуалізації наукових даних та в системах автоматизованого проектування.

Live2D – розроблений японською компанією Cybernoids набір програмних інструментів для анімації двовимірних персонажів в реальному часі, також засоби для виводу їх на екран за допомогою OpenGL на різних платформах. Разом з цими інструментами, представлений набір готових моделей персонажів, безкоштовних для використання в некомерційних цілях.



### 5.3 Деталі реалізації додатку для побудови доповненої реальності без маркерів

До меню додана можливість зміни порогового значення для детектора усмішки, вибір режиму спостереження за обличчями або копіювання емоцій.

Клас `AREngineAnime`, нащадок `AREngineBase` створює два об'єкти класів `CvEngineLive2d` (дод. А) та `GLEngineLive2d` (дод. Б). Діаграма класів зображена на рисунку 5.3.

Додатково до проекту включена бібліотека Live2D та файли моделі і текстур для відображення анімованного персонажу, взяті з офіційного ресурсу розробників бібліотеки Live2D, де представлені декілька моделей-зразків з усіма необхідними ресурсами для розробників та їх ліцензії, прийнятні для цього проекту [19]. Також додані до проекту XML-файли з даними навчання класифікатора Віюлі-Джонса, які готові для пошуку облич, очей і усмішки, та поставляються разом з бібліотекою OpenCV.

Клас `CvEngineLive2d` розширює `CvEngineBase` з метою обробки нових типів повідомлень, створення об'єкта-обробника зображення типу `CvProcessorFaceDetection` та пересилання повідомлення до модуля відображення після обробки кожного кадру від камери з даними по знайденим на зображенні об'єктам.

Обробка зображення в класі `CvProcessorFaceDetection` відбувається за алгоритмом, схема якого зображена на рисунку 3.2. Найбільш вимоглива до обчислювальної потужності частина алгоритму – саме пошук об'єктів на зображенні, створення і знищення об'єктів – класифікаторів Віюлі-Джонса для цього – написані на мові C++ та виконуються у нативному кодї, тобто поза віртуальною машиною Java. Виклик усіх нативних функцій йде з класу `CvProcessorFaceDetection`, нативний код зв'язаний з Java-кодом за допомогою технології JNI.

У класі `GLEngineLive2D` відбувається обчислення значень на основі отриманих після обробки зображення даних за алгоритмом, схема якого зображена на рисунку 5.3, та з використанням набору констант. Ці значення будуть передані через екземпляр `GLRendererLive2D` до моделі Live2D-персонажу та вплинуть на наступні його риси: положення на екрані, кути нахилу голови, напрямок погляду та міміку очей і рота.

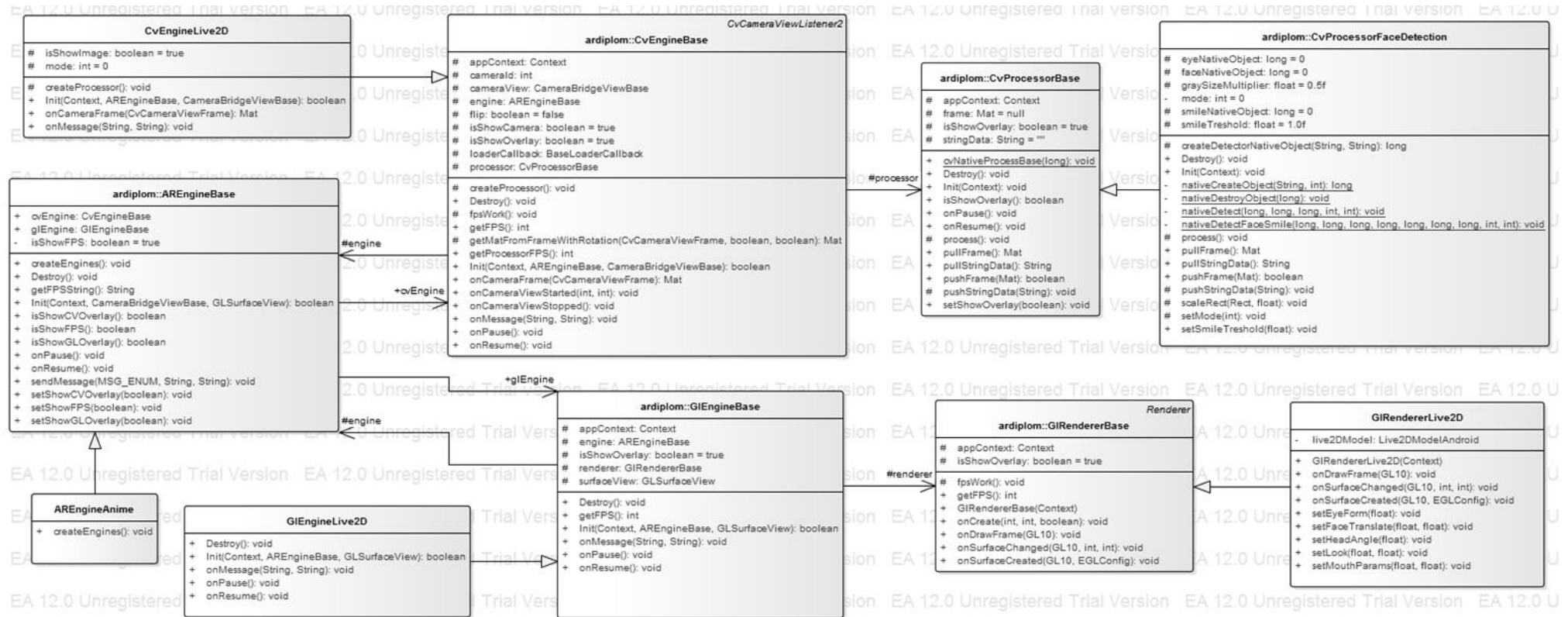


Рисунок 5.3 – Схема класів додатку для побудови доповненої реальності на основі розпізнавання обличчя

У класі `GIRendererLive2D` відбувається завантаження моделі та текстур персонажу з файлів. При виводі кожного кадру на екран розраховуються значення параметрів, які повинні змінюватися плавно, або в залежності від часу – для більш приємного вигляду анімації. Виконуються установка потрібних матриць до OpenGL, параметри встановлюються моделі персонажу, та їй передається OpenGL-площина для його малювання.

На рисунку 5.4 зеленим прямокутником означена область, де алгоритмом було виявлено обличчя.



Рисунок 5.4 – Розпізнавання обличчя та додаток доповненої реальності на його основі у режимі слідкування за обличчям

На рисунку 5.5 вимкнено відображення OpenGL частини, тому також можна бачити область обличчя та області, де були знайдені інші типи об'єктів – усмішка (блакитний прямокутник) та двоє очей (червоні прямокутники). Однак на практиці всі ці об'єкти одночасно виділяються лише в умовах доброго освітлення та в деякому діапазоні відстані від камери, можливо алгоритм був погано навчений або метод неефективний для цього типу об'єктів.

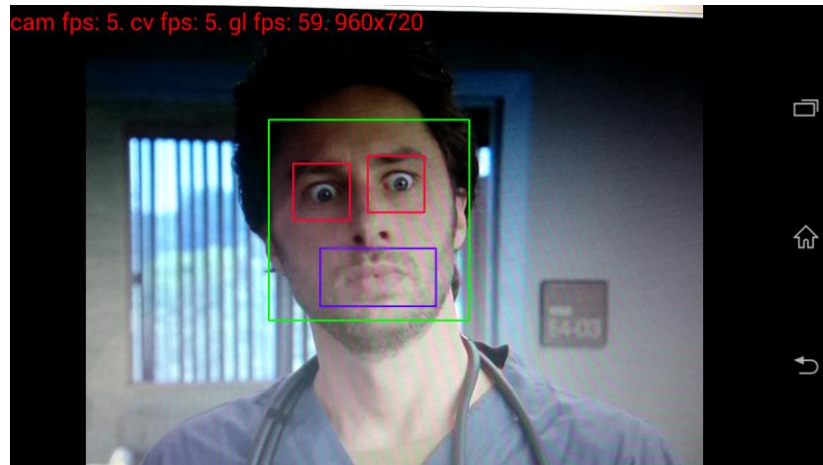


Рисунок 5.5 – Розпізнавання об'єктів трьох типів за методом Віоли-Джонса

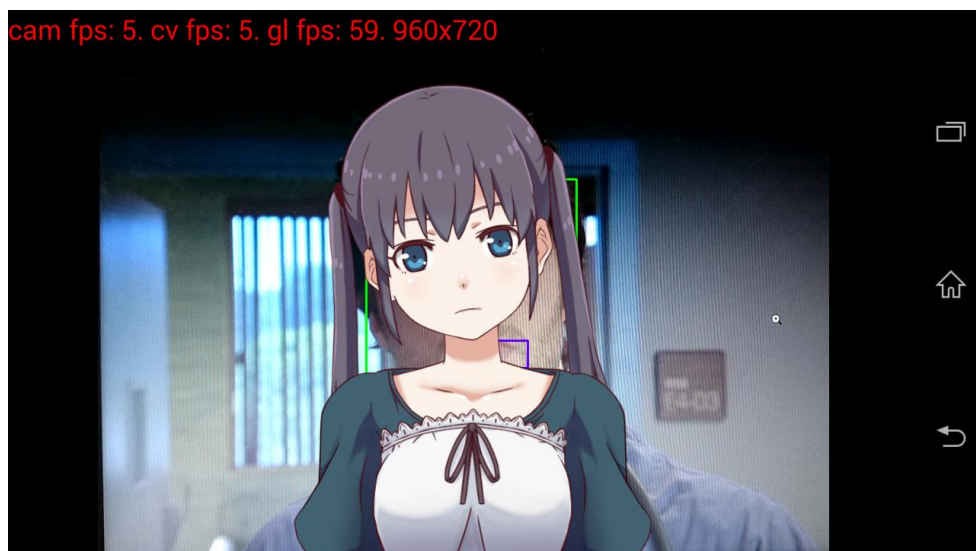


Рисунок 5.6 – Додаток доповненої реальності у режимі копіювання міміки на основі розпізнаного обличчя та його рис

#### 5.4 Деталі реалізації додатку для побудови доповненої реальності на основі розпізнавання маркерів

До меню (рис. 5.2) додана кнопка для переходу у режим калібрування камери та можливість очистити буфер відсканованих маркерів, які знаходяться у пам'яті для оптимізації їх повторного читання на випадок помилкового їх розкодування.

Якщо не знайдено збережених даних калібрування камери, на верхній частині екрана виводиться повідомлення про таку необхідність та кнопка для переходу у режим калібрування (рис. 5.2).

До проекту доданий код з офіційного додатку-прикладу до бібліотеки OpenCV (пакет `org.opencv.samples.cameracalibration`). Цей код відповідає за екран калібрування камери стандартними засобами OpenCV. Модифікований лише спосіб збереження даних калібрування для повторного використання та до інтерфейсу користувача додані кнопки, необхідні в повноекранному додатку (рис. 5.7).

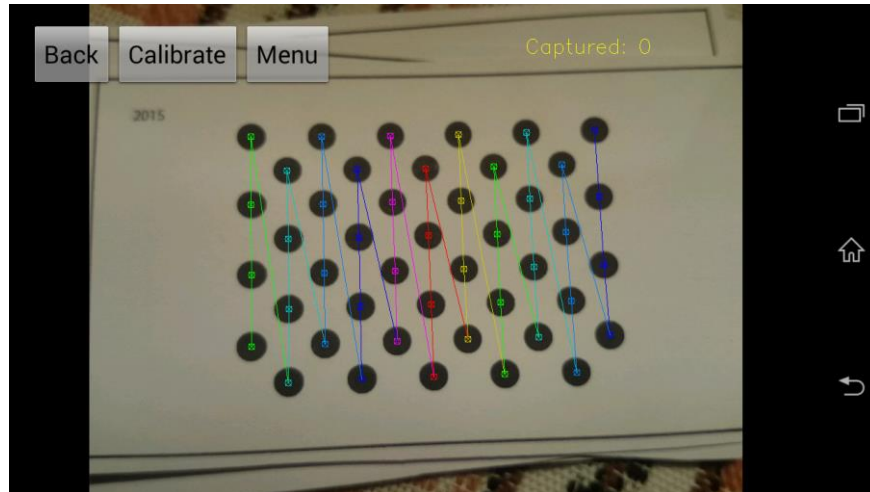


Рисунок 5.7 – Екран калібрування камери

Клас `AREngineVCard` нащадок `AREngineBase` створює два об'єкти класів `CvEngineVCard` та `GlEngineVCard`. Діаграма класів зображена на рисунку 5.8. Префіксом класів цього проекту обрано `VCard` від англійського `business card`, що перекладається як візитна картка.

Додатково до проекту включена бібліотека під назвою `qrcode`, яка буде використана для читання QR-кодів з маркерів. Вона написана на мові Java, має відкритий код та розповсюджується під ліцензією GPL v2 [20].

Клас `CvEngineVCard` розширює `CvEngineBase` з метою створення об'єкта – обробника зображення типу `CvProcessorVCard`, передачі йому команд та пересилання повідомлень до модуля відображення після обробки кожного кадру від камери з кодом маркеру та відповідною йому матриці.

Обробка зображення в класі `CvProcessorVCard` відбувається за алгоритмом, схема якого зображена на рисунку 4.6. Найбільш вимогливі до обчислювальної потужності частини алгоритму – пошук маркерів на зображенні, вирівнювання його перед розпізнаванням QR-коду та обчислення матриці на основі координат кутів маркеру – написані на мові C++ та виконуються у нативному коді, тобто поза віртуальною машиною Java. Виклик усіх нативних функцій йде з класу `CvProcessorVCard`, нативний код зв'язаний з Java-кодом за допомогою технології JNI. Також в цьому класі створюється

об'єкт типу `QRCodeDecoder` з бібліотеки `qrcode`, до якого буде надсилатися зображення маркеру після вирівнювання. З отриманого тексту з QR-коду виділяється число після знаку «#», що є кодом візитної картки. Читання QR-коду відбувається не з кожним кадром, а коли масив раніше прочитаних кодів, що очищається через деякий проміжок часу або за допомогою кнопки в меню, є порожнім.



До екземпляру класу `GLEngineVCard` потрапляє повідомлення із значеннями матриці виду та код маркеру від модуля обробки зображення. Виконується пошук даних картки за кодом, в результаті чого об'єкт класу `GLCard` з даними або пустий, якщо за кодом нічого не знайдено, передається до екземпляру класу `GLRendererVCard`. З нього же отримується матриця виду, до якої вносяться значення з повідомлення.

Клас `GLEngineVCard` є нащадком `GLRendererBase`, який в свою чергу реалізує протокол `Renderer`, що використовується при малюванні `OpenGL`. В ньому створюються буфери для координат вершин об'єкту в тривимірному просторі, для текстур та для значень, які поєднують координати на текстурі з вершинами. Компілюються шейдерні програми, які необхідні в `OpenGL ES` версії 2.0. При отриманні об'єкту картки від `GLEngineVCard`, якщо він відрізняється від поточного за кодом, оновлюються значення в буфері вершин на основі розміру, співвідношення сторін та положення відносно маркера картки, які можуть бути назначені індивідуально для карток з різними кодами. Створюється нова текстура, на якій друкуються текстові дані з об'єкта картки. Колір фону та інші стилі також можуть бути індивідуальні. Далі з кожним кадром проводяться необхідні операції з матрицями, та вони разом з текстурою завантажуються до шейдерної програми на відеоадаптер, та відбувається малювання кадру.

Результат роботи додатку можна бачити на рисунку 5.9, код маркеру «0». На рисунку 5.10 – результат для маркеру з кодом «1» під іншим кутом. На жаль, є проблема з відтворенням геометричних перетворень, які б відповідали реальності. Помилка виникає на етапі отримання векторів перетворень на основі чотирьох кутів маркеру або при створенні матриці виду з них.

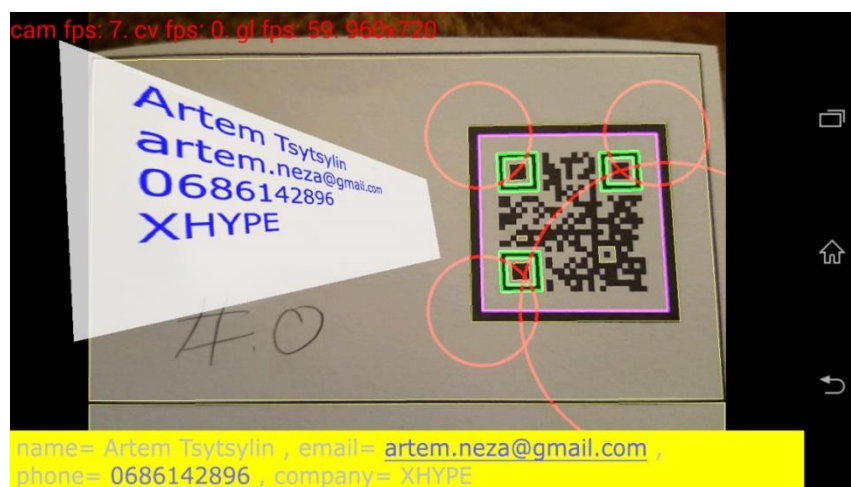


Рисунок 5.9 – Результат роботи додатку



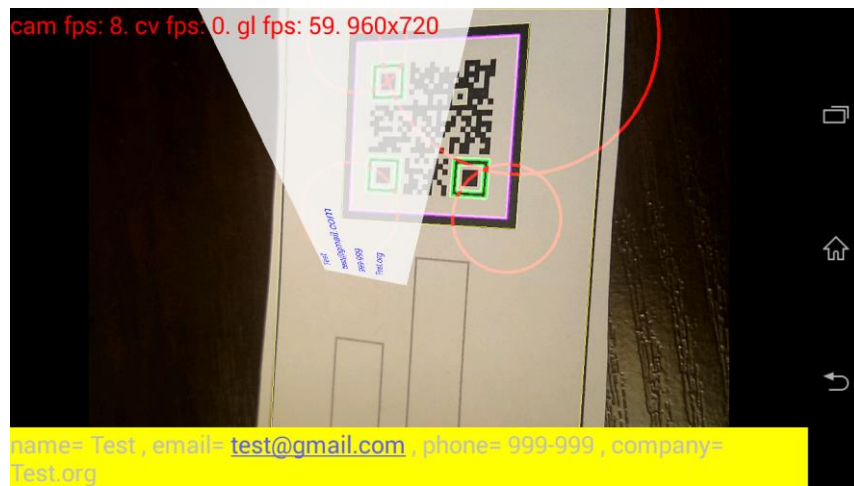


Рисунок 5.10 – Результат роботи додатку з іншим маркером під іншим кутом

В нижній частині вікна на жовтому фоні відображається текстова інформація з картки, яку можна використати одним натисканням, якщо це телефонний номер, електронна пошта або адреса веб-сторінки. Система Android запропонує декілька варіантів відкриття посилань, як на рисунку 5.11.

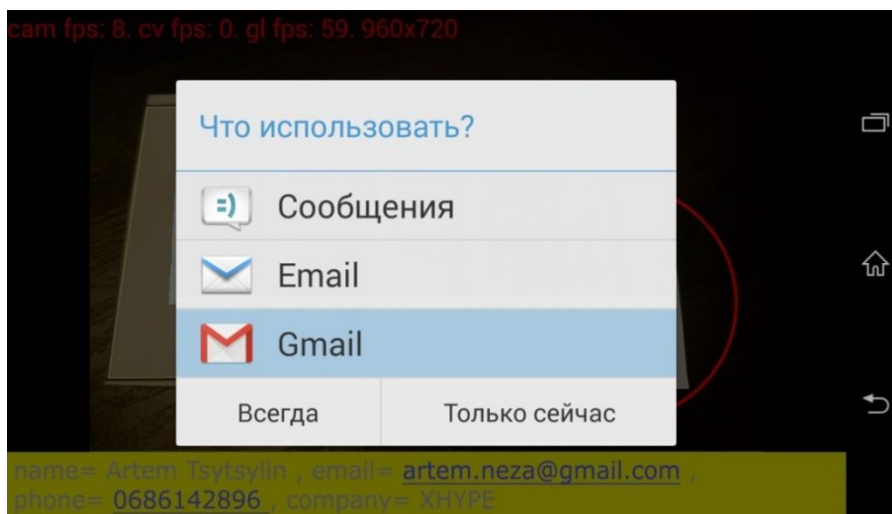


Рисунок 5.11 – Вікно з варіантами відкриття посилань

## 6 ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

### 6.1 Загальні питання з охорони праці

Завданням даної магістерської роботи було дослідження методів систем доповненої реальності, і як результат розробка способів побудови доповненої реальності і реалізація системи під платформу Android, яка покаже потенціал технології доповненої реальності на мобільних пристроях. Так як в процесі роботи використовувався персональний комп'ютер, то аналіз потенційно небезпечних і шкідливих виробничих чинників виконується для умов праці з використанням персонального комп'ютера.

Умови праці на робочому місці, безпека технологічних процесів, машин, механізмів, устаткування та інших засобів виробництва, стан засобів колективного та індивідуального захисту, що використовуються працівником, а також санітарно-побутові умови повинні відповідати вимогам нормативних актів про охорону праці. В законі України «Про охорону праці» [1] визначається, що охорона праці - це система правових, соціально-економічних, організаційно-технічних, санітарно-гігієнічних і лікувально-профілактичних заходів та засобів, спрямованих на збереження життя, здоров'я і працездатності людини у процесі трудової діяльності.

При роботі з обчислювальною технікою змінюються фізичні і хімічні фактори навколишнього середовища: виникає статична електрика, електромагнітне випромінювання, змінюється температура і вологість, рівень вміст кисню і озону в повітрі. Повітря забруднюється шкідливими хімічними речовинами антропогенного походження за рахунок деструкції полімерних матеріалів, які використовуються для обробки приміщень та обладнання. Неправильна організація робочого місця сприяє загальному і локальній напрузі м'язів шиї, тулуба, верхніх кінцівок, викривлення хребта і розвитку остеохондрозу. На всіх підприємствах, в установах, організаціях повинні створюватися безпечні і нешкідливі умови праці. Забезпечення цих умов покладається на власника або уповноважений ним орган (далі роботодавець). Умови праці на робочому місці, безпека технологічних процесів, машин, механізмів, устаткування та інших засобів виробництва, стан засобів колективного та індивідуального захисту, що використовуються працівником, а також санітарно-побутові умови повинні відповідати вимогам нормативних актів про охорону праці. Роботодавець повинен впроваджувати сучасні засоби техніки безпеки, які запобігають виробничому травматизмові, і забезпечувати санітарно-гігієнічні умови, що запобігають виникненню

професійних захворювань працівників. Він не має права вимагати від працівника виконання роботи, поєднаної з явною небезпекою для життя, а також в умовах, що не відповідають законодавству про охорону праці. Працівник має право відмовитися від дорученої роботи, якщо створилася виробнича ситуація, небезпечна для його життя чи здоров'я або людей, які його оточують, і навколишнього середовища.

### **6.1.1 Правові та організаційні основи охорони праці**

Основним організаційним напрямом у здійсненні управління в сфері охорони праці є усвідомлення пріоритету безпеки праці і підвищення соціальної відповідальності держави, і особистої відповідальності працівників.

Державна політика в галузі охорони праці визначається відповідно до Конституції України Верховною Радою України і спрямована на створення належних, безпечних і здорових умов праці, запобігання нещасним випадкам та професійним захворюванням. Відповідно до статті 3 Закону України «Про охорону праці» [1] (далі – Закону) законодавство про охорону праці складається з Закону, Кодексу законів про працю України [2], Закону України "Про загальнообов'язкове державне соціальне страхування від нещасного випадку на виробництві та професійного захворювання, які спричинили втрату працездатності" [3] та прийнятих відповідно до них нормативно-правових актів, норм міжнародного договору (ратифіковані Конвенції і Рекомендації МОТ, директиви Європейської Ради).

На законодавчому рівні визначено такі пріоритетні напрямки з безпеки праці:

- кожен працівник несе безпосередню відповідальність за порушення зазначених Законом, нормами і правилами вимог;
- напрямки реалізації конституційного права громадян на їх життя і здоров'я в процесі трудової діяльності:
  - пріоритет життя і здоров'я працівників по відношенню до результатів виробничої діяльності підприємства;
  - повна відповідальність роботодавця за створення належних – безпечних і здорових умов праці;
  - соціальний захист працівників, повне відшкодування збитків особам, які потерпіли від нещасних випадків на виробництві та професійних захворювань;
  - комплексне розв'язання завдань охорони праці;
  - підвищення рівня промислової безпеки шляхом забезпечення суцільного технічного контролю за станом виробництв, технологій та продукції, а також сприяння підприємствам у створенні безпечних та нешкідливих умов праці;

- соціальний захист працівників, повне відшкодування збитків особам, які потерпіли від нещасних випадків на виробництві та професійних захворювань;
- використання економічних методів управління охороною праці, участь держави у фінансуванні заходів щодо охорони праці;
- використання світового досвіду організації роботи щодо поліпшення умов і підвищення безпеки праці на основі міжнародної співпраці.

Користувачі персональних комп'ютерів, для яких ця робота є головною, підлягають медичним оглядам: попереднім — під час влаштування на роботу і періодичним — протягом професійної діяльності раз на два роки. Жінок з часу встановлення вагітності та в період годування дитини грудьми до роботи з ПК не допускають.

Наявні трудові відносини між працівниками і роботодавцями в Україні за темою дипломного проекту регулюються Кодексом законів про працю (КЗпП) України, відповідно до якого права працюючої людини на охорону праці охороняються всебічно та норми охорони праці неухильно інтегровані до правил внутрішнього розпорядку організації/підприємства.

### **6.1.2 Організаційно-технічні заходи з безпеки праці**

В організації/підприємстві проводиться навчання і перевірка знань з питань охорони праці відповідно до вимог Типового положення про порядок проведення навчання і перевірки знань з питань охорони праці, затвердженого наказом Держнаглядохоронпраці України від 26.01.2005 N 15, зареєстрованого в Міністерстві юстиції України 15.02.2005 за N 231/10511 [4].

Обов'язковими вимогами враховане наступне:

- ознайомлення з правилами безпеки праці, одержання відповідних інструктажів засвідчується у журналі інструктажів.
- перед допуском до самостійної роботи кожен працівник має право на навчання з питань охорони праці і роботодавець зобов'язаний, і проводить таке навчання у вигляді двох інструктажів з питань охорони праці:
  - 1) вступного, який проводять працівники служби охорони праці об'єкта господарювання з усіма працівниками, яких приймають на роботу незалежно від їхньої освіти та стажу роботи за програмою, в якій подають загальні питання охорони праці із врахуванням її особливостей на об'єкті господарювання;
  - 2) первинного, який проводять керівники структурних підрозділів на місці праці з кожним працівником до початку їхньої роботи на цьому робочому місці.

Проходження працівником цих інструктажів з питань охорони праці підтверджується записами у відповідних журналах обліку інструктажів і скріплюється підписами осіб, які проводили інструктажі та осіб, які отримали інструктажі.

- 3) Повторний (не рідше одного разу в 6 місяців).
- 4) Позаплановий (при зміні правил охорони праці).
- 5) Поточний (проводять з працівниками перед виконанням робіт, на яких оформляється наряд-допуск)

– обов’язкові організаційні заходи перед початком, під час і після завершення роботи повинні включати перевірку (візуально) наявності і справності електрообладнання та його заземлення, а під час виконання роботи вимогу «не залишати без нагляду обладнання, яке працює». Після закінчення роботи - вимагається прибирання робочого місця, відключення всіх електроприладів від електромережі.

## 6.2 Аналіз стану умов праці

Робота над створенням системи оптимізації запитів до бази даних проходитиме в приміщенні гуртожитку СНУ ім. В. Даля (бульвар Дружби народів 41б). Для даної роботи достатньо однієї людини, для якої надано робоче місце зі стаціонарним комп’ютером.

### 6.2.1 Вимоги до приміщень

Геометричні розміри приміщення зазначені в табл. 6.1.

Таблиця 6.1 – Розміри приміщення

Найменування	Значення
Довжина, м	6
Ширина, м	3
Висота, м	2,5
Площа, м <sup>2</sup>	18
Об’єм, м <sup>3</sup>	45

Згідно з [5] розмір площі для одного робочого місця оператора персонального комп’ютера має бути не менше 6 кв. м, а об’єм — не менше 20 куб. м. Отже, дане приміщення цілком відповідає зазначеним нормам.

Для забезпечення потрібного рівного освітленості кімната має вікно та систему загального рівномірного освітлення, що встановлена на стелі. Для дотримання вимог пожежної безпеки встановлено порошковий вогнегасник та систему автоматичної пожежної сигналізації.

### 6.2.2 Вимоги до організації місця праці

При порівнянні відповідності характеристик робочого місця нормативним основні вимоги до організації робочого місця за [6] і відповідними фактичними значеннями для робочого місця, констатуємо повну відповідність.

Таблиця 6.2 - Характеристики робочого місця

Найменування параметра	Фактичне значення	Нормативне значення
Висота робочої поверхні, мм	750	680 ÷ 800
Висота простору для ніг, мм	730	не менше 600
Ширина простору для ніг, мм	660	не менше 500
Глибина простору для ніг, мм	700	не менше 650
Висота поверхні сидіння, мм	470	400 ÷ 500
Ширина сидіння, мм	400	не менше 400
Глибина сидіння, мм	400	не менше 400
Висота поверхні спинки, мм	600	не менше 300
Ширина опорної поверхні спинки, мм	500	не менше 380
Радіус кривини спинки в горизонтальній площині, мм	400	400
Відстань від очей до екрану дисплея, мм	800	700 ÷ 800

Приміщення кабінету має об'єм 45 м<sup>3</sup>, площу – 18 м<sup>2</sup>.

Температура в приміщенні протягом року коливається у межах 18–24°C, відносна вологість — близько 50%. Система вентилявання приміщення — природна неорганізована, а опалення — централізоване.

Розміщення вікон забезпечує природне освітлення з коефіцієнтом природного

освітлення не менше 1,5%, а загальне штучне освітлення, яке здійснюється за допомогою восьми люмінесцентних ламп, забезпечує рівень освітленості не менше 200 Лк.

За ступенем пожежної безпеки приміщення належить до категорії В.

### **6.2.3 Навантаження та напруженість процесу праці**

За фізичним навантаженням робота відноситься до категорії легкі роботи (Ia), її виконують сидячи з періодичним ходінням. Щодо характеру організування виконання дипломної роботи, то він підпадає під нав'язаний режим, оскільки певні розділи роботи необхідно виконати у встановлені конкретні терміни. За ступенем нервово-психічної напруги виконання роботи можна віднести до II – III ступеня і кваліфікувати як помірно напружений – напружений за умови успішного виконання поставлених завдань.

Під час виконання робіт використовують ПК та периферійні пристрої (лазерні та струменеві), що призводить до навантаження на окремі системи організму. Такі перекося у напруженні різних систем організму, що трапляються під час роботи з ПК, зокрема, значна напруженість зорового аналізатора і довготривале малорухоме положення перед екраном, не тільки не зменшують загального напруження, а навпаки, призводять до його посилення і появи стресових реакцій.

Наявні психофізіологічні небезпечні та шкідливі фактори:

- а) фізичного перевантаження:
  - статичного;
  - динамічного;
- б) нервово-психічного перевантаження:
  - розумового перенапруження;
  - монотонності праці;
  - перенапруження аналізаторів;
  - емоційних перевантажень.

Роботу за дипломним проектом визнано, таку, що займає 50% часу робочого дня та за восьмигодинної робочої зміни рекомендовано встановити додаткові регламентовані перерви тривалістю 15 хв через кожну годину роботи.

### 6.3 Виробнича санітарія

На підставі аналізу небезпечних та шкідливих факторів при виробництві (експлуатації), пожежної безпеки можуть бути надалі вирішені питання необхідності забезпечення працюючих достатньою кількістю освітлення, вентиляції повітря, організації заземлення, тощо.

#### 6.3.1 Аналіз небезпечних та шкідливих факторів при виробництві (експлуатації) виробу

Роботу, пов'язану з ЕОП з ВДТ, у тому числі на тих, які мають робочі місця, обладнані ЕОМ з ВДТ і ПП, виконують із забезпеченням виконання НПАОП 0.00-7.15-18 [9] «Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями», які встановлюють вимоги безпеки до обладнання робочих місць, до роботи із застосуванням ЕОМ з ВДТ і ПП. Переважно роботи за проектами виконують у кабінетах чи інших приміщеннях, де використовують різноманітне електрообладнання, зокрема персональні комп'ютери (ПК) та периферійні пристрої.

Основними робочими характеристиками персонального комп'ютера є наступні:

- робоча напруга  $U = +220\text{В} \pm 5\%$ ;
- робочий струм  $I = 2\text{А}$ ;
- споживана потужність  $P = 350\text{Вт}$ .

Робоче місце має відповідати вимогам Державних санітарних правил і норм роботи з візуальними дисплейними терміналами електронно-обчислювальних машин, затверджених постановою Головного державного санітарного лікаря України від 10.12.98 N 7 [6].

Аналіз небезпечних та шкідливих виробничих факторів виконується у табличній формі (табл. 6.3).



Таблиця 6.3 – Аналіз небезпечних і шкідливих виробничих факторів

Небезпечні і шкідливі виробничі фактори	Джерела факторів (види робіт)	Кількіс на оцінка	Нормативні документи
1	2	3	4
<b>Фізичні</b>			
- підвищений рівень напруги електричної мережі, замикання якої може відбутися через тіло людини	-//-	4	[7]
- недостатність природного світла	порушення умов праці (вимог до приміщень)	2	[8]
- недостатне освітлення робочої зони	порушення гігієнічних параметрів виробничого середовища	3	[8]
<b>Психофізіологічні:</b>			
- нервово-психічна перевантаження (розумове, перенапруження аналізаторів-зорових)	- пошук інформації для постановки теми; - пошук та аналіз аналогів і літератури; - пошук наявних технологій, моделювання та аналіз алгоритмів; - виконання роботи за темою диплома, тестування; - оформлення роботи	4	[9] [6]
- фізичні (статичне – сидіння)	порушення умов праці (організації місця праці- сидіння користувача, ) та організації робочого часу - безпервна робота)	2	[9] [6]

### 6.3.2 Пожежна безпека

Приміщення оснащено системою автоматичної пожежної сигналізації, має 1 вогнегасник ВП-5 із зарядом вогнегасної речовини 8-12 кг, відповідно до вимог чинного законодавства України. Проходи до засобів пожежогасіння вільні, не захаращуються та у разі потреби забезпечувати евакуацію всіх людей, які перебувають у приміщенні через один евакуаційний вихід з дверима на шляху евакуації, що відчиняється в напрямку виходу з будівлі від робочого місця. В приміщенні наявна затверджена «План-схема евакуації з кабінету (приміщення)».

Пожежна безпека при застосуванні ЕОМ забезпечується:

- 1) системою запобігання пожежі,
- 2) системою протипожежного захисту,
- 3) організаційно-технічними заходами.

Згідно ДБН В.2.5-28:2018 [8] таке приміщення, площею 25 м<sup>2</sup>, відноситься до категорії "В" (пожежонебезпечної) та для протипожежного захисту в ньому проектом передбачено устаткування автоматичною пожежною сигналізацією із застосуванням датчиків-сповіщувачів РІД-1 (сповіщувач димовий ізоляційний) в кількості 1 шт., і застосуванням первинних засобів пожежогасіння.

Горючими матеріалами в приміщенні, де розташовані ЕОМ, є:

- 1) поліамід – матеріал корпусу мікросхем, горюча речовина, температура самозаймання 420° С,
- 2) полівінілхлорид – ізоляційний матеріал, горюча речовина, температура запалювання 335° С, температура самозаймання 530° С,
- 3) склотекстоліт ДЦ – матеріал друкарських плат, важкогорючий матеріал, показник горючості 1.7А, не схильний до температурного самозаймання,
- А) пластикат кабельний №.489 – матеріал ізоляції кабелів, горючий матеріал, показник горючості більше 2.1,
- 5) деревина – будівельний і обробний матеріал, з якого виготовлені меблі, горючий матеріал, показник горючості більше 2.1, температура запалювання 255° С, температура самозаймання 399° С.

Продуктами згорання, що виділяються на пожежі, є: окис вуглецю; сірчистий газ; окис азоту; синильна кислота; акромін; фосген; хлор і ін. При горінні пластмас, окрім звичних продуктів згорання, виділяються різні продукти термічного розкладання: хлорангідридні кислоти, формальдегіди, хлористий водень, фосген, синильна кислота, аміак, фенол, ацетон, стирол.

## **6.4 Гігієнічні вимоги до параметрів виробничого середовища**

### **6.4.1 Освітлення**

Збільшення освітленості сприяє поліпшенню працездатності навіть в тих випадках, коли процес праці практично не залежить від зорового сприйняття. При поганому освітленні людина швидко втомлюється, працює менш продуктивно, виникає потенційна небезпека помилкових дій і нещасних випадків.

Освітленість приміщення має велике значення при роботі на ПЕОМ. Вона багато в чому визначається колірною і мережевий обстановкою. Для зменшеного поглинання світла стеля і стіни вище панелей (1,5-1,7м.). Якщо вони не облицьовані звукопоглинальним матеріалом, фарбуються білою водоемульсійною фарбою (коефіцієнт відбиття повинен бути не менше 0,7). Для забарвлення стіни панелей рекомендується віддавати перевагу світлим фарбам.

Природне освітлення, коли робочі місця з ПЕОМ розташовуються в один ряд по довжині приміщення на відстані 0,8 - 1,0 м від стіни з віконними прорізами, і екрани знаходяться перпендикулярно цієї стіни. Основний потік природного світла при цій повинен бути зліва. Не допускається спрямування основного світлового потоку природного світла праворуч, ззаду і спереду працює на ПЕОМ. Оптимальна відстань очей до екрана відео монітора повинна становити 60-70 см, допустиме не менше 50 см. Розглядати інформацію ближче 50 см не рекомендується.

У приміщенні, де розташовані ЕОМ передбачається природне бічне освітлення, рівень якого відповідає ДБН В.2.5-28:2018 [8]. Джерелом природного освітлення є сонячне світло. Регулярно повинен проводитися контроль освітленості, який підтверджує, що рівень освітленості задовольняє [6] і для даного приміщення в світлий час доби достатньо природного освітлення.

*Розрахунок освітлення.*

Для будівель виробництв світловий коефіцієнт приймається в межах 1/6 - 1/10:

$$\sqrt{a^2 + b^2} \cdot S_b = (1/8 \div 1/10) \cdot S_n \quad (6.1)$$

де  $S_b$  – площа віконних прорізів, м<sup>2</sup>;

$S_n$  – площа підлоги, м<sup>2</sup>.

$$S_n = a * b = 6 * 3 = 18 \text{ м}^2$$

$$S_{\text{вік}} = 1/8 * 18 = 2,25 \text{ м}^2$$

Приймаємо 2 вікна площею  $S = 1,13 \text{ м}^2$  кожне.

Розрахунок штучного освітлення виробляється по коефіцієнтах використання світлового потоку, яким визначається потік, необхідний для створення заданої освітленості при загальному рівномірному освітленні. Розрахунок кількості світильників  $n$  виробляється по формулі (4.2):

$$n = \frac{E \cdot S \cdot Z \cdot K}{F \cdot U \cdot M} \quad (6.2)$$

де  $E$  – нормована освітленість робочої поверхні, визначається нормами – 300 лк;

$S$  – освітлювана площа, м<sup>2</sup>;  $S = 18$  м<sup>2</sup>;

$Z$  – поправочний коефіцієнт світильника ( $Z = 1,15$  для ламп розжарювання та ДРЛ;  $Z = 1,1$  для люмінесцентних ламп) приймаємо рівним 1,1;

$K$  – коефіцієнт запасу, що враховує зниження освітленості в процесі експлуатації – 1,5;

$U$  – коефіцієнт використання, залежний від типу світильника, показника індексу приміщення і т.п. – 0,575

$M$  – число люмінесцентних ламп в світильнику – 2;

$F$  – світловий потік лампи – 5400лм (для ЛБ-80).

Підставивши числові значення у формулу (4.2), отримуємо:

$$n = \frac{300 \cdot 18 \cdot 1,1 \cdot 1,5}{5400 \cdot 0,575 \cdot 2} \approx 1$$

Приймаємо освітлювальну установку, яка складається з одного світильника, який складаються з 2-х люмінесцентних ламп загальною потужністю 160 Вт, напругою – 220 В.

## 6.5 Вентилювання

Здійснюється провітрювання приміщення, в залежності від погодних умов, тривалість повинна бути не менше 10 хв. Найкращий обмін повітря здійснюється при наскрізному провітрюванні.

## 6.6 Розрахунок захисного заземлення (забезпечення електробезпеки будівлі).

Загальний опір захисного заземлення визначається за формулою:

$$R_{\text{ззн}} = \frac{R_3 \cdot R_n}{R_n \cdot n \cdot \eta_3 + R_3 \cdot \eta_n}, \quad (6.3)$$

де  $R_3$  - опір заземлення, якими когут бать труби, опори, кути і т.п., Ом;

$R_n$  - опір опори, яке з'єднує заземлювачі, Ом;

$n$  - кількість заземлювачів;

$\eta_3$  - коефіцієнт екранування заземлювача; приймається в межах  $0,2 \div 0,9$ ;  $\eta_3 = 0,7$

$\eta_n$  - коефіцієнт екранування сполучної стійки; приймається в межах  $0,1 \div 0,7$ ;  $\eta_n = 0,5$ ;

Опір заземлення визначається за формулою:

$$R_3 = \frac{\rho}{2\pi \cdot l} \cdot \left( \ln \frac{2 \cdot l}{d} + \frac{1}{2} \ln \frac{4 \cdot t + l}{4 \cdot t - l} \right), \quad (6.4)$$

де  $\rho$  - питомий опір ґрунту, залежить від типу ґрунту, Ом·м;

для піску -  $400 \div 700$  Ом·м; приймаємо  $\rho = 400$  Ом·м;

$l$  - довжина заземлювача, м; для труб - 2-3 м;  $l = 3$  м;

$d$  - діаметр заземлювача, м; для труб - 0,03-0,05 м;  $d = 0,05$  м;

$t$  - відстань від середини забитого в ґрунт заземлювача до рівня землі, м;  $t = 2$  м.

$$R_3 = \frac{400}{2 \cdot 3,14 \cdot 3} \left( \ln \frac{2 \cdot 3}{0,05} + \frac{1}{2} \ln \frac{4 \cdot 2 + 3}{4 \cdot 2 - 3} \right) = 110, \text{ Ом}$$

Опір смуги, що з'єднує заземлювачі, визначається за формулою:

$$R_n = \frac{\rho}{2\pi \cdot L} \cdot \ln \frac{2 \cdot L^2}{b \cdot t_1}, \quad (6.5)$$

де  $L$  - довжина смуги, що з'єднує заземлювачі (м) і приблизно дорівнює периметру будівлі:  $P_{\text{буд.}} = 42 \cdot 2 + 38 \cdot 2 = 160$  м;  $L = 160$  м;

$b$  - ширина смуги, м;  $b = 0,03$  м;

$t_1$  - глибина заземлення від рівня землі, м;  $t_1 = 0,5$  м.

$$R_n = \frac{400}{2 \cdot 3,14 \cdot 160} \cdot \ln \frac{2 \cdot 160^2}{0,03 \cdot 0,5} = 5,99, \text{ Ом}$$

Кількість заземлювачів захисного заземлення визначається за формулою:

$$n = \frac{2 \cdot R_z}{4 \cdot \eta_z}, \quad (6.6)$$

де 4 - допустимий загальний опір, Ом;

2 - коефіцієнт сезонності.

Визначаємо загальний опір захисного заземлення:

$$R_{ззп} = \frac{110 \cdot 5,99}{5,99 \cdot 79 \cdot 0,7 + 110 \cdot 0,5} = 1,7 \text{ Ом}$$

Висновок: дане захисне заземлення буде забезпечувати електробезпеку будівлі, так як виконується умова:  $R_{ззп} < 4 \text{ Ом}$ .

3) При виникненню пожеж при роботі на ПЕОМ від таких можливими джерел запалювання як:

- іскри і дуги коротких замикань;
- перегрів провідників, резисторів та інших радіодеталей ПЕОМ, від тривалої перевантаження та наявність перехідного опору;
- іскри при розмиканні і розмиканні ланцюгів;
- розряди статичної електрики;
- необережному поводженню з вогнем, а також вибухи газо-повітряних і пароповітряних сумішей.

## 6.7 Екологія

Діяльність за темою магістерської роботи, а саме: оптимізація запитів до бази даних в процесі її виконання впливає на навколишнє природне середовище і регламентується нормами діючого законодавства: Законом України «Про охорону навколишнього природного середовища» [11], Законом України «Про забезпечення санітарного та епідемічного благополуччя населення» [12], Законом України «Про відходи» [13].

В процесі діяльності з виконанням дипломного проектуванням виникають процеси поводження з відходами ІТ галузі. Нижче надано перелік відходів, що утворюються в процесі роботи:

- Відпрацьовані люмінесцентні лампи - I клас небезпеки.
- Змінні носії інформації - IV клас небезпеки.
- Відпрацьовані вогнегасники - IV клас небезпеки.
- Макулатура - IV клас небезпеки.

### **6.8 Висновки до розділу 6**

В даному розділі проведено аналіз потенційних небезпечних та шкідливих виробничих факторів, причин пожеж. Розглянуті заходи, які дозволяють забезпечити гігієну праці і виробничу санітарію. На підставі аналізу розроблені заходи з техніки безпеки та рекомендації з пожежної профілактики.

## 6.9 Перелік корисних посилань до розділу 6

1. Закон України «Про охорону праці». Режим доступу: <https://zakon.rada.gov.ua/laws/show/2694-12> - 10.14.1992 р.

2. Кодекс законів про працю України. Режим доступу: <https://zakon.rada.gov.ua/laws/show/322-08> - 10.12.1971 р.

3. Закон України "Про загальнообов'язкове державне соціальне страхування від нещасного випадку на виробництві та професійного захворювання, які спричинили втрату працездатності". Режим доступу: <https://dnaop.com/html/2065/doc-zakon-ukrajini-pro-zagalynoobovjzskove-derzhavne-socialyne-strahuvannya-vid-neshhasnogo-vipadku-na-virobnictvi-ta-profesijnogo-z> - 23.09.1999 р.

4. Нормативно-правовий акт з охорони праці. НПАОП 0.00-4.12-05 «Типове положення про порядок проведення навчання і перевірки знань з питань охорони праці» - Режим доступу: <https://zakon.rada.gov.ua/laws/show/z0231-05>

5. Державні санітарні норми. ДСН 3.3.6.042-99 «Санітарні норми мікроклімату виробничих приміщень» - Режим доступу: <https://zakon.rada.gov.ua/rada/show/va042282-99> - 01.02.1999 р.

6. Державні санітарні правила і норми. ДСанПіН 3.3.2.007-98 «Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин» - Режим доступу: <https://zakon.rada.gov.ua/rada/show/v0007282-98> - 10.12.1998 р.

7. Державний стандарт України. ДСТУ Б В.2.5-82:2016 «Електробезпека в будівлях і спорудах. Вимоги до захисних заходів від ураження електричним струмом» - Режим доступу: <http://epicentre.co.ua/dstu/doc28522.html> - 01.07.2016 р.

8. Державні будівельні норми. ДБН В.2.5-28:2018 «Природне і штучне освітлення» - Режим доступу: <http://www.minregion.gov.ua/wp-content/uploads/2018/12/V2528-1.pdf> - 03.10.2018

9. Нормативно-правовий акт з охорони праці. НПАОП 0.00-7.15-18 «Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями» - Режим доступу: <https://zakon.rada.gov.ua/laws/show/z0508-18> - 14.02.2018 р.

10. Державний стандарт України. ДСТУ Б В.1.1-36:2016 «Визначення категорій приміщень, будинків та зовнішніх установок за вибухопожежною та пожежною небезпекою» - Режим доступу: <https://zakon.rada.gov.ua/rada/show/v0158858-16> - 15.06.2016 р.

11. Закон України «Про охорону навколишнього природного середовища» - Режим доступу: <https://zakon.rada.gov.ua/laws/show/1264-12> - 26.06.1991 р.



12. Закони України «Про охорону навколишнього природного середовища» - Режим доступу - <https://zakon.rada.gov.ua/laws/show/4004-12> - 24.02.1994 р.

13. Закон України «Про відходи» - Режим доступу: <https://zakon.rada.gov.ua/laws/show/187/98-%D0%B2%D1%80> – 05.03.1998 р.

## ВИСНОВКИ

При виконанні магістерської роботи було досліджено тему доповненої реальності. Аналіз виявив, що на сьогодні ця область дуже швидко розвивається та буде корисною в багатьох галузях.

Було прийнято рішення про створення двох додатків для системи Android для мобільних пристроїв. Для цього були обрані безкоштовні та багатоплатформні бібліотеки та інструменти.

Спочатку була створена універсальна програмна частина системи, на базі якої побудовані два додатки, та в майбутньому можуть бути побудовані інші системи доповненої реальності.

Перший додаток повністю відповідає поставленій задачі – накладає мультиплікаційного персонажа поверх кадру з камери та анімує його в залежності від облич, які були знайдені на кадрі. На пристрої, на якому проводилося тестування, швидкодія виявилася достатньою, щоб оцінити можливості цієї системи доповненої реальності.

Другий додаток, відповідно до поставленої задачі, розпізнає картки з маркерами, читає дані, закодовані в QR-кодах, на основі цього виводить на екран віртуальну візитну картку з інформацією з бази даних та дозволяє використати цю інформацію одним натисканням. Нажаль, виникли проблеми на одному з останніх етапів обробки зображення, тому геометрична форма віртуальної картки не повністю відповідає дійсності. Але ця система дійсно має перспективи та після вирішення цієї проблеми можна привести її до вигляду зручного та конкурентоздатного додатку.

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Аналіз можливостей і перспектив використання доповненої реальності у теорії і на практиці [Текст] // Актуальні проблеми науки, економіки і освіти XXI века : матеріали II Міжнародної научно-практичної конференції, 05.03 – 26.09.2012: в 2-х ч. Ч. 2 – Самара: Самарський інститут РГТЭУ, 2012. – 392 с.— с. 247-251.
2. ARToolKit [Електронний ресурс] – Режим доступа: [www/URL: http://www.hitl.washington.edu/artoolkit/](http://www.hitl.washington.edu/artoolkit/) – 2004г. – Загл. з екрана.
3. Behind the work: Garage Magazine augmented reality app [Електронний ресурс] – Режим доступа: [www/URL: http://scanable.com/featured/behind-work-garage-magazine-augmented-reality-app/](http://www/URL: http://scanable.com/featured/behind-work-garage-magazine-augmented-reality-app/) – 2015г. – Загл. з екрана.
4. ANA and Evangelion collaboration [Електронний ресурс] – Режим доступа: [www/URL: http://www.aviationwire.jp/archives/11686](http://www/URL: http://www.aviationwire.jp/archives/11686) – 15.11.12г. – Загл. з екрана.
5. Gizmo uses augmented reality to bring greeting cards into the 21st century [Електронний ресурс] – Режим доступа: [www/URL: http://www.techhive.com/article/2045869/gizmo-uses-augmented-reality-to-bring-greeting-cards-into-the-21st-century.html](http://www/URL: http://www.techhive.com/article/2045869/gizmo-uses-augmented-reality-to-bring-greeting-cards-into-the-21st-century.html) – 05.08.13г. – Загл. з екрана.
6. Study concludes augmented reality technology under development at BMW will improve driver attention [Електронний ресурс] – Режим доступа: [www/URL: http://www.carsafetyrules.com/study-concludes-augmented-reality-technology-under-development-at-bmw-will-improve-driver-attention/0109/](http://www/URL: http://www.carsafetyrules.com/study-concludes-augmented-reality-technology-under-development-at-bmw-will-improve-driver-attention/0109/) – 09.01.13г. – Загл. з екрана.
7. HERE City Lens [Електронний ресурс] – Режим доступа: [www/URL: https://www.windowsphone.com/ru-ru/store/app/here-city-lens/b0a0ac22-cf9e-45ba-8120-815450e2fd71](http://www/URL: https://www.windowsphone.com/ru-ru/store/app/here-city-lens/b0a0ac22-cf9e-45ba-8120-815450e2fd71) – 22.01.14г. – Загл. з екрана.
8. Role of Augmented Reality in Medical Industry [Електронний ресурс] – Режим доступа: [www/URL: http://www.augmentedrealitytrends.com/augmented-reality/medical-augmented-reality.html](http://www/URL: http://www.augmentedrealitytrends.com/augmented-reality/medical-augmented-reality.html) – 15.01.14г. – Загл. з екрана.
9. Parallel Tracking and Mapping for Small AR Workspaces Industry [Електронний ресурс] – Режим доступа: [www/URL: http://www.robots.ox.ac.uk/~gk/PTAM/](http://www/URL: http://www.robots.ox.ac.uk/~gk/PTAM/) – 03.02.14г. – Загл. з екрана.
10. P. Viola and M. Jones. Robust real-time object detection. [Текст] / P. Viola and M. Jones. Technical Report CRL 2001/01. The Cambridge Research Laboratory, February 2001.
11. Р.Гонсалес Цифрова обробка зображень [Текст] / Р.Гонсалес, Р.Вудс - М : Техносфера, 2005. – 1072 с

12. Yoav, Freund. A decision-theoretic generalization of on-line learning and an application to boosting. / Yoav Freund and Robert E. Schapire *Journal of computer and system sciences* 55, pp. 119-139, 1997.
13. Історія створення QR кода [Електронний ресурс] – Режим доступу: [www/URL: http://qr.co.ua/history.htm](http://qr.co.ua/history.htm) – Загл. з екрана.
14. N. Otsu. A threshold selection method from gray-level histograms. [Текст] / N. Otsu. *IEEE Trans. Sys., Man., Cyber.* 9: pp. 62-66. 1979.
15. Suzuki, S. *Topological Structural Analysis of Digitized Binary Images by Border Following*. [Текст] / Suzuki, S. Abe, K., *CVGIP* 30 1, pp 32-46. 1985.
16. Основи QR-кода, або как написати свій генератор QR-кода [Електронний ресурс] – Режим доступу: [www/URL: http://qrcodes.com.ua/433-osnovy-qr-code-ili-kak-napisat-svoj-generator-qr-koda/](http://qrcodes.com.ua/433-osnovy-qr-code-ili-kak-napisat-svoj-generator-qr-koda/) 11.01.12г. – Загл. з екрана.
17. Морелос-Сарагоса, Р. Мистецтво завадостійкого кодування. Методи, алгоритми, застосування. [Посібник] / Р. Морелос-Сарагоса; Пер. с англ. - М. : Техносфера, 2005. - 320 с.
18. Ерухимов, В.Л. Розробка мультимедійних додатків з використанням бібліотек OpenCV та IPP [Посібник] / Ерухимов В.Л., Лысенков И.Д. Н. Новгород 2013г.
19. Sample Data Live2D Cubism Documents [Електронний ресурс] – Режим доступу: [www/URL: http://sites.cybernoids.jp/cubism\\_e/samples](http://sites.cybernoids.jp/cubism_e/samples) – 2014г. – Загл. з екрана.
20. Open Source QR Code Library [Електронний ресурс] – Режим доступу: [www/URL: http://qrcode.sourceforge.jp/](http://qrcode.sourceforge.jp/) – 2007г. – Загл. з екрана.

**ДОДАТОК А**  
**Лістинг “CvEngineLive2d.java”**

```
1     package com.kharj.ardiplom;
2     import java.io.File;
3     import java.io.FileOutputStream;
4     import java.io.InputStream;
5     import org.opencv.android.BaseLoaderCallback;
6     import org.opencv.android.CameraBridgeViewBase;
7     import org.opencv.android.LoaderCallbackInterface;
8     import org.opencv.android.CameraBridgeViewBase.CvCameraViewFrame;
9     import org.opencv.core.Core;
10    import org.opencv.core.Mat;
11    import org.opencv.core.Scalar;
12    import org.opencv.objdetect.CascadeClassifier;
13    import com.kharj.ardiplom.AREngineBase.MSG_ENUM;
14    import android.app.Service;
15    import android.content.Context;
16    import android.util.Log;
17    import android.view.Surface;
18    import android.view.SurfaceView;
19    import android.view.WindowManager;
20    public class CvEngineLive2D extends CvEngineBase {
21        protected boolean isShowImage = true;
22        protected boolean flip = false;
23        @Override
24        public boolean Init(Context context, AREngineBase _engine,
25    CameraBridgeViewBase _cameraView){
26            if(context == null || _cameraView == null)
27                return false;
28            fpsLastTime = System.currentTimeMillis();
29            fpsFrames = 0;
30            this.engine = _engine;
31            this.cameraView = _cameraView;
32            this.appContext = context;
33            this.loaderCallback = new BaseLoaderCallback(appContext) {
34                @Override
35                public void onManagerConnected(int status) {
36                    switch (status) {
```

```

37         case LoaderCallbackInterface.SUCCESS:
38             {
39                 Log.i(AREngineBase.TAG, "OpenCV loaded
40 successfully");
41                 //Lib
42                 System.loadLibrary("ar-diplom-native");
43                 //Processor
44                 createProcessor();
45                 if(processor != null){
46                     processor.Init(appContext);
47                 }
48                 cameraView.enableView();
49             } break;
50         default:
51             {
52                 super.onManagerConnected(status);
53             } break;
54         }
55     }
56 };
57     if(loaderCallback == null)
58         return false;
59     cameraView.setVisibility(SurfaceView.VISIBLE);
60     cameraView.setCvCameraViewListener(this);
61     return true;
62 }
63 @Override
64 protected void createProcessor(){
65     processor = new CvProcessorFaceDetection();
66 }
67 @Override
68 public Mat onCameraFrame(CvCameraViewFrame inputFrame) {
69     fpsWork();
70     final int rotation = ((WindowManager)
71 appContext.getSystemService(Context.WINDOW_SERVICE)).getDefaultDisplay
72 ().getRotation();
73     Mat source = inputFrame.rgba();
74     if((rotation == Surface.ROTATION_270 && !flip) || (rotation
75 != Surface.ROTATION_270 && flip)){

```

```

76         Core.flip(source, source, 0);
77     }
78     if(processor != null){
79         processor.pushFrame(source);
80         source = processor.pullFrame();
81         String faceData = processor.pullStringData();
82         engine.sendMessage(MSG_ENUM.TO_GL,          "FaceData",
83     faceData);
84     }
85     if(!isShowImage){
86         source.setTo(new Scalar(0, 0, 0));
87     }
88     return source;
89 }
90 @Override
91 public void onMessage(String code, String message) {
92     super.onMessage(code, message);
93     Log.i(AREngineBase.TAG, "msg to cv"+message);
94     if(code == "showImage"){
95         isShowImage = ((message=="1") ? true : false);
96     }else if(code == "smileTreshold"){
97         try{
98             int intValue = Integer.parseInt(message);
99             ((CvProcessorFaceDetection)processor).setSmileTreshold(((float
100 )intValue)*0.01f);
101         }catch(NumberFormatException ex){
102         }
103     }else if(code == "flip"){
104         this.flip = Boolean.parseBoolean(message);
105     }
106     }
107 }

```

**ДОДАТОК Б**  
**Лістинг “GLEngineLive2d.java”**

```
1     package com.kharj.ardiplom;
2     import java.util.NoSuchElementException;
3     import java.util.Scanner;
4     import android.content.Context;
5     import android.graphics.PixelFormat;
6     import android.opengl.GLSurfaceView;
7     import android.util.Log;
8     public class GLEngineLive2D extends GLEngineBase {
9         @Override
10        public boolean Init(Context _context, AREngineBase _engine,
11        GLSurfaceView _surfaceView){
12            if(_context == null || _surfaceView == null)
13                return false;
14            this.appContext = _context;
15            this.surfaceView = _surfaceView;
16            this.engine = _engine;
17            renderer = new GLRendererLive2D(appContext);
18            if(renderer == null)
19                return false;
20            surfaceView.getHolder().setFormat(PixelFormat.TRANSLUCENT);
21            surfaceView.setEGLContextClientVersion(1);//GLES 1.0
22            surfaceView.setEGLConfigChooser( 8, 8, 8, 8, 16, 0 );
23            surfaceView.setPreserveEGLContextOnPause(true);
24            surfaceView.setRenderer(renderer);
25            return true;
26        }
27        @Override
28        public void Destroy(){
29            super.Destroy();
30        }
31        @Override
32        public void onResume(){
33            super.onResume();
34        }
35        @Override
36        public void onPause(){
```



```

37         super.onPause();
38     }
39     @Override
40     public void onMessage(String code, String message) {
41         if(code=="FaceData"){
42             //float faceX = 0;
43             //float faceY = 0;
44             try{
45                 Scanner scanner = new Scanner (message);
46                 scanner.next ();
47                 final          float          faceX          =
48 ((float)scanner.nextInt()/1000.0f)-0.5f;
49                 final          float          faceY          =
50 ((float)scanner.nextInt()/1000.0f)-0.5f;
51                 int isSmile = scanner.nextInt();
52                 final float mouthForm = (isSmile==0) ? -0.5f :
53 1.0f;
54                 final float mouthOpen = (isSmile==0) ? -0.5f :
55 0.25f;
56                 final          float          faceAngle      =
57 ((float)scanner.nextInt()/1000.0f);
58                 surfaceView.queueEvent(new Runnable() {
59                     public void run() {
60                         ((GlRendererLive2D)renderer).setFaceTranslate(faceX, faceY);
61
62                         ((GlRendererLive2D)renderer).setMouthParams(mouthForm,
63 mouthOpen);
64
65                         //Do not change to 0.
66                         if(Math.abs(faceAngle)>0.01f)
67 ((GlRendererLive2D)renderer).setHeadAngle(faceAngle);
68                     }
69                 });
70                 //Log.i(AREngineBase.TAG,          "MSG="+message+"
71 XY:"+faceX+" "+faceY);
72
73         }
74     }
75 }

```

## Додаток В

### Комп'ютерна презентація

СХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
ІМЕНІ ВОЛОДИМИРА ДАЛЯ

ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ТА ЕЛЕКТРОНИКИ  
КАФЕДРА КОМП'ЮТЕРНИХ НАУК ТА ІНЖЕНЕРІЇ

Магістерська робота

на тему:

**«ДОСЛІДЖЕННЯ СПОСОБІВ ПОБУДОВИ  
ДОПОВНЕНОЇ РЕАЛЬНОСТІ І РЕАЛІЗАЦІЯ СИСТЕМИ  
ПІД ПЛАТФОРМУ ANDROID»**

Студента групи КН-18дм Гаврилова Данила Вадимовича

Науковий керівник Сафонова Світлана Олександрівна

Рисунок В.1 - Слайд №1

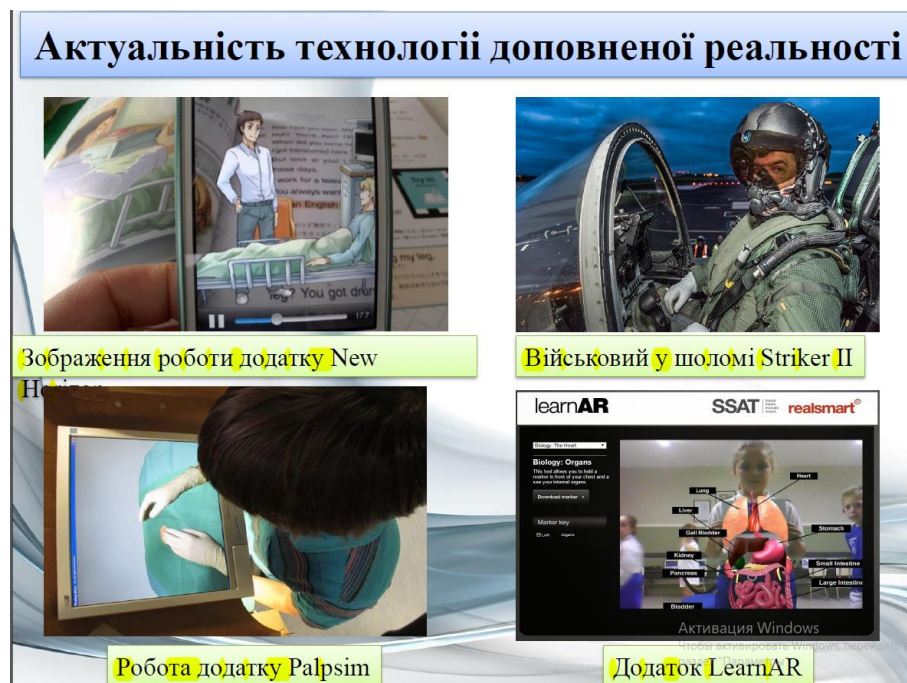


Рисунок В.2 - Слайд №2

### Постановка задачі дослідження

Мета роботи – створити та дослідити систему, яка покаже потенціал технології доповненої реальності на мобільних пристроях.

<p>Перший додаток буде виводити на екран мультиплікаційного персонажа та анімувати його залежно від знайдених облич у кадрі. Він буде мати два режими – коли персонаж залишається нерухомим, лише слідкуючи поглядом за обличчями, та коли копіює переміщення обличчя відносно камери та деякі його риси.</p>	<p>Другий додаток буде відображати віртуальні візитні картки, використовуючи реальну картку з зображенням на ній маркером як точку відліку. Дані для відображення на віртуальній картці завантажуються з бази даних, щоб відображати більше корисної інформації та дозволити швидко оперувати нею.</p>
---	--

Додатки мають підтримуватися багатьма мобільними пристроями на системі Android версії від 4.0, тому важливу роль грає оптимізація компонентів, чутливих до потужності системи.

Рисунок В.3 - Слайд №3

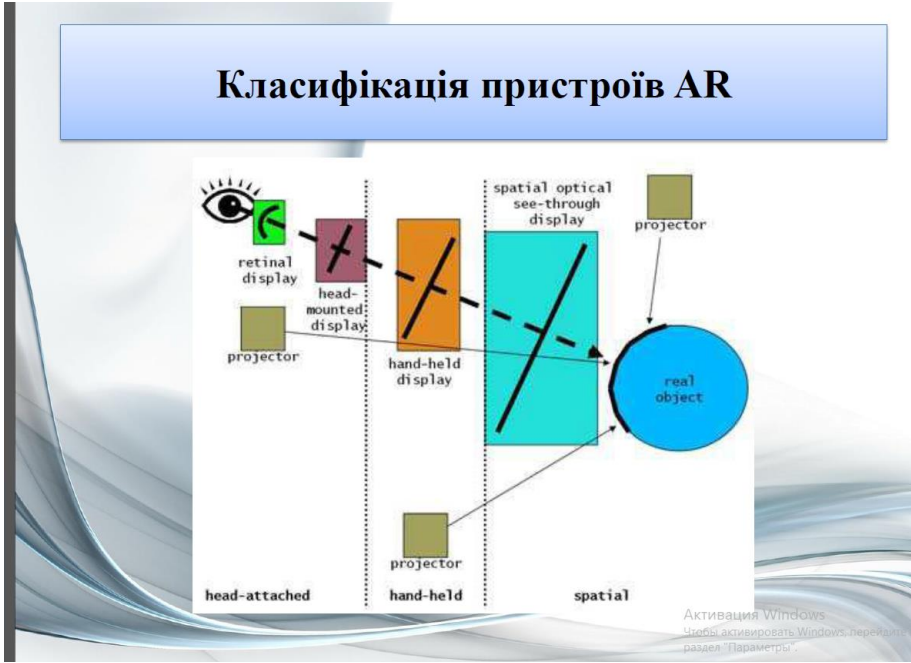


Рисунок В.4 - Слайд №4




Рисунок В.5 - Слайд №5



Рисунок В.6 - Слайд №6

### Алгоритми побудови доповненої реальності на основі розпізнавання маркерів

Artem Tsytsylin  
artem.neza@gmail.com





NORTH EAST  
WEST SOUTH

Структура маркеру у вигляді візитної картки

Приклад визначення орієнтації QR-коду за трьома точками

Активация Windows  
Чтобы активировать Windows, перейдите в раздел "Параметры".

Рисунок В.7 - Слайд №7

### Алгоритм читання QR-коду

Структура QR-коду

- 1. Код версії
- 2. Код маски і рівня корекції
- 3. Дані з кодами корекції
- 4. Незмінні частини QR-коду
  - 4.1. Пошуковий візерунок
  - 4.2. Вирівнювальний візерунок
  - 4.3. Лінії синхронізації
- 5. Біле поле



Знаходження пошукових узорів

Визначення версії коду

Пошук узорів для вирівнювання;

Читання системної інформації

Читання заголовку типу даних

Застосування маски до заголовку

Читання даних

Застосування маски до даних.

Активация Windows  
Чтобы активировать Windows, перейдите в раздел "Параметры".

Рисунок В.8 - Слайд №8



## Реалізація додатку для побудови доповненої реальності без маркерів

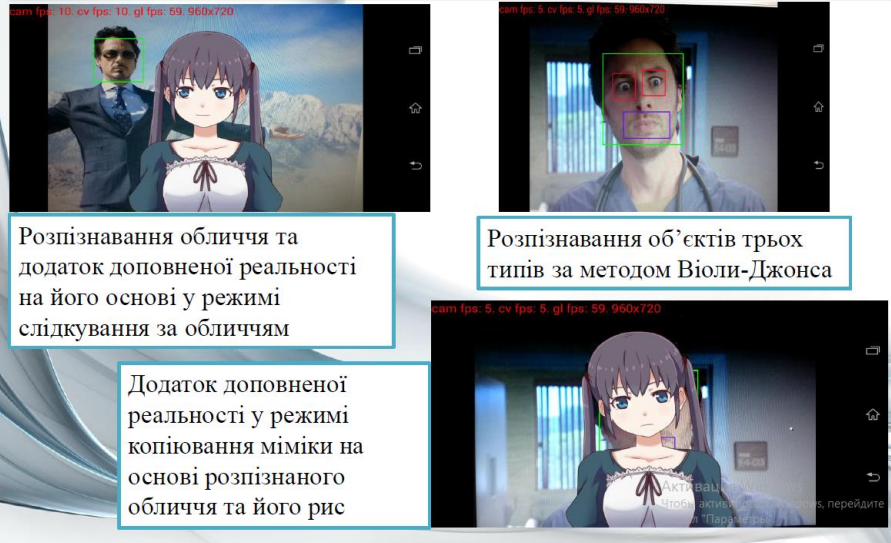


Рисунок В.11 - Слайд №11

## Реалізація додатку для побудови доповненої реальності на основі розпізнавання маркерів

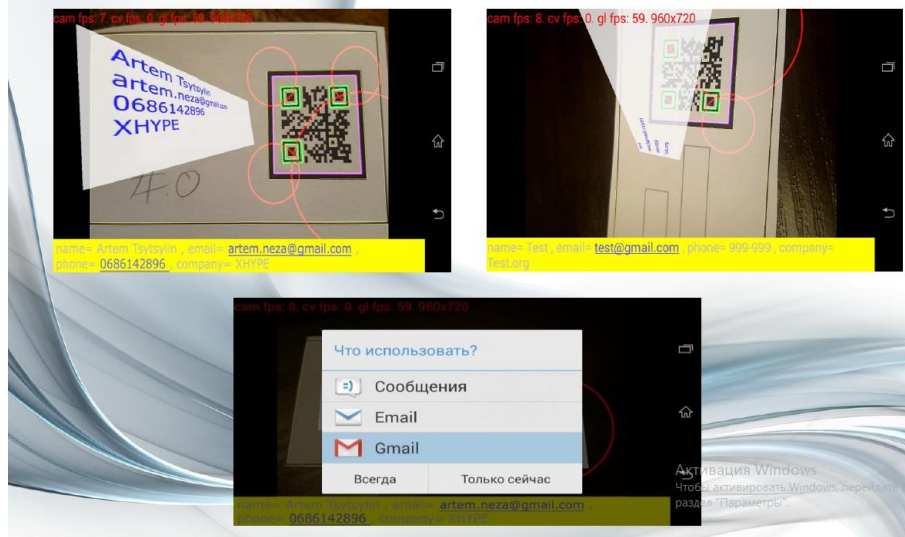


Рисунок В.12 - Слайд №12

## Висновки

- Створена універсальна програмна частина системи, на базі якої побудовані два додатки, та в майбутньому можуть бути побудовані інші системи доповненої реальності.
- Перший додаток повністю відповідає поставленій задачі – накладає мультиплікаційного персонажа поверх кадру з камери та анімує його в залежності від облич, які були знайдені на кадрі. На пристрої, на якому проводилося тестування, швидкодія виявилася достатньою, щоб оцінити можливості цієї системи доповненої реальності.
- Другий додаток розпізнає картки з маркерами, читає дані, закодовані в QR-кодах, на основі цього виводить на екран віртуальну візитну картку з інформацією з бази даних та дозволяє використати цю інформацію одним натисканням. Нажаль, виникли проблеми на одному з останніх етапів обробки зображення, тому геометрична форма віртуальної картки не повністю відповідає дійсності. Але ця система дійсно має перспективи та після вирішення цієї проблеми можна привести її до вигляду зручного та конкурентноздатного додатку.

Рисунок В.13 - Слайд №13