

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ**  
**«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»**  
Навчально-науковий комплекс “Інститут прикладного системного аналізу”  
(повна назва інституту/факультету)

Кафедра Системного проектування  
(повна назва кафедри)

«На правах рукопису»  
УДК 004.932.2

«До захисту допущено»

Завідувач кафедри

\_\_\_\_\_

\_\_\_\_\_

“ \_\_\_\_\_ ” \_\_\_\_\_ 20\_\_ р.

## Магістерська дисертація

зі спеціальності

8.05010102 Інформаційні технології проектування  
(код і назва спеціальності)

на тему:

Алгоритми комп’ютерного зору в системах доповненої реальності

Виконав (-ла): студент (-ка) 6 курсу, групи ДА-31м  
(шифр групи)

Архангельський Кирило Євгенович  
(прізвище, ім’я, по батькові)

\_\_\_\_\_

(підпис)

Науковий керівник к.т.н. доц. Харченко К.В.  
(посада, науковий ступінь, вчене звання, прізвище та ініціали)

\_\_\_\_\_

(підпис)

Консультант Охорона праці к.б.н. доц. Гусєв А.М.  
(назва розділу) (науковий ступінь, вчене звання, прізвище, ініціали)

\_\_\_\_\_

(підпис)

Рецензент \_\_\_\_\_  
(посада, науковий ступінь, вчене звання, науковий ступінь, прізвище та ініціали)

\_\_\_\_\_

(підпис)

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

Студент \_\_\_\_\_  
(підпис)

Київ – 2015 року

**Національний технічний університет України  
«Київський політехнічний інститут»**

Факультет (інститут) ННК “Інститут прикладного системного аналізу”  
(повна назва)

Кафедра Кафедра Системного проектування  
(повна назва)

Освітньо-кваліфікаційний рівень «магістр»  
Напря́м підготовки 6.050101 Комп’ютерні науки  
(код і назва)

Спеціальність 8.05010102 Інформаційні технології проектування  
(код і назва)

ЗАТВЕРДЖУЮ  
Завідувач кафедри  
\_\_\_\_\_ А.І. Петренко  
(підпис) (ініціали, прізвище)

«\_\_\_» \_\_\_\_\_ 20\_\_ р.

**ЗАВДАННЯ**  
**на магістерську дисертацію студенту**  
Архангельському Кирилу Євгеновичу  
(прізвище, ім’я, по батькові)

1. Тема дисертації Алгоритми комп’ютерного зору в системах доповненої реальності

науковий керівник дисертації Харченко К.В к.т.н., доц.,  
(прізвище, ім’я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від «13» лютого 2015 р. № 19/1-ст

2. Строк подання студентом дисертації 12.06.2015

3. Об’єкт дослідження: JSARToolKit, WebGL, three.js

4. Предмет дослідження: Бібліотеки комп’ютерного зору, для створення веб-додатку з використання доповненої реальності

5. Перелік завдань, які потрібно розробити:

- Провести детальний огляд існуючих бібліотек комп’ютерного зору;
- Розробити архітектуру додатку;
- Розробити прототип;
- Провести аналіз отриманих результатів.

6. Орієнтовний перелік ілюстративного матеріалу: презентація на тему “ Алгоритми комп’ютерного зору в системах доповненої реальності ”

7. Орієнтовний перелік публікацій: Архангельський К.Є. Алгоритми комп’ютерного зору в системах доповненої реальності / Архангельський К.Є. // Системний аналіз та інформаційні технології: матеріали 17-ї Міжнародної науково-технічної конференції SAIT 2015, Київ 22-25 червня 2015 р. — 2015. - с.184

8. Консультанти розділів дисертації

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Охорона праці	доц. к.б.н. Гусєв А.М.		
Основна частина	доц, к.т.н. Харченко К.В.		

9. Дата видачі завдання 30.09.2014

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Строк виконання етапів магістерської дисертації	Примітка
1	Отримання завдання	30.09.2014	
2	Збір інформації	21.01.2015	
3	Дослідження та аналіз вимог завдання, вибір доступних рішень для вирішення поставленої задачі	26.02.2015	
4	Огляд та порівняння існуючих бібліотек комп’ютерного зору, для розроблення додатку з використанням доповненої реальності	10.03.2015	
5	Розробка архітектури додатку	22.03.2015	
6	Розробка прототипу додатку доповненої реальності	05.04.2015	
7	Проведення аналізу отриманих результатів.	26.04.2015	
8	Робота над розділом охорони праці	10.05.2015	
9	Оформлення дипломної роботи	30.05.2015	
10	Отримання допуску до захисту та подача роботи в ДЕК	12.06.2015	

Студент

\_\_\_\_\_ (підпис)

К.Є. Архангельський  
(ініціали, прізвище)

Науковий керівник дисертації

\_\_\_\_\_ (підпис)

К.В. Харченко  
(ініціали, прізвище)

# РЕФЕРАТ

## на магістерську дисертацію

виконану на тему: Алгоритми комп'ютерного зору в системах доповненої  
реальності

студентом: Архангельським Кирилом Євгеновичем

Робота виконана на 100 сторінках, містить 47 ілюстрацій, 4 таблиці. При підготовці використовувалася література з 53 різних джерел.

### **Актуальність**

Технологія доповненої реальності з кожним днем стає все більш популярною і все частіше використовується в різних областях. Ця технологія має великий потенціал і тому вона активно розвивається.

При взаємодії з об'єктами в доповненій реальності на мобільних телефонах або будь-яких інших пристроях (ПК або окуляри доповненої реальності) користувач дивиться на реальний світ через камеру свого пристрою і при цьому зображення доповнюється віртуальними об'єктами, інтегрованими в фізичне оточення. Найчастіше взаємодія з користувачем в таких додатках обмежується звичайними натисканнями по екрану або за допомогою будь-якого іншого засобу введення, як і в звичайних додатках. Така взаємодія з віртуальними об'єктами є обмеженою і не зручною, наприклад, через те що розмір екрану пристрою маленький або громіздкості, ноутбук, тощо.

Для усунення цієї проблеми потрібна реалізація альтернативних методів взаємодії.

Зазначені обставини визначають актуальність і практичну значимість даної роботи.

## **Мета**

Метою цієї дипломної роботи є дослідження, і розробка системи доповненої реальності з підтримкою розпізнавання маркерів і накладення на них 3D об'єкту з текстурою.

## **Завдання**

Для досягнення поставленої мети необхідне рішення наступних завдань:

- Огляд існуючих рішень;
- Дослідження вимог, методів і алгоритмів вирішення поставленого завдання;
- Розробка структури програмного забезпечення;
- Створення дослідного зразка програмного забезпечення;

## **Об'єкт досліджень**

У відповідності до поставленої мети об'єктом досліджень обрано бібліотеку JSARToolKit.

## **Предмет досліджень**

Предметом досліджень в даній роботі обрано методи і алгоритми бібліотеки JSARToolKit, та функціональність WebGL для створення ефекту доповненої реальності.

## **Методи досліджень**

Проведення аналізу надійності використання отриманого веб-додатку з використанням бібліотеки комп'ютерного зору JSARToolKit та бібліотеки трьохвимірної графіки WebGL.

## **Наукова новизна**

Наукова новизна роботи полягає в перевірці надійності веб-додатку написаного для тестування можливостей доповненої реальності.

## **Практична цінність**

В ході роботи був розроблений веб-додаток з використанням бібліотеки JSARToolKit для роботи з доповненою реальністю. Також проведено налаштування бібліотеки three.js яка є розширенням WebGL для роботи з графічними елементами.

Результат даної роботи можна використовувати в подальшому для створення лабораторного практикуму на кафедрі СП на тему комп'ютерного зору та доповненої реальності.

## **Ключові слова**

Доповнена реальність, комп'ютерний зір, JSARToolKit, WebGL, WebRTC, three.js, веб-додаток.

# РЕФЕРАТ

## на магистерскую диссертацию

выполненную на тему: Алгоритмы компьютерного зрения в системах дополненной реальности

студентом: Архангельским Кириллом Евгеньевичем

Работа выполнена на 100 страницах, содержит 47 иллюстраций, 4 таблицы. При подготовке использовалась литература из 53 разных источников.

### **Актуальность**

Технология дополненной реальности с каждым днем становится все более популярной и все чаще используется в различных областях. Эта технология имеет большой потенциал и поэтому она активно развивается.

При взаимодействии с объектами в дополненной реальности на мобильных телефонах или любых других устройствах (ПК или очки дополненной реальности) пользователь смотрит на реальный мир через камеру устройства и при этом изображение дополняется виртуальными объектами, интегрированными в физическое окружение. Чаще всего взаимодействие с пользователем в таких приложениях ограничивается обычными нажатиями по экрану или с помощью любого другого средства ввода, как и в обычных приложениях. Такое взаимодействие с виртуальными объектами ограничено и не удобно, например, из-за того что размер экрана устройства маленький или громоздкости, ноутбук, и тому подобное.

Для устранения этой проблемы нужна реализация альтернативных методов взаимодействия.

Указанные обстоятельства определяют актуальность и практическую значимость работы.

## **Цель**

Целью этой работы является исследование и разработка системы дополненной реальности с поддержкой распознавания маркеров и наложения на них 3D объекта с текстурой.

## **Задачи**

Для достижения поставленной цели необходимо решение следующих задач:

- Обзор существующих решений;
- Исследование требований, методов и алгоритмов решения поставленной цели;
- Разработка структуры программного обеспечения;
- Создание опытного образца программного обеспечения;

## **Объект исследований**

В соответствии с поставленной целью объектом исследований избрана библиотека JSARToolKit.

## **Предмет исследований**

Предметом исследований в данной работе избраны методы и алгоритмы библиотеки JSARToolKit, и функциональность WebGL для создания эффекта дополненной реальности.

## **Методы исследований**

Проведение анализа надежности использования полученного веб-приложения с использованием библиотеки компьютерного зрения JSARToolKit и библиотеки трехмерной графики WebGL.



## **Научная новизна**

Научная новизна работы заключается в проверке надежности веб-приложения написанного для тестирования возможностей дополненной реальности.

## **Практическая ценность**

В ходе работы было разработано веб-приложение с использованием библиотеки JSARToolKit для работы с дополненной реальностью. Также проведена настройка библиотеки three.js которая является расширением WebGL для работы с графическими элементами.

Результат данной работы можно использовать в дальнейшем для создания лабораторного практикума на кафедре СП на тему компьютерного зрения и дополненной реальности.

## **Ключевые слова**

Дополненная реальность, компьютерное зрение, JSARToolKit, WebGL, WebRTC, three.js, веб-приложение.

# **ABSTRACT**

## **on master's thesis**

on topic: Algorithms for computer vision systems augmented reality

Student: Kyrylo E. Arkhanhelskyi

Work carried out on 100 pages containing 47 figures, 4 tables. The paper was written with references to 53 different sources.

### **Topicality**

Augmented reality technology with each passing day becomes more and more popular and are increasingly used in various fields. This technology has great potential and is therefore actively developing.

When interacting with objects in augmented reality on mobile phones or any other devices (PC or augmented reality glasses), the user is looking at the real world through the camera device while the image is complemented by virtual objects are integrated into the physical environment. Most often, the interaction with the user in such applications is limited usually by pressing on the screen or by any other input means, as in normal applications. Such interaction with virtual objects is limited and it is not convenient, for example, due to the fact that the screen size of the device small or bulkiness, laptop, and the like.

To resolve this issue necessary to implement alternative methods of interaction.

These circumstances determine the relevance and practical significance of the work.

### **Purpose**

The purpose of this work is to study and develop a system of augmented reality enabled detection markers and imposing on them a 3D object with texture.

## **Solution**

To achieve this goal, the following tasks:

- Review of existing solutions;
- Study requirements, methods and algorithms to solve this problem;
- Development of the structure of the software;
- Creating a prototype software;

## **The object of research**

In accordance with the purpose of the object of study chosen library JSARToolKit.

## **Subject of research**

The subject of the research in this paper chosen methods and algorithms library JSARToolKit, and functionality WebGL to create the effect of augmented reality.

## **Research Methods**

Analysis of the reliability of the use of the resulting Web application using the library computer vision library JSARToolKit and three-dimensional graphics WebGL.

## **Scientific novelty**

The scientific novelty of this work is to verify the reliability of Web-based applications written to test for augmented reality.

## **The practical value of research**

The work was developed a web application using the library JSARToolKit to work with augmented reality. Also held three.js library setting that is an extension of WebGL to work with graphic elements.

The result of this work can be used in the future to create a laboratory practical work at the Department of the joint venture on the theme of computer vision and augmented reality.

### **Keywords**

Augmented reality, computer vision, JSARToolKit, WebGL, WebRTC, three.js, a web application.

## ЗМІСТ

ЗМІСТ .....	8
СПИСОК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, СКОРОЧЕНЬ ТА ТЕРМІНІВ	11
ВСТУП .....	12
1 ДОПОВНЕНА РЕАЛЬНІСТЬ .....	14
1.1 Визначення доповненої реальності .....	14
1.2 Класифкація систем доповненої реальності .....	15
1.3 Пристрої доповненої реальності .....	18
1.3.1 Дісплеї .....	19
1.3.2 Пристрої введення.....	21
1.3.3 Пристрої відстеження .....	22
1.3.4 Desktopний пристрій .....	22
1.4 Інтерфейси доповненої реальності .....	22
1.4.1 Матеріальний AR інтерфейс .....	23
1.4.2 Спільний AR інтерфейс .....	23
1.4.3 Гібридний AR інтерфейс .....	24
1.4.4 Мультимодальний AR інтерфейс .....	24
1.5 Мобільні системи доповненої реальності .....	25
1.5.1 Соціально прийнятні технології .....	26
1.5.2 Персональні системи.....	26
1.5.3 Технології відслідковування для мобільних систем .....	27
1.6 Висновки.....	28
2 ОПИСАННЯ ВИКОРИСТОВУВАНИХ ТЕХНОЛОГІЙ КОМП'ЮТЕРНОГО ЗОРУ ТА ДОПОВНЕНОЇ РЕАЛЬНОСТІ .....	29
2.1 Бібліотека з відкритим кодом для комп'ютерного зору OpenCV .....	29
2.2 Використання бібліотеки OpenCV для побудови доповненої реальності ...	33
2.3 Графічні бібліотеки .....	36
2.3.1 OpenGL .....	36
2.3.2 WebGL .....	37
2.4 Інші бібліотеки комп'ютерного зору та доповненої реальності.....	38
2.4.1 AForge.Net .....	38
2.4.2 Metaio SDK.....	39

2.4.3	Vuforia SDK.....	40
2.4.4	Огляд бібліотеки комп'ютерного зору BoofCV .....	41
2.4.5	OpenVX.....	49
2.4.6	ARToolkit.....	51
2.5	Висновки.....	51
3	ОБЛАСТІ ЗАСТОСУВАННЯ ДОПОВНЕНОЇ РЕАЛЬНОСТІ .....	52
3.1	Огляд сервісів доповненої реальності.....	52
3.1.1	Проект Araned.....	52
3.1.2	Мобільні AR додатки.....	53
3.1.3	Спеціалізовані додатки.....	54
3.1.4	Розважальні та освітні додатки.....	55
3.1.5	Рекламні та комерційні додатки .....	59
3.2	Висновки.....	60
4	ВИКОРИСТАННЯ АЛГОРИТМІВ КОМП'ЮТЕРНОГО ЗОРУ В СИСТЕМАХ ДОПОВНЕНОЇ РЕАЛЬНОСТІ .....	62
4.1	Feature detection.....	62
4.2	Генетичні алгоритми .....	64
4.3	Узагальнений алгоритм розпізнавання маркера .....	65
4.4	Переклад кольорового зображення в градації сірого .....	66
4.5	Бінаризація зображення .....	68
4.6	Визначення замкнутих областей.....	70
4.7	Виділення контуру.....	71
4.8	Виділення кутів маркера.....	73
4.9	Перетворення координат .....	74
4.10	Висновки.....	76
5	РОЗРОБКА ПРОТОТИПУ .....	77
5.1	Вибір бібліотеки для розробки.....	77
5.1.1	Огляд альтернатив.....	77
5.2	Проектування прототипу .....	78
5.2.1	Архітектура .....	79
5.2.2	Розробка системи .....	81
5.2.3	Надійність .....	85

	10
5.3 Висновки.....	89
6 ОХОРОНА ПРАЦІ.....	90
6.1 Вступ.....	90
6.2 Характеристика приміщення.....	90
6.3 Мікроклімат робочого місця користувача ПК.....	91
6.4 Освітлення.....	97
6.5 Висновки.....	100
7 ВИСНОВКИ .....	101
8 СПИСОК ПОСИЛАНЬ.....	103

## **СПИСОК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, СКОРОЧЕНЬ ТА ТЕРМІНІВ**

OpenCV – бібліотека комп'ютерного зору з відкритим кодом

BoofCV – бібліотека комп'ютерного зору з відкритим кодом

OpenVX – стандарт комп'ютерного зору

AForgE.NET – C# фреймворк с відкритим вихідним кодом

Metaio SDK - набір інструментів для впровадження елементів доповненої реальності в додатки та програми

Vuforia SDK - платформа доповненої реальності та інструментарій розробника програмного забезпечення доповненої реальності

Unity3D - це інструмент для розробки дво- і тривимірних додатків та ігор

OpenGL - програмний інтерфейс для написання додатків, що використовують двовимірну і тривимірну комп'ютерну графіку

e-commerce - це сфера економіки, яка включає в себе всі фінансові та торговельні транзакції, що здійснюються за допомогою комп'ютерних мереж



## ВСТУП

Сучасний рівень розвитку обчислювальної техніки дозволяє зняти з людини частину праці, котру до цього вона витрачала на монотонну, однотипну роботу. Автоматизував деякий процес, людина може витратити сили на інші напрямки котрі потребують залучення розуму. Одним з методів автоматизації є пристосування комп'ютерного зору в різних галузях. Останнім часом це дуже популярний напрямок, котрий постійно розвивається, з'являються нові проекти на основі котрих лежить комп'ютерний зір та доповнена реальність.

Комп'ютерний зір – теорія і технологія створення машин, котрі можуть бачити. Як наукова дисципліна, комп'ютерний зір відноситься до теорії і технології створення штучних систем, котрі отримують інформацію із відеоданих. Як технологічна дисципліна, комп'ютерний зір прагне застосувати теорії і моделі комп'ютерного зору до створення систем комп'ютерного зору.

Доповнена реальність – змішана реальність, яка створюється за допомогою додаткових графічних елементів, що виводяться на екран пристрою.

Людина основну частину інформації про зовнішній світ отримує через зоровий канал та вельми ефективно обробляє, аналізує, а також інтерпретує отриману інформацію ззовні. Тому одразу постало питання, як можна реалізувати подібну систему обробки відеоданих для обчислювальної техніки. Данні технології використовуються для обчислювальної техніки. Технології комп'ютерного зору та доповненої реальності використовуються в достатньо популярних областях науки та техніки, таких як підвищення продуктивності, контроль виробничого обладнання, системи управління рухомим апаратами, біомедичні дослідження, підвищення якості виробів, що випускаються, автоматизація процесів і багато інших. Більше того, успіх сучасного бізнесу в основному залежить від якості пропонованої продукції. А для забезпечення якості

необхідний візуальний контроль. Отже ціль нашого дослідження показати на практиці як можна використати технології доповненої реальності в різних областях життєдіяльності людини.

# 1 ДОПОВНЕНА РЕАЛЬНІСТЬ

## 1.1 Визначення доповненої реальності

Термін доповненої реальності (augmented reality, AR) імовірно був запропонований працюючим на корпорацію Boeing дослідником Томом Коделом в 1990 р.

Важливо розуміти відмінності між доповненою реальністю і змішаною реальністю. У широкому сенсі доповнена реальність являє собою процес перегляду реального світу і віртуальних об'єктів одночасно, де віртуальна інформація накладається, вирівнюється і інтегрується в фізичному світі. У літературі по людино-машинній взаємодії доповнена реальність знаходиться в безперервному діапазоні інтерфейсів від «реальності» до віртуальної реальності «повного занурення» (Рисунок 1).

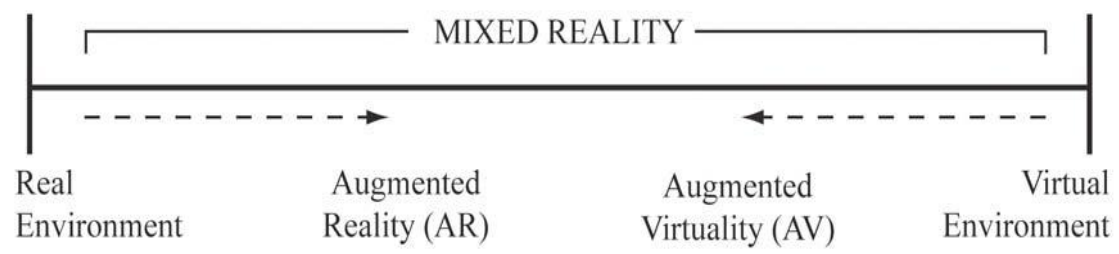


Рисунок 1.1 Діапазон технологій змішаної реальності

Істотною відмінністю доповненої реальності від віртуальної є збереження фізичного світу як контексту, в якому представлені віртуальні об'єкти і з яким вони взаємодіють. Віртуальна реальність повністю абстрагується від фізичного світу, щоб помістити користувача повністю у віртуальний світ. Віртуальна реальність використовує спеціальні позиційні трекери з дисплеями (окуляри віртуальної реальності), які динамічно оновлюють видимий користувачем простір у віртуальному середовищі. Важливо розуміти, що доповнена реальність повністю змінює цю парадигму, і в підсумку віртуальні об'єкти розміщуються в реальному

оточенні користувача.[1]

Таким чином, доповнена реальність (augmented reality) - це технології, що дозволяють доповнювати зображення реальних об'єктів різними об'єктами комп'ютерної графіки, а також поєднувати зображення, отримані від різних джерел комп'ютерного середовища: відеокамер, акселерометрів, компасів і т.д. Схема середовища доповненої реальності представлена на рисунку 2. На відміну від «віртуальної реальності», яка передбачає повністю штучний синтезований світ (відеоряд), доповнена реальність припускає інтеграцію віртуальних об'єктів у природні відеосцени.

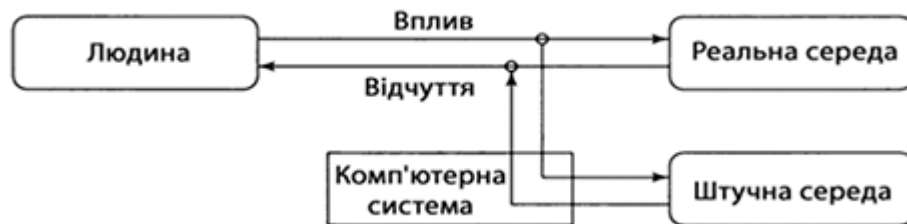


Рисунок 1.2 Схема середовища доповненої реальності

Рональд Азума (Ronald Azuma) виділив ряд ознак, якими має володіти розширена реальність:

- 1) комбінування реального та віртуального світу;
- 2) інтерактивність;
- 3) тривимірне представлення об'єктів.

## 1.2 Класифкація систем доповненої реальності

Людина отримує уявлення про навколишній простір за допомогою великого набору органів чуття. Система доповненої реальності, будучи посередником між людиною і реальністю, повинна створювати сигнал для одного з таких органів.

Таким чином, за типом подання інформації системи доповненої реальності бувають:

1. Візуальні. В їх основі лежить зорове сприйняття людини. Завдання таких систем - створити зображення, яке буде використано людиною. Оскільки зображення для людини є більш інформативним і зрозумілим, такий вид систем є більш поширеним.

2. Аудіо. Такі системи орієнтовані на слухове сприйняття. Особливості таких систем розглядаються в роботі. Найчастіше такі системи використовуються в навігації. Наприклад, вони видають спеціальні сигнали, коли людина досягає певного місця. Можливе використання стереоскопічного ефекту, що дозволяє людині йти в потрібному напрямку, орієнтуючись на джерело звуку. Прикладом такої системи є Hear & There.[2]

3. Аудіовізуальні. Це комбінація двох попередніх типів, однак, аудіоінформація в них має лише допоміжний характер.

Системи доповненої реальності завжди потребують інформації, одержуваної з навколишнього середовища. Саме на основі цих даних будуються віртуальні об'єкти. Кожна з таких систем володіє певним набором сенсорів - пристроїв, що дозволяють збирати інформацію з навколишнього середовища: звукові і електромагнітні коливання, прискорення і т.д. Для класифікації має сенс розділяти сенсори не по типам реєстрованих фізичних величин, а за їх призначенням, оскільки подібні за своєю природою сигнали можуть нести різну інформацію. За типом сенсорів можна виділити наступні системи:

1. Геопозиційні. Орієнтуються, перш за все, на сигнали систем позиціонування GPS або ГЛОНАСС. На додаток до приймачів таких сигналів геопозиційному системи можуть використовувати компас і акселерометр для визначення кута повороту щодо вертикалі і азимута.[5]

2. Оптичні. Такі системи обробляють зображення, отримане з камери, які можуть переміщатися разом з системою або незалежно від неї.

Системи доповненої реальності можна розрізнити за ступенем взаємодії з користувачем. У деяких системах користувач грає пасивну роль, він лише спостерігає за реакцією системи на зміни в навколишньому середовищі. Інші ж системи вимагають активного втручання користувача - він може управляти як роботою самої системи для досягнення результатів, так і змінювати віртуальні об'єкти. За цією ознакою системи діляться на:

1. Автономні. Вони не вимагають втручання користувача. Завдання таких систем зводиться до надання інформації про об'єкти. Наприклад, подібні системи можуть аналізувати об'єкти, що знаходяться в полі зору людини і видавати довідкову інформацію про них. Також системи такого типу використовуються в медицині. Наприклад, система Gait Aid для людей з порушеннями опорно-рухового апарату. Вона шляхом використання віртуальних об'єктів надає мозку додаткову інформацію, яка допомагає координувати рухи.[4]

2. Інтерактивні. Такі системи засновані на взаємодії з користувачем. На різні дії користувач отримує різну відповідь. У подібних системах необхідний пристрій введення інформації. В якості такого пристрою може застосовуватися сенсорний екран мобільного телефону, планшет або спеціальний маніпулятор. Вибір пристроїв введення залежить від специфіки системи. У разі простих дій з віртуальним об'єктом, достатньо простого вказівного пристрою. Якщо ж необхідна імітація будь-яких реальних процесів і виконання складних маніпуляцій з об'єктами використовуються спеціальні маніпулятори, які мають різну кількість ступенів свободи. Прикладом можуть служити пристрої PHANTOM.

Інтерактивність виражається в різному ступені. Бувають системи, що дозволяють користувачеві активно змінювати віртуальне середовище. Зазвичай це системи-симулятори будь-яких реальних дій. Вони використовуються у разі, коли використання реальних об'єктів неможливо, наприклад, спеціалізовані медичні тренажери, що дозволяють початківцям лікарям відпрацьовувати необхідні навички.

Існують інші системи, де користувачеві не потрібно змінювати віртуальне середовище. Замість цього користувач вибирає, які віртуальні об'єкти він хоче побачити. Користувач також має можливість маніпулювати віртуальними об'єктами, але не на рівні структури, а на рівні відображення, тобто застосовувати, наприклад афінні перетворення типу повороту, переміщення і т.д. До даної групи можна віднести різні архітектурні системи, що дозволяють побачити, як впишеться в реально існуючу обстановку нова споруда або його частину, а також навігаційні та геоінформаційні системи. Подібні системи можуть показувати частини об'єктів інтересу, приховані іншими будівлями, додаткову інформацію про обрані об'єкти і т.д.[3]

За ступенем мобільності системи доповненої реальності можна класифікувати як:

1. Стационарні. Системи цього типу призначені для роботи в фіксованому місці; переміщення таких систем означає часткове або повне припинення їх працездатності.

2. Мобільні. Системи цього типу можуть без зусиль переміщатися; часто таке переміщення і лежить в основі виконуваної ними функції.

Належність до того чи іншого типу визначається функціями системи. Так, симулятор хірургічного столу не повинен бути мобільним, оскільки його завдання - відтворити для людини спеціальні умови, максимально наближені до реальних. У той же час навігаційна система повинна бути якомога більш мобільною, щоб вона могла переміщатися разом з транспортним засобом або людиною, не створюючи додаткових витрат на її переміщення.

### **1.3 Пристрої доповненої реальності**

Основними пристроями, використовуваними в системах доповненої реальності є дисплеї, пристрої введення, пристрої відстеження і комп'ютер.

### 1.3.1 Дісплеї

Існує три основних типи дисплеїв, використовуваних в доповненій реальності: head mounted displays (HMD), ручні дисплеї і просторові.[6]



Рисунок 1.3 Оптико-прозорий HMD

HMD є пристроєм, який закріплюється на голові користувача, або на спеціальному шоломі і поміщає зображення реального і віртуального середовища перед очима користувача. HMD може бути або відео-прозорим або оптико-прозорим (рисунок 3). Відео-прозорі системи більш вимогливі, ніж оптико-прозорі так, як вони вимагають, щоб користувач носив дві камери на голові, а отже обробки даних з обох камер, щоб відобразити як "реальну частину" доповненої сцени, так і віртуальні об'єкти. У цей час оптико-прозорі системи використовують технологію половини срібного дзеркала дозволяє дивитися на реальний фізичний світ через спеціальну «лінзу» накладає додаткову графічну інформацію. Сцена, а також реальний світ в такій системі сприймаються більш природно. З іншого боку, у відео-прозорих системах доповнена реальність вже спочатку поєднана з реальною, що дає набагато більше контролю над результатом. Таким чином, контроль над часом кінцевої сцени може бути досягнутий шляхом



синхронізації віртуального зображення до його відображення. В оптично-прозорому варіанті, відображення реального світу не може бути загальмовано, однак це призводить до затримки при введенні в систему, графіки і обробці зображень.[7]

Під ручними дисплеями розуміються невеликі обчислювальні пристрої з дисплеєм, які користувач може тримати в руках. Вони використовують відео-прозорі методи для накладення графіки на реальне середовище, а так само використовують додаткові датчики, такі як компас, GPS і акселерометр. Для реалізації систем доповненої реальності часто використовуються системи розпізнавання маркерів, такі як ARToolKit, або методи комп'ютерного зору, такі як SLAM. Такими пристроями можуть бути: смартфони, КПК і планшетні ПК. Смартфони портативні і широко поширені пристрої, що поєднують потужний процесор, камеру, набір необхідних сенсорів, що робить їх дуже перспективною платформою для AR систем. Планшетні ПК набагато потужніші, ніж смартфони, але вони значно дорожчі і занадто важкі для тривалого використання. Однак з розвитком технологій планшети стають все більш перспективною платформою для роботи з доповненою реальністю.

Просторова доповнена реальність (SAR) з використанням відеопроєкторів, оптичних елементів, голограм, та інших технологій надає графічну інформацію безпосередньо на фізичні об'єкти, не вимагаючи від користувача носити або тримати дисплей. Просторові дисплеї відокремлюють більшість технологій від користувача і інтегрують їх у навколишнє середовище. Це дозволяє SAR природно масштабуватися до груп користувачів, що дозволяє здійснювати їх спільну роботу, підвищуючи інтерес до таких систем доповненої реальності в університетах, лабораторіях, музеях. Існують три різні підходи до SAR, які в основному відрізняються за способом доповнення навколишнього середовища: відео-прозорий, оптико-прозорий і прямого доповнення. Відео-прозорі SAR будуються на основі екрану, загального для використання. Оптично-прозорі просторові

дисплеї генерують зображення, які поєднані в межах фізичного середовища. Просторові оптичні технології, такі як плоскі або зігнуті дзеркала світоделитель, прозорі екрани або оптичні голограми є важливими компонентами таких систем. Як і відео-прозорі дисплеї, побудовані на базі оптико-прозорих технологій системи не підтримують мобільних додатків за рахунок просторово-вирівняної оптики і технології дисплеїв.

В таблиці 1.1 представлено порівняння різних типів дисплеїв.

Таблиця 1.1 Порівняння різних типів дисплеїв.

Тип	HMD		Ручний	Просторові		
Техно-логія	Відео-прозорий	Оптико-прозорий	Відео-прозорий	Відео-прозорі	Оптико-прозорі	Пряме доповнення
Переваги	візуальний контроль, синхронізація віртуального і реального оточення, Очікування обробки відеопотоку з камер	технологія половини срібного дзеркала, натуральне сприйняття реального оточення	портативний, потужний, поєднання багатьох датчиків	Ціна, Може бути адаптований для стандартного устаткування	Найбільш натуральне сприйняття реальних об'єктів	Відображення прямо на поверхні фізичних об'єктів
Недоліки	Необхідно закріплювати пристрій на голові, Штучно сприйняття реального оточення	Тимчасова затримка, Тремтіння віртуальних елементів	Маленький дисплей, Вага (для планшетів)	Не підтримує мобільні системи	Не підтримує мобільні системи	Не залежить від користувача (всі бачать одне і теж)

### 1.3.2 Пристрої введення

Є багато типів пристроїв введення для AR систем. Деякі системи використовують рукавички. Інші, такі як ReachMedia використовують бездротові браслети. У разі смартфонів, телефон сам по собі може бути використаний як вказівний пристрій, наприклад, у додатку Google Sky Map на Android телефон вимагає від користувача направити камеру в напрямку зірки або планети, про яку

він хоче дізнатися. Вибір пристроїв введення в значній мірі залежить від типу розроблюваної системи і типу обраного дисплея. Наприклад, якщо додаток вимагає, щоб руки користувача були вільними, пристрої введення будуть обрані відповідно аналогічним чином, якщо система використовує портативний дисплей, розробники можуть використовувати як пристрій вводу сенсорний екран.[8]

### **1.3.3 Пристрої відстеження**

Пристроями стеження є цифрові камери або інші оптичні датчики, GPS, акселерометри, компаси, бездротові датчики і т.д. Кожна з цих технологій має різну ступінь точності і багато в чому залежить від типу розроблюваної системи. Основні технології відстеження доповненої реальності: механічна, магнітна, GPS, ультразвукова, інерціальна і оптична.

### **1.3.4 Десктопний пристрій**

Система доповненої реальності повинна володіти потужним процесором і мати достатній обсяг оперативної і відео пам'яті для обробки зображень з камери. З розвитком технологій і появою нових портативних і одночасно потужних пристроїв, таких як смартфони і планшетні ПК, проблема нестачі потужності відійшла на другий план.

## **1.4 Інтерфейси доповненої реальності**

Один з найбільш важливих аспектів при створенні систем доповненої реальності - це створити відповідний інтуїтивно зрозумілий інтерфейс між користувачем і віртуальними об'єктами в системі. Існують чотири основних способи взаємодії в додатках AR: матеріальний AR інтерфейс, спільний AR інтерфейс, гібридні AR інтерфейс, і нові мультимодальні інтерфейси.

### 1.4.1 Матеріальний AR інтерфейс

Матеріальні інтерфейси підтримують пряму взаємодію з реальним світом, з використанням реальних фізичних об'єктів та інструментів. Класичним прикладом матеріальних користувальницьких інтерфейсів є VOMAR додаток, розроблений компанією Kato, яке дозволяє людині вибрати і переставити меблі в кімнаті в доповненої реальності за допомогою реального, фізичного інтуїтивно зрозумілого жесту. Жести на основі команд, таких як "зачерпнути" об'єкт, щоб вибрати його для руху або «удару» по ньому, для відміни.

Ще один приклад матеріального користувацького інтерфейсу AR є ТаRuMa. ТаRuMa є настільним інтерфейсом, що використовують фізичні об'єкти для взаємодії з цифровим світом. Використовуючи реальні об'єкти, які користувач носить із собою, як запитів для пошуку місць або інформації на карті. Перевага такого додатка є, використання об'єктів в якості ключових слів, що усуває мовний бар'єр звичайного графічного інтерфейсу (хоча більшість з них мають мульти мовну версію, вони часто бувають неправильно переведені). З іншого боку, використання об'єктів як ключових слів, може бути неоднозначним, через багато можливих трактувань.[9]

Іншим прикладом відчутного взаємодії AR є використання рукавичок або спеціальних браслетів.

### 1.4.2 Спільний AR інтерфейс

Спільні AR інтерфейси включають в себе використання декількох дисплеїв для підтримки віддаленої спільної діяльності. Для створення спільної робочої області використовуються 3D інтерфейс. У віддаленому обміні, AR може легко інтегруватися з декілька пристроїв з декількох місць для проведення телеконференцій.

Приклад такого інтерфейсу може бути Studierstube. При першому поданні Studierstube, розробники представили інтерфейс, який «використовує спільну

доповнену реальність для об'єднання декількох користувальницьких інтерфейсів: багатокористувацької, контекстного та регіонного, а також додатків, 3D-окн, хостів, дисплейних платформ і операційних систем».

Віддалений обмін може бути використаний для поліпшення телеконференцій. Такі інтерфейси можуть бути інтегровані з медичними додатками для виконання діагностики, операцій.[10]

### **1.4.3 Гібридний AR інтерфейс**

Гібридні інтерфейси поєднують особливості різних, але взаємодоповнюючих інтерфейсів, а також можливість взаємодіяти з допомогою широкого спектру пристроїв. Вони забезпечують гнучку платформу для незапланованої, щоденної взаємодії, коли невідомо заздалегідь, який тип дисплея або пристрою буде використовуватися.

### **1.4.4 Мультимодальний AR інтерфейс**

Мультимодальні інтерфейси об'єднують реальні форми пристроїв введення з натуральними формами мови та поведінки, такими як мова, дотик, жести рукою або напрям погляду. Ці типи інтерфейсів останнім часом найбільш популярні. Прикладом є «six sense» - розроблений в MIT інтерфейс жестів, званий WUW. WUW надає користувачеві інформацію, яка проектується на різні поверхні, а взаємодія здійснюється через природні жести рук, рухи користувачів або безпосередні маніпуляції з самим об'єктом. В іншому прикладі мультимодальної взаємодії використовується розпізнавання погляду користувача. Цей тип взаємодії в даний час швидко розвивається і, безсумнівно, буде одним з кращих типів взаємодії в майбутньому, оскільки він пропонує відносно надійну, ефективну і дуже мобільну форму взаємодії людини з комп'ютером. У мультимодальній системі є можливість гнучко поєднувати типи взаємодії або переключатися з одного режиму введення до іншого в залежності від завдання або налаштувань.

Крім того, мультимодальні інтерфейси можуть забезпечувати свободу вибору режиму взаємодії для користувача в залежності від контексту (громадське місце, музей, бібліотека і т.д.). Це свобода вибору способу взаємодії має вирішальне значення для більш широкого розповсюдження систем доповненої реальності в громадських місцях.[11]

## **1.5 Мобільні системи доповненої реальності**

Мобільні системи доповненої реальності включають в себе мобільні додатки для телефонів. Мобільні AR мається на увазі використання різних мобільних інтерфейсів для взаємодії користувача з віртуальними, даними, які доповнюють реальний світ. Використання мобільних телефонів для доповненої реальності має як переваги так і недоліки. Більшість мобільних пристроїв в даний час обладнано камерами, що робить мобільний телефон однієї з найбільш зручних платформ для реалізації систем доповненої реальності. Крім того, більшість стільникових телефонів мають додаткові вбудовані датчики такі як: акселерометри, магнітометри і GPS-приймачі, які можуть поліпшити роботу AR програми. Але, незважаючи на швидкий прогрес у розвитку мобільних телефонів, їх обчислювальна потужність для складних додатків і досі досить мала. В результаті, в багатьох додатках використовується клієнт-серверна архітектура, коли дані передають на віддалений комп'ютер, який виробляє обчислення і відправляє результат назад на мобільний пристрій. Але при такому підході може виникнути проблема обмеженої пропускну здатності, а це може бути критичним для складних AR систем. Тим не менш, з урахуванням швидкого розвитку мобільних технологій, ця проблема незабаром може бути вирішена, а це означає, що скоро з'явиться можливість створення додатків, що обробляють дані для AR локально в реальному часі.[12]

Успішною мобільною AR системою, як додатки, є система яка дозволяє користувачеві зосередитися на самому функціоналі системи, реалізує взаємодію з пристроєм в натуральному і соціально прийнятному вигляді, а також надає користувачеві додаткову корисну інформацію. Це вказує на необхідність розробки в легких, портативних мобільних пристроях, що володіють достатньою потужністю для складних обчислень і високими характеристиками датчиків для надійного стеження і розпізнавання.

### **1.5.1 Соціально прийнятні технології**

Багато дослідницьких груп піднімали проблему соціально прийнятних технологій. Мобільні системи, постійно стикаються з проблемою соціального визнання при переході з лабораторій до споживачів. Для систем, щоб бути успішним на ринку, розробники повинні враховувати не тільки технічні дані, але також і ергономічні та естетичні показники систем.

### **1.5.2 Персональні системи**

Мобільні системи доповненої реальності повинні бути особистими, це означає, що відображаєма інформація повинна бути доступна іншим людям, тільки з дозволу самого користувача. Розроблена в MIT система «six sense» хоч і дуже просунута, але не надає високого ступеня конфіденційності для своїх користувачів. Через використання прямого функціонального доповнення техніки без використання будь-яких пристроїв для захисту інформації, будь-який бажаний може побачити те ж саме, що і користувач. Це створює дилему: відсутність додаткових пристроїв робить WUW зручним і стильним пристроєм, привабливим для користувача; однак, це згубно впливає на конфіденційність.[13]

### 1.5.3 Технології відслідковування для мобільних систем

Добре відомо, що для якісних AR систем, щоб надати реалістичний результат потрібно дуже точно відстежувати реальну середу для подальшої інтеграції в неї віртуальних об'єктів. Найбільш поширений тип системи спостереження для мобільних систем - це стеження шляхом комбінування даних надходять з декількох датчиків. У вуличних системах в основному використовують GPS або інерційні методи відстеження за допомогою акселерометрів, гіроскопів, компасів та інших датчиків, поряд з методами комп'ютерного зору. Система GPS забезпечує простоту відстеження, незважаючи на малу точність. Для більш точної оцінки положення користувача і його орієнтація GPS використовується в поєднанні з різними інерціальними датчиками. Таким чином, точки інтересу користувача звужуються, і це дозволяє спростити візуальне відстеження. У приміщенні GPS володіє поганими показниками, а отже не може бути використана, тому використовуються тільки візуальні і інерційні методи. Поєднання цих методів має свої особливості: візуальне відстеження досягає найкращих результатів при низькій частоті руху, а інерціальні датчики краще працюють при високій частоті руху. Під час повільного руху вони не дають добрих результатів через шум і дрейфу зсуву. Взаємодоповнюючий характер цих систем призводить до спільного їх використання в більшості гібридних систем.[14]

Деякі системи покладаються тільки на комп'ютерний зір, але більшість з них розраховані на роботу в приміщеннях, де навколишнє середовище легко контролюється. Коли справа доходить до візуального відстеження на вулиці, з'являються зовнішні фактори, які значно ускладнюють завдання. Одна з найбільш "просунутих" мобільних систем є Google Goggles; ця система може: розпізнавати об'єкти простої форми, наприклад, штрих-коди або книги; визначати місце



розташування й напрямок руху, завдяки GPS і акселерометру, які допомагають системі визначити напрям погляду, щоб звизити точку інтересу.

## **1.6 Висновки**

Огляд існуючих додатків показав основні тенденції в розвитку технології доповненої реальності. Ця технологія в наш час дуже бурхливо розвивається і може знайти застосування в багатьох областях.

Аналіз додатків показав, що зараз вони мають, в основному, або розважальний, або вузькоспрямований характер. Основна увага в таких додатках приділяється їх можливостям при взаємодії з навколишнім середовищем і користувачем. Зважаючи на це, можна зробити висновок про актуальність розробки системи з розпізнаванням жестів, для взаємодії з віртуальними об'єктами.

Для створення функціонуючої системи необхідна досить потужна платформа, якою може бути сучасній мобільний пристрій, так як вони широко поширені і їх можливості постійно зростають.

## **2 ОПИСАННЯ ВИКОРИСТОВУВАНИХ ТЕХНОЛОГІЙ КОМП'ЮТЕРНОГО ЗОРУ ТА ДОПОВНЕНОЇ РЕАЛЬНОСТІ**

### **2.1 Бібліотека з відкритим кодом для комп'ютерного зору OpenCV**

OpenCV (Open Source Computer Vision Library) —бібліотека алгоритмів комп'ютерного зору, обробки зображень і чисельних алгоритмів спільного призначення з відкритим кодом. Реалізована на C/C++, також розробляється для Python, Ruby, Matlab. Може вільно використовуватися в академічних і комерційних цілях – розповсюджується в умовах ліцензії BSD (Berkeley Software Distribution). Бібліотека складається з 5 модулів (модуль `sxcore` є основним і називається ядром бібліотеки), кожний з котрих, реалізує визначений клас функціональності.[15]

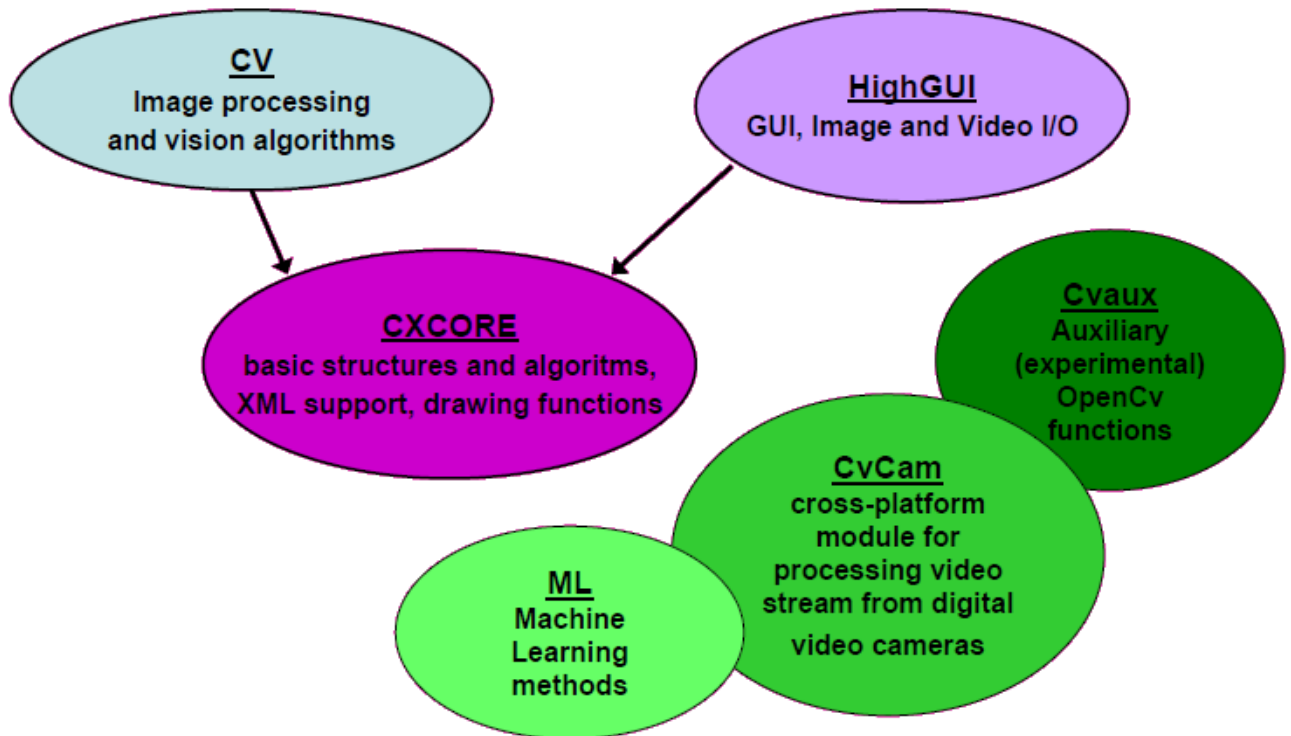


Рисунок 2.1 - Архітектура OpenCV [4]

Ядро `cxcore` реалізує наступну функціональність:

- 1) Базові операції над багатомірними числовими масивами;
- 2) Матрична алгебра, математичні функції, генератори випадкових чисел;
- 3) Базові функції 2Dграфіки;
- 4) Підтримка більш складних структур даних: розріджені масиви, динамічні зростаючі послідовності, графи.

Модуль `cv` надає функціональність для обробки зображень і комп'ютерного зору:

- 1) Базові операції над зображеннями (фільтрація, геометричні перетворення, перетворення кольорових просторів і такі інші);
- 2) Аналіз зображень (вибір відмінних знаків, морфологія, пошук контурів, гістограми);
- 3) Структурний аналіз (описання форм, плоскі розбиття);

- 4) Аналіз руху, спостереження за об'єктами;
- 5) Виявлення об'єктів, зокрема облич;
- 6) Калібровка камер, елементи відновлення просторової структури.

[5]

Модуль `highgui` призначений для введення / виведення зображень та відео, створення інтерфейсу користувача.

- 1) Захоплення відео з камер та із відео файлів, читання / запис статичних зображень;
- 2) Функції для організації простого інтерфейсу користувача.

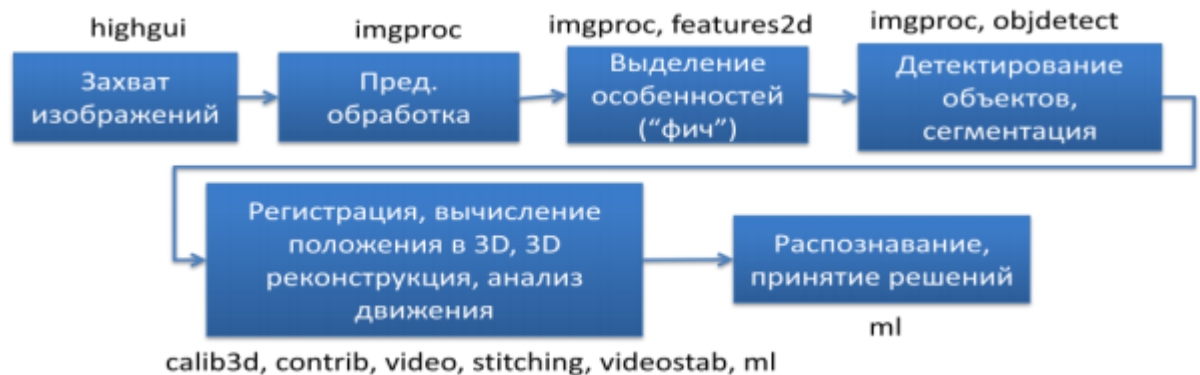


Рисунок 2.2 - Архітектура модуля `highgui` [6]

`SvAux` – модуль містить експериментальні та застарілі функції:

- 1) Об'ємний зір: стерео калібрація, само калібрація;
- 2) Пошук стерео відповідності, кліки в графах;
- 3) Знаходження та опис рис обличчя;
- 4) Порівняння форм;
- 5) Приховані Марківські кола;
- 6) Описання текстур.

`SvCam` — модуль для захоплення відео, на даний момент підтримка припинена. Якщо казати про те, що знаходиться в структурах даних `OpenCV`, то перше, про що треба казати, це зображення. Оскільки бібліотека створена

лабораторією Intel, немає нічого дивного в використанні для цих цілей IplImage — формату, взятого із Intel Image Processing Library (IPL). В IplImage інкапсулюється інші стандартні формати зображень. Крім глобальних операцій по створенню нового зображення або видаленню існуючого, є декілька макросів по піксельного редагування. В структурі OpenCV, використовується 4 модулі бібліотеки OpenCV cv, cxsam та highgui, описаних в розділі «Склад, структура і призначення системи» в підрозділі «Огляд OpenCV». [16]

Структурні типи даних:

- 1) IplImage – зберігає зображення, та необхідну інформацію.
- 2) CvMemStorage – організовує динамічний список для збереження даних.
- 3) CvSeq – зберігає сегменти даних у вигляді динамічного списку.
- 4) CvCapture – використовується, для збереження адреси відео захоплюючого пристрою.

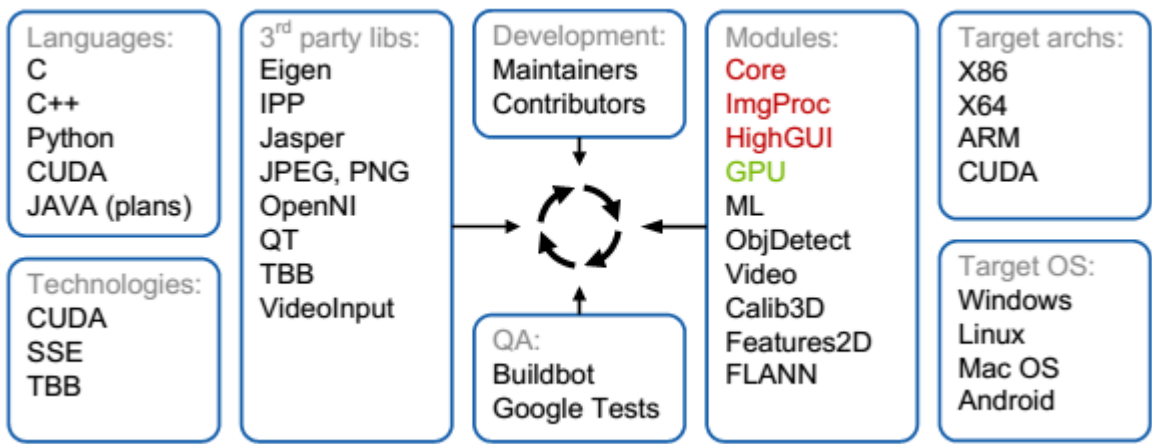


Рисунок 2.3 - Структура OpenCV [6]

## 2.2 Використання бібліотеки OpenCV для побудови доповненої реальності

Для вирішення даного завдання ми скористаємося описаною вище бібліотекою OpenCV. За допомогою неї буде проводитись пошук маркера на відеопотоці з подальшим відображенням віртуального об'єкта. Процес пошуку проводиться за допомогою алгоритму feature point detection. Перш за все, розглянемо побудову 3D простору по знайдений 2D гомографії.

Під гомографією розуміється матриця побудови взаємно-однозначного відображення проектованого простору на себе, що переводить точки в точки, прямі в прямі і зберігає відношення інцидентності точок і прямих, а також подвійне відношення четвірки колінеарних точок. Вона дозволяє приводити зображення до єдиної перспективи і геометрії.

Для побудови 3D простору нам необхідно знати 2 матриці: внутрішню (intrinsic matrix) і зовнішню (extrinsic matrix). Потім, скориставшись методами OpenGL, можна намалювати віртуальний об'єкт поверх маркера.

Внутрішня матриця (або матриця проєкції) складається з параметрів використовуваної камери: фокальної відстані по двох осях ( $f_x$ ,  $f_y$ ) і координат центру фокуса ( $c_x$ ,  $c_y$ ).

$$\begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$$

Процес знаходження параметрів камери називається її калібрацією. Для його проведення необхідно зробити знімки калібровочного шаблону (шахової дошки) за допомогою використовуваної камери, перенести фотографії на комп'ютер і

провести підрахунок параметрів матриці. З метою отримання якомога більш точних параметрів слід дотримуватись таких умов:

- Шаблон шахової дошки роздруковується на чистому аркуші формату А4.
- Лист шаблону повинен лежати рівно на поверхні, краї не повинні бути загнутими, бажано відсутність яких-небудь вигинів.
- Розмір знімків з камери повинен бути приведений до розміру кадру відеопотоку. У деяких випадках можлива ситуація, коли дозвіл фотографій вище дозволу відео. Тому, перед калібруванням необхідно зменшити знімки до потрібного розміру.
- Кількість знімків шаблону повинно бути не менше 10. Виконати їх потрібно з різних ракурсів. Чим більше буде знімків, тим точніші будуть отримані параметри матриці проекції, а ця точність далі буде впливати на наявність / відсутність зрушень при побудові 3D об'єктів.

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

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

Діагональні елементи відповідають за розтягнення моделі. Інші елементи  $r$  - за поворот об'єкту в просторі. Елементи  $t$  відповідають за перенесення. В цілому, структура даної матриці може варіюватися залежно від рівнянь перетворень координат, але описані ключові елементи в ній присутні завжди.

Обчислення її проводиться шляхом визначення чотирьох пар точок об'єкта та відповідного його положення в кадрі. Точки положення в кадрі - це вершини чотирикутника, що описує (обмежує) об'єкт в кадрі. Отримати дані точки можна, якщо подіяти на крайні точки шаблону гомографією.[43]

Щодо точок самого об'єкта варто відзначити кілька моментів:

- Точки об'єкта задаються в 3D, а точки на кадрі в 2D. Відповідно, якщо поставити їм нульове значення  $z$ , то початок координат на  $z$  буде зрушено щодо площини об'єкта на кадрі.



Рисунок 2.4 Положення об'єкта на маркері з напрямками осей координат

- Точки об'єкта задаються таким чином, щоб було зручно працювати в 3D просторі, а саме: початок координат знаходиться прямо по центру шаблону, а одиниця довжини дорівнює половині меншої сторони. У цьому випадку, ми виключаємо залежність від конкретних розмірів шаблону в пікселях, а 3D простір буде відмасштабований по меншій стороні.

Сконструйовані матриці потім використовуються для обчислення вектора повороту і перенесення з наступним занесенням в матрицю моделі. Варто



відзначити, що процес обчислення матриці моделі займає кілька мілісекунд і суттєво не впливає на швидкість роботи програми. Основне навантаження йде в галузі аналізу відеопотоку і пошуку на ньому шаблонного зображення.[42]

Описаний вище підхід можна застосувати до всіх бібліотека комп'ютерного зору в плані побудови доповненої реальності. З практичної точки зору він виглядає дещо скрутним зважаючи на необхідність проведення калібрування камери для отримання коректної матриці проектування. Одним із способів рішення даної проблеми є створення набору параметрів різних камер; залежно від використовуваної камери буде вироблятися завантаження необхідних параметрів.

Можна цілком побудувати свій власний фреймворк доповненої реальності. Розробляючи різні способи фільтрації точок, можна домогтися того, що маркер буде знаходитися навіть при сильному куті нахилу і великій відстані. Це дозволить отримати більш реалістичну картинку і дасть можливість користувачам розглянути модель з усіх боків на різній відстані.

## **2.3 Графічні бібліотеки**

### **2.3.1 OpenGL**

Графічна бібліотека OpenGL, як затверджений індустріальний стандарт, розроблена і затверджена в 1992 році дев'ятьма провідними IT-фірмами: Digital Equipment Corp., Evans & Sutherland, Hewlett-Packard Co., IBM Corp., Intel Corp., Intergraph Corp., Silicon Graphics Corp, Inc., Sun Microsystems, Inc., Microsoft Corp. В основі стандарту лежить бібліотека IRIS GL, розроблена Silicon Graphics. OpenGL широко використовується при розробці комп'ютерних ігор. Бібліотека OpenGL досить проста у використанні і навчанні, має дуже широкий спектр можливостей. Ось деякі з її переваг:

- *Стабільність*: OpenGL усталений стандарт. Всі зміни, внесені в нього, аноншуються заздалегідь і реалізуються так, щоб вже існуюче ПЗ не сблолю на нових графічних картах.
- *Надійність*: Всі додатки, що використовують OpenGL, гарантують однаковий візуальний результат, незалежно від устаткування і операційної системи.
- *Переносимість*: Програми, що використовують OpenGL, можуть запускатися на різних архітектурах і під різними операційними системами (природно, за умови перекомпіляції додатку, тобто OpenGL забезпечує переносимість на рівні вихідних кодів).

Головна особливість OpenGL - його клієнт-серверна архітектура, що дозволяє помістити клієнт (додаток, що використовують OpenGL) і сервер (виконавча частина OpenGL) на різні машини.[17]

OpenGL розвивається за допомогою механізму «розширень» - спеціальних модифікацій базової версії API OpenGL, які додають нові можливості та / або розширюють старі. Коли накопичується достатня кількість таких змін (розширень), консорціум OpenGL випускає специфікацію нової версії OpenGL.

### **2.3.2 WebGL**

WebGL (Web-based Graphics Library) - програмна бібліотека для мови програмування JavaScript, що дозволяє створювати на JavaScript інтерактивну 3D-графіку, що функціонує в широкому спектрі сумісних з нею веб-браузерів. За рахунок використання низькорівневих засобів підтримки OpenGL, частина коду на WebGL може виконуватися безпосередньо на відеокартах. WebGL - це контекст елемента canvas HTML, який забезпечує API 3D графіку без використання плагінів. Специфікація версії 1.0 була випущена 3 березня 2011 року. Проект зі створення бібліотеки управляється некомерційною організацією Khronos Group.

## 2.4 Інші бібліотеки комп'ютерного зору та доповненої реальності

### 2.4.1 AForge.Net

AForge.NET є C # фреймворком з відкритим вихідним кодом, створеним для розробників і дослідників у галузі комп'ютерного зору і штучного інтелекту. Основним ідеологом та розробником даного проекту є одна людина (Андрій Кирилов), однак протягом усього періоду свій внесок вносили розробники з різних країн світу.

Фреймворк включає в себе наступні компоненти:

- AForge.Imaging — обробка зображень і набір різних фільтрів;
- AForge.Vision — набір методів і алгоритмів комп'ютерного зору;
- AForge.Video — обробка відеопотоку;
- AForge.Neuro — побудова та робота з нейронними мережами;
- AForge.Genetic — набір генетичних алгоритмів;
- AForge.Robotics — спеціальний набір методів для застосування в галузі робототехніки;

та ряд інших.

На основі AForge.NET була розроблена бібліотека Grاتف, використовувана для побудови доповненої реальності. Вона написана на мові C # і добре переноситься на різні платформи.[46]

Крім зазначених вище, існує ряд інших кроссплатформених бібліотек комп'ютерного зору. Серед них можна виділити ROS (Robot Operating System) -

бібліотека з відкритим вихідним кодом, застосовувана для створення програмного забезпечення роботів. VXL, Integrating Vision Toolkit, ViSP - C ++ фреймворки з набором модулів з обробки та аналізу зображень, відеопотоку, пошуку шаблонів і об'єктів, класифікаторами і багатьом іншим. Список подібних бібліотек досить широкий, і чимале число з них побудовані на основі OpenCV.

### 2.4.2 Metaio SDK

З допомогою інструментів Metaio складні і ресурсомісткі функції комп'ютерного зору, такі, наприклад, як тривимірний трекінг реальних об'єктів, виконуються на мобільних платформах Intel якісно. Це означає, насамперед, що додатки, що використовують технології ДР, працюватимуть точно і стабільно. Впроваджена підтримка багатьох носимих гаджетів, поліпшена візуалізація та швидкість обробки об'єктів. До слова сказати, Metaio SDK - єдиний SDK в мобільному сегменті, що здійснює надійний безмаркерний трекінг тривимірних об'єктів реального світу, як, втім, і двовимірних. Остання версія Metaio SDK підтримує також трекінг по контурах, що дозволяє користувачам відстежувати геометрію реальних об'єктів, що значною мірою вирішує проблему непостійності освітленості. [18]

Основні характеристики версії Metaio SDK 5.0:

- Доступний для Android, iOS і ПК платформ, підтримує 2D, 3D, ID і SLAM трекінг;
- Підтримує носимі гаджети Google Glass, Epson Moverio BT-100 і Vuzix M-100;
- Підтримує 3D безмаркерний трекінг на основі САД даних (трекінг по контурах);
- Покращена якість візуалізації за допомогою програмованих шейдерів;

- Вбудований потужний 3D движок, оснащений багатопотоковим трекінгом і конвеєрами рендеринга;
- Підтримка складних 3D моделей (більше 32 тисячі багатокутників);
- Просте управління 3D контентом і його налагодження (BoundingBox, Normals, Wireframe).

### 2.4.3 Vuforia SDK

Vuforia SDK - це програмне забезпечення для мобільних пристроїв, яке дозволяє створювати додатки доповненої реальності. Воно використовує технологію комп'ютерного зору для того, щоб розпізнавати і відстежувати плоскі зображення і прості 3D-об'єкти в режимі реального часу. Ця можливість реєстрації зображень дозволяє визначати розташування й орієнтації віртуальних об'єктів, таких як 3D-моделі, в реальному світі, коли вони розглядаються через камеру мобільного пристрою. Положення і орієнтація віртуального об'єкта відстежується в реальному часі, так що точки зору глядача на об'єкт співвідносяться з їх точкою зору на зображення, так що здається, що віртуальний об'єкт є частиною реальної сцени світу.[19]

Vuforia SDK підтримує різні 2D і 3D цільові типи, включаючи безмаркерні цілі. Додаткові можливості SDK включає локалізовані виявлення оклюзії за допомогою "віртуальних кнопок, зображень виконання цільового відбору, а також можливість створити і змінити цільові набори програмно під час виконання.

Vuforia забезпечує API для C ++, Java, Objective-C, і .Net мов. Є розширення ігрового движка Unity. Таким чином, SDK підтримує як рідні для IOS і Android мови, так і одночасно дозволяє розробляти додатки доповненої реальності в Unity, які легко переносимі на обидві платформи. Додатки, розроблені з використанням Vuforia тому сумісні з широким спектром мобільних пристроїв, включаючи iPhone , iPad, Android і телефонів і планшетів під управлінням ОС Android версії 2.2 або вище і ARMv6 або 7 процесор з FPU.

#### 2.4.4 Огляд бібліотеки комп'ютерного зору VoofCV

Конкурентом OpenCV можна вважати бібліотеку комп'ютерного зору VoofCV вона реалізована на Java на відміну від OpenCV. На жаль, якщо ми захочемо написати щось серйозне, ми виявимо, що набір бібліотек з уже реалізованими функціями Computer Vision не так великий, особливо це стосується платформи Android. Найчастіше для цієї мети використовують OpenCV, написану на C ++. Існує такий проект як VoofCV, який представляє з себе бібліотеку комп'ютерного зору, написану на чистому Java. В останньому релізі з'явилася підтримка Android. Нижче ми розглянемо основні функції, надані бібліотекою на конкретному прикладі.[20]

Отже, VoofCV - бібліотека комп'ютерного зору з відкритим вихідним кодом, написана на чистому Java. Алгоритми, використовувані усередині, добре оптимізовані і, як показує практика, по швидкості в деяких випадках не поступаються реалізації на C ++ OpenCV. Основні можливості бібліотеки:

- Робота з відео та web камерами;
- 3D Computer Vision;
- Фільтри (розмиття, градієнт), прибирання шуму (за допомогою вейвлетів);
- бінаризації, морфологічні операції;
- Виділення меж (Кенні, Собель);
- Пошук точок інтересу;
- Пошук ліній, сегментів, прямокутників;
- Stereo зображення.

Крім цього, бібліотека містить ще безліч функцій, які у комп'ютерному зорі. Підключення бібліотеки - можна завантажити код з офіційного сайту і зібрати самому, а можна скачати вже зібрані jar файли і підключити їх до проекту.[9]



Рисунок 2.5 - експериментальне зображення



Рисунок 2.6 - розмиття Гаусса

Для отримання такого зображення ми використовуємо функцію `BlurImageOps.gaussian`:

```
// Тут і далі image - вихідне зображення, що має тип ImageUInt8  
ImageUInt8 blurred = new ImageUInt8(image.width,image.height) ImageUInt8 blurred  
= new ImageUInt8(image.width,image.height);  
BlurImageOps.gaussian(image,blurred,-1,5,null);
```

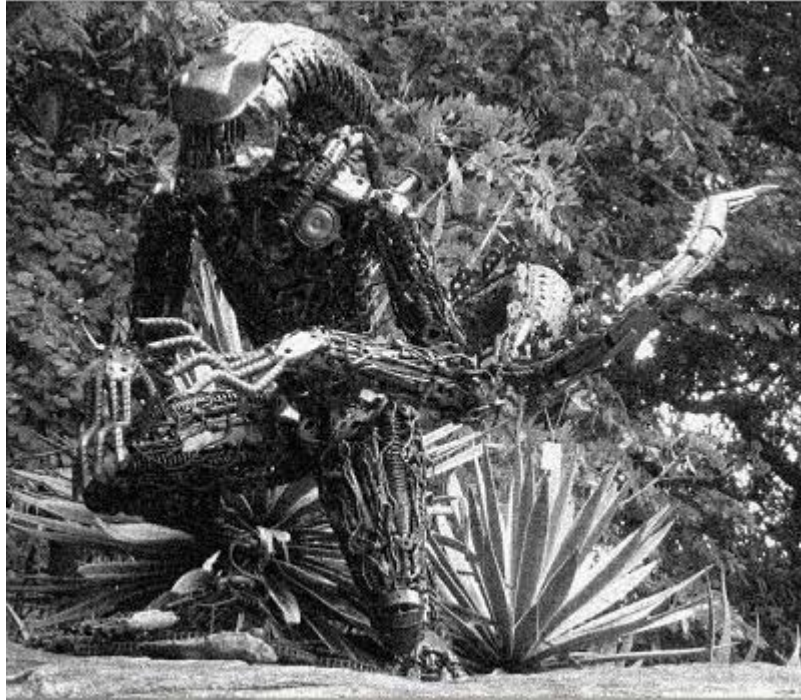


Рисунок 2.7 - зображення з шумом



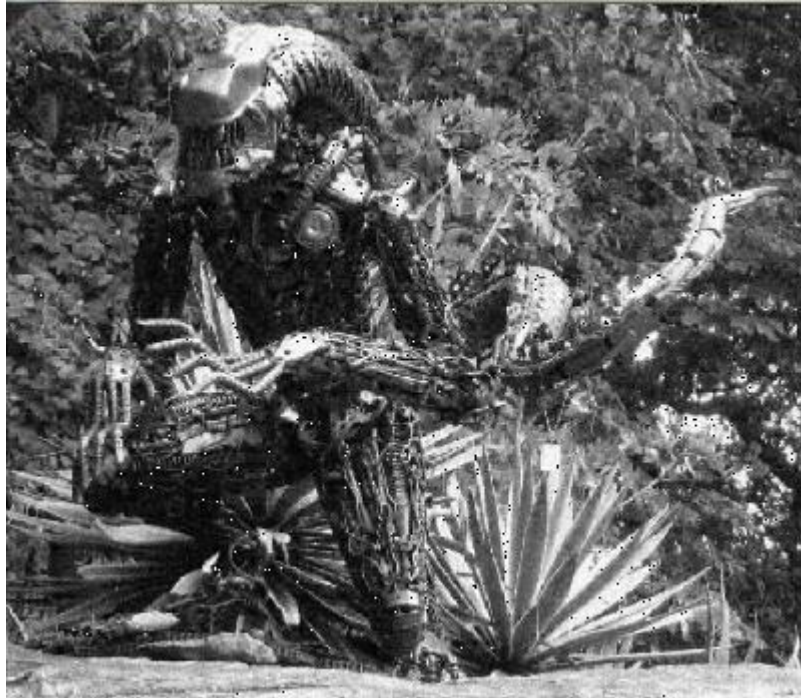


Рисунок 2.8 - зображення після фільтрування

Додаємо і прибираємо шум:

```
Random rand = new Random(234);  
    ImageUInt8noisy = image.clone();  
    GeneralizedImageOps.addGaussian(noisy,rand,20,0,255);  
    ImageUInt8denoised = new ImageFloat32(ImageUInt8.width,  
ImageUInt8.height);  
    int numLevels = 4;  
WaveletDenoiseFilter< ImageUInt8> denoiser =  
FactoryImageDenoise.waveletBayes(ImageUInt8.class,numLevels);
```



Рисунок 2.9 - виділення границь

Код:

```
ImageUInt8 canny = new ImageUInt8(image.width,image.height);
//Canny
    DetectEdgeContour<ImageUInt8> simple =
FactoryDetectEdgeContour.canny(30,200,false,ImageUInt8.class,ImageSInt16.class);
    simple.process(image);
    List<List<Point2D_I32>> edges = simple.getContours();
    for( List<Point2D_I32> l : edges ) {
        for( Point2D_I32 p : l ) {
            canny.set(p.x,p.y, 255);
        }
    }
}
```



Рисунок 2.10 - бінарзація зображення

Код:

```
ImageUInt8 binary = new ImageUInt8(image.width,image.height);  
float mean = PixelMath.sum(image)/(image.width*image.height);  
ThresholdImageOps.threshold(image, binary,mean,false);
```



Рисунок 2.11 - розмітка бінарізованого зображення

Код:

```
ImageSInt32 blobs = new ImageSInt32(image.width, image.height);  
int numBlobs = BinaryImageOps.labelBlobs4(binary, blobs);
```



Рисунок 2.12 - пошук точок зацікавленості

Для того, щоб використовувати бібліотеку в Android існує клас `ConvertBitmap`, що містить методи конвертації з формату Android у формат бібліотеки `VoofCV` і назад.

Наприклад так:

```
Bitmap map = BitmapFactory.decodeResource(getResources(), R.drawable.car5);
ImageUInt8 image = ConvertBitmap.bitmapToGray(map, (ImageUInt8)null, null);
```

Безумовно, це лише мала частина всіх можливостей бібліотеки. За функціоналом `VoofCV`, звичайно програє `OpenCV`. Однак, поріг входження в комп'ютерний зір для Android істотно знижується при використанні `VoofCV`. При цьому бібліотека продемонструвала непогану продуктивність, що дозволяє її використовувати в системах реального часу. Так пошук і локалізація прямокутника з потрібними параметрами (наприклад автомобільний номер) відбувається приблизно за 0.2 - 0.5 сек, залежно від величини картинки. Бібліотека до всього іншого непогано документована. [9]

## 2.4.5 OpenVX

Стандарт створений для того, щоб прискорювати алгоритми комп'ютерного зору насамперед на мобільних і вбудованих архітектурах.

OpenVX представляє з себе C API двох рівнів: immediate mode і graph mode. Перший - це окремі функції, за структурою дуже схожі на примітиви з OpenCV. У всіх функціях є еквівалент в OpenCV, здебільшого в модулі imgproc.

Верхній рівень дозволяє описати алгоритм комп'ютерного зору у вигляді орієнтованого графа, де кожен вузол відповідає функції. Для кожної функції immediate mode існує еквівалентний вузол графа. Граф повністю задається перед виконанням. Користувач може визначити свою функцію через C API callback і включити її в граф. Обидва рівні API працюють з контейнерами для зображень, внутрішня структура яких не специфікована.

Таким чином, дані контейнери є непрозорими (opaque), що дає більшу свободу реалізації. Розподіл зображень і вузлів графа по потоках і прискорювачів майже ніяк не управляється стандартом, і, відповідно, залишається на розсуд розробника, що реалізує стандарт. В рамках цієї архітектури можна зберігати зображення і виконувати функції OpenVX на прискорювачах, що може сильно прискорити алгоритм або зробити його більш енергоефективним. Взагалі, Graph API відкриває великий простір для оптимізацій. Наприклад, можна виконувати кілька операцій паралельно на різних ядрах або різних прискорювачах. За певних умов можна виконувати два послідовних вузла графа одночасно в одному потоці, виконуючи обчислення на льоту, або обробляти зображення по шматках (tiling), для того, щоб підвищити кеш-ефективність. Підтримка tiling для користувальницьких вузлів реалізована за допомогою розширення стандарту OpenVX 1.0 Tiling Extension - інтерфейс, затверджений групою OpenVX, але не обов'язковий до реалізації для OpenVX 1.0. Надалі, можливо, Tiling API стане обов'язковою частиною наступних версій стандарту.

OpenVX з самого початку розроблявся так, щоб ефективно взаємодіяти з OpenCV: обмін даними по можливості без накладних витрат, дуже близькі специфікації функцій. Можливо, надалі, OpenCV буде використовувати OpenVX для прискорення на ряді платформ.

## OpenVX and OpenCV are Complementary



	 OpenCV	 OpenVX
<b>Governance</b>	Open Source Community Driven No formal specification	Formal specification and conformance tests Implemented by hardware vendors
<b>Scope</b>	Very wide 1000s of functions of imaging and vision Multiple camera APIs/interfaces	Tight focus on hardware accelerated functions for mobile vision Use external camera API
<b>Conformance</b>	No Conformance testing Every vendor implements different subset	Full conformance test suite / process Reliable acceleration platform
<b>Use Case</b>	Rapid prototyping	Production deployment
<b>Efficiency</b>	Memory-based architecture Each operation reads and writes memory	Graph-based execution Optimizable computation, data transfer
<b>Portability</b>	APIs can vary depending on processor	Hardware abstracted for portability

Рисунок 2.13 Порвняння OpenVX та OpenCV

Дуже багато розробників технологій комп'ютерного зору зараз при роботі орієнтуються на одну платформу, вибираючи алгоритм і оптимізуючи його реалізацію для однієї системи. За умови широкої підтримки OpenVX для різних платформ, зусилля з адаптації алгоритму для іншої платформи будуть мінімальними. У найближчому майбутньому це призведе до зростання класних додатків комп'ютерного зору для мобільних телефонів і планшетів. У більш далекій перспективі OpenVX стимулюватиме виробників чіпів до створення прискорювачів, орієнтованих на OpenVX. Це приведе до радикального прискорення технологій комп'ютерного зору і створить умови для більш розумних

додатків, про які зараз ніхто навіть не мріє - такі, як змінена реальність на носимих пристроях (augmented reality on wearable devices) і функції безпечного водіння, доступні на автомобілях економ-класу .[20]

#### **2.4.6 ARToolkit**

ARToolkit - це бібліотека комп'ютерного стеження для створення додатків з доповненою реальністю. Для цього він використовує можливості відео спостереження, розрахунок реального стану та орієнтації камери по відношенню до квадратного фізичного маркеру в режимі реального часу. Коли реальний стан камери відомий, віртуальна камера може бути розташована в тій же точці, 3D модель накладається на реальний маркер. Так ARToolKit вирішує дві ключові проблеми доповненої реальності; відстеження погляду і віртуальної взаємодії об'єктів.

ARToolKit був спочатку розроблений Хироказу Като Нара інституту науки і технологій в 1999 році і був випущений в університеті Лабораторія НІТ Вашингтоні. В даний час він функціонує як проект з відкритим вихідним кодом на SourceForge з комерційною ліцензії доступні ARToolWorks. ARToolKit дуже широко використовується (бібліотека з більш ніж 160000 завантажень з 2004 року).

### **2.5 Висновки**

Існує багато бібліотек комп'ютерного зору, Які можна використовувати для проектів з використанням доповненої реальності. Основою майже усіх бібліотек є OpenCV, вони так чи інакше використовують деякі алгоритми цієї бібліотеки.



## **3 ОБЛАСТІ ЗАСТОСУВАННЯ ДОПОВНЕНОЇ РЕАЛЬНОСТІ**

### **3.1 Огляд сервісів доповненої реальності**

Хоча існує безліч інноваційних способів використання доповненої реальності, можна виділити чотири типи додатків, в яких найчастіше використовуються AR технології: рекламні та комерційні, розважальні та освітні, спеціалізовані та мобільне програми.

#### **3.1.1 Проект Araned**

Проект Araned, являє собою комплекс програмного забезпечення для вивчення анатомії на рівні вищої освіти з використанням технології доповненої реальності.

Фактично, автор відсканував реальні кістки і перевів їх в електронний атлас, щоб мати можливість роздруковувати муляжі на 3D-принтері.

Застосування технології доповненої реальності дозволяє наводити на надруковані муляжі свої смартфони, щоб вивчати судини, нерви, анатомічні утворення прямо поверх структури кістки.

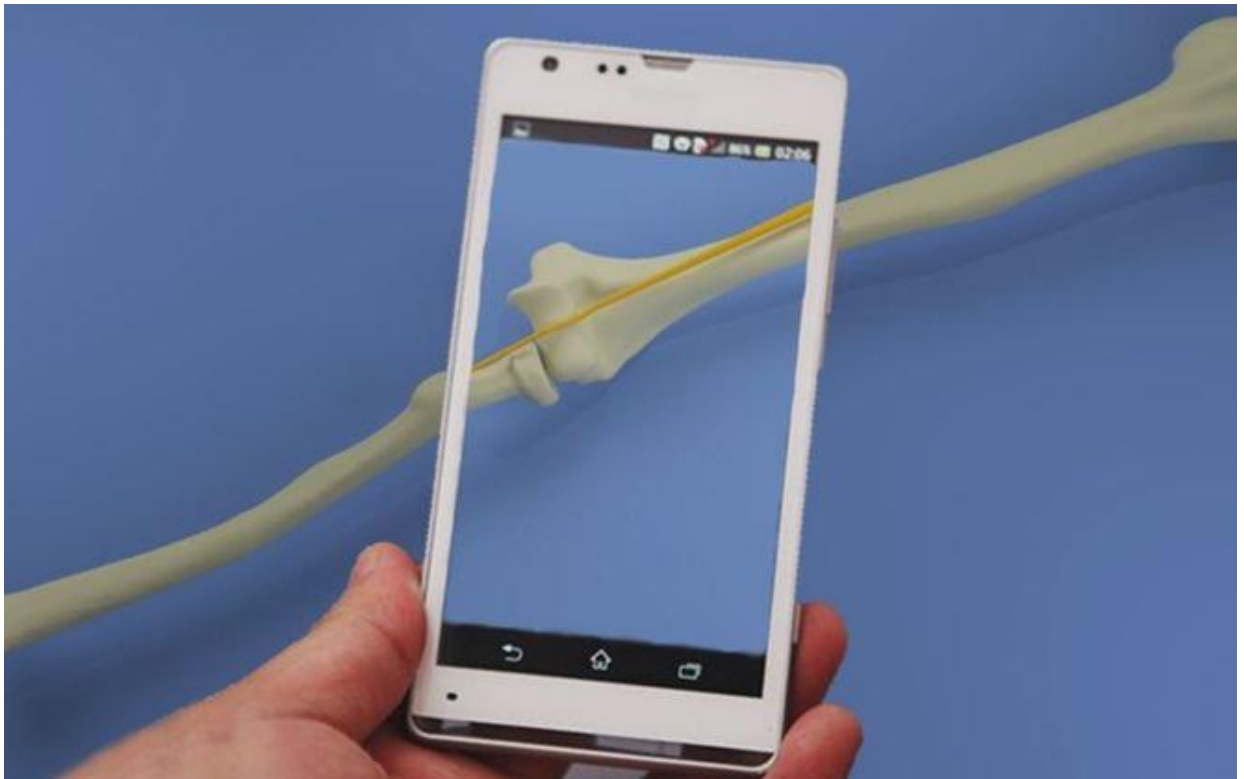


Рисунок 3.1 Макет майбутнього додатку

Програма дозволяє моделювати різні патологічні стани та оперативні втручання. Використовуючи систему Araped, користувачі можуть створювати спеціалізовані набори для хірургічних, травматологічних, гінекологічних та інших напрямів.

### 3.1.2 Мобільні AR додатки

Мобільні додатки доповненої реальності найчастіше є розважальними, освітніми, навігаційними або інформаційними.

Наприклад додаток WikitudeDrive, яке являє собою навігатор, який дозволяє користувачеві бачити дорогу під той час коли той дивиться маршрут на навігаторі (малюнок 9, ліворуч); Пожежний 360 - розважальна програма, що дозволяє користувачеві боротися віртуальним вогнем як справжньому пожежному (малюнок 9, справа), і Le Bar гід, навігаційний додаток, що допомагає користувачеві знайти найближчий бар, де подають пиво Stella Artois.[24]

У зв'язку з відносною новизною AR для мобільних додатків, в даний час з'являються бібліотеки для створення мобільних додатків доповненої реальності. Вони постійно розвиваються.



Рисунок 3.2 Приклади мобільних додатків. WikitudeDrive (ліворуч) і Пожежний 360 (праворуч)

Мобільні AR додатки є одними з небагатьох програм з доповненою реальністю, які можуть дійти до широкої публіки. Тим не менш, навіть ці додатки стикаються з деякими проблемами. Існують, проблеми, пов'язані з GPS-датчиками, які можуть бути не досить точними коли необхідно точно розмістити віртуальні теги. Є проблеми пов'язані з обмеженими можливостями апаратної частини, так як потрібна висока потужність для обробки зображень. Що стосується програмної сторони, існують проблеми з доступом до відео API, пов'язані із закритістю. Тим не менш, сучасні тенденції призводять до все більшої популяризації відкритих систем, що спрощує розробку додатків з доповненою реальністю, так як є повний доступ до пристрою.

### 3.1.3 Спеціалізовані додатки

Існує досить широкий спектр галузей науки і техніки, в яких може застосовуватися доповнена реальність. Однак, в першу чергу можна виділити такі з них:

- 1) Медицина;

2) Сервісні додатки

3) Картографія.

У медицині технології доповненої реальності затребувані для створення реалістичних тренажерів. Це дозволяє лікарям, практикуватися в проведенні, різного роду, хірургічних операцій на тренажері, що створює умови, максимально наближені до реальних, і тільки потім починати працювати з пацієнтами. При цьому інтерактивність і реалістичність тренажерів гарантуватимуть правильність дій лікаря при проведенні реальної операції. Як приклад може розглядатися система BoneSim дозволяє імітувати операції на кістковій тканині.[25]

Як приклад сервісних додатків можна привести систему компанії BMW для ремонту автомобілів.

У картографії доповнена реальність стала затребувана в зв'язку з широким розповсюдженням мобільних пристроїв оснащених великою кількістю датчиків. Так, подібні системи можуть ідентифікувати навколишні об'єкти, дозволяючи людині з легкістю орієнтуватися в просторі. Як приклад можна привести сервіс Layar дозволяє отримувати в реальному часі доступ до інформації про навколишній світ через камеру мобільного телефону. Це може бути інформація про кафе, ресторани, готелі і так далі.

### **3.1.4 Розважальні та освітні додатки**

Розважальна та освітня область включає в себе «культурні» додатки для огляду визначних пам'яток і музейні путівники, ігрові програми являють собою традиційні ігри з додаванням AR інтерфейсів. Так само в цю групу можна включити мобільні додатки, які використовують AR для розважальних або освітніх цілей.[26]

Серед «культурних» додатків, існує кілька систем, в яких AR використовується для реконструкції стародавніх руїн, наприклад, в (малюнок 3.3) або для надання віртуальної історичної довідки.



Рисунок 3.3 Реконструкція стародавніх руїн за допомогою AR

Є також кілька систем, які використовують AR для музейних путівників. Використовували переваги доповненої реальності, в якості інтерфейсу надає ефективну взаємодію з користувачем через мультимедійні презентації, природну і інтуїтивно зрозумілу техніку і низькі експлуатаційні витрати на придбання технології для музею у разі використання смартфона в якості інтерфейсу. І дійсно, використання смартфона або іншого мобільного пристрою - більш функціональний і досконалий метод, ніж пошук експоната за номером в путівнику, особливо коли майже у кожного відвідувача музею є свій власний мобільний пристрій (рисунок 3.4).

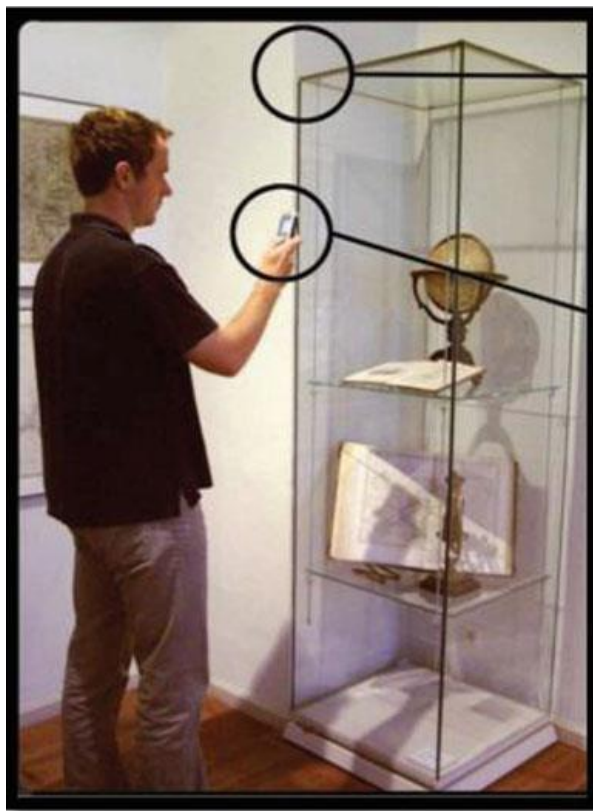


Рисунок 3.4 Мобільний AR додаток для музею

AR може також використовуватися для навчання. У галузі освіти системи доповненої реальності можуть мати широке застосування в багатьох областях, таких як історія, математика і т.д. Наприклад, Mark Billinghurst розробив Magic Book, книга, сторінки якої включені прості технології AR, для того, щоб зробити читання більш захоплюючим. Malaka створили мобільну систему доповненої реальності, використовуючи їх ранні розробки з проекту GEIST. Вони створили додаток для надання допомоги користувачам при вивченні історії за допомогою гри-розповіді, в якій користувач повинен звільнити привид з минулого.[27]

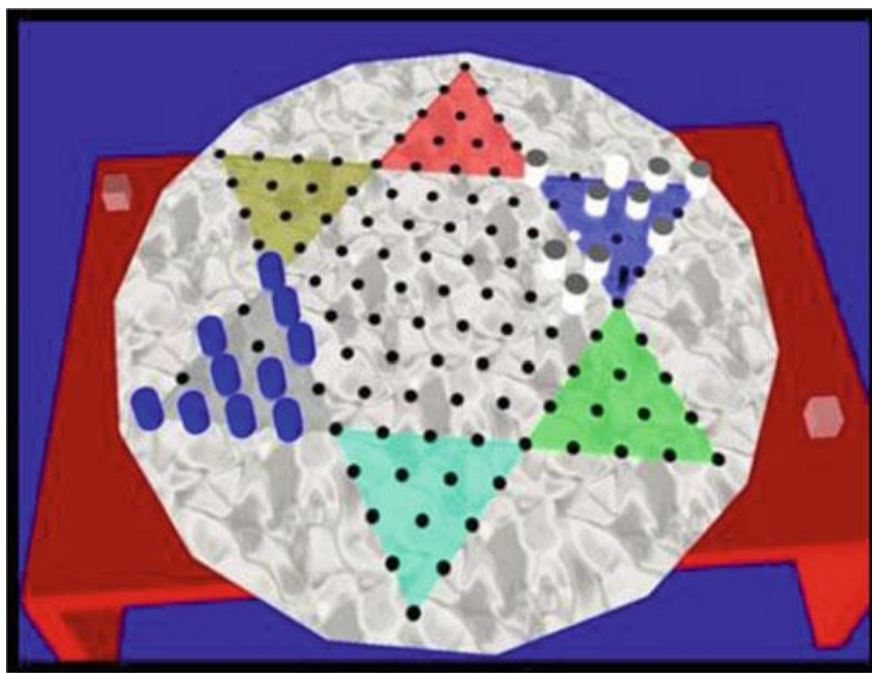


Рисунок 3.5 AR гра ARCC

Ігрові програми з доповненою реальністю мають багато переваг перед традиційними іграми. Наприклад, здатність введення анімації та інших мультимедійних ефектів може не тільки додати інтересу до гри, але також може служити для навчання гравців. В роботі, автори реалізували за допомогою доповненої реальності китайську гру в шашки, яка називається ARCC (рис. 3.5).

І тут, доповнена реальність не повною мірою досягла свого потенціалу, щоб вийти на ринок. Наприклад, наведені музейні системи наведення що були розроблені застосовується лише до конкретного музею або виставки. Це пов'язано з тим, що системи спиралися на навколишнє середовище музею або виставки для розпізнавання, на відміну від виявлення об'єктів виключно використовуючи алгоритми комп'ютерного зору. Як було видно з методів комп'ютерного зору в розділі AR, деякі об'єкти мають неправильну форму і, хоча, це може здатися легким для людини, але розпізнавання довільних образів дуже складний процес для комп'ютера.[28]

### 3.1.5 Рекламні та комерційні додатки

Доповнена реальність в основному використовується маркетологами для просування нових продуктів. Більшість додатків використовують маркери, які користувач розміщує перед камерою, спеціальне програмне забезпечення розпізнає його і доповнює будь-якою інформацією. Так, наприклад, у грудні 2008 року відома автомобільна компанія, MINI, використовувала доповнену реальність для реклами в декількох німецьких автомобільних журналах. Читач просто міг зайти на сайт MINI, помістити оголошення перед камерою, і 3-D модель MINI з'являлася на екрані, як показано на рисунку 3.6. Beyond Reality випустили рекламний журнал, сторінки якого є маркерами і можуть бути розпізнані спеціальним програмним забезпеченням на сайті, Обробка маркера є відправною точкою для гри в доповненій реальності. У наслідку в таку систему додаються "платні" варіанти програмного забезпечення, які дозволяють користувачеві отримати додаткові можливості.[23]



Рисунок 3.6 Рекламний додаток для MINI



Доповнена реальність також є вирішенням проблеми в побудові та поданні макетів. Дійсно, виробники стикаються з необхідністю, при дорогому виробництві продукту, виявляти, до комерціалізації, необхідні зміни або просто продемонструвати макет і визначити, чи відповідає продукт очікуванням. Якщо приймається рішення про внесення змін, зазвичай доводиться виготовляти новий прототип, що означає додаткові витрати часу і грошей. Група при Інституті Промислової Технологій та Автоматики (Itia) Національної Ради Досліджень (КНП) Італії в Мілані працює з AR і VR системами в якості інструменту для віртуального прототипування. Itia-CNR-бере участь у дослідженнях промислових систем і додатків, які за допомогою VR і AR, в реальному часі, використовують 3D-моделювання для тестування продуктів, їх розвитку та оцінки (рисунок 3.7). [29]



Рисунок 3.7 Віртуальний прототип фабрики

### 3.2 Висновки

Огляд існуючих додатків показав основні тенденції в розвитку технології доповненої реальності. Ця технологія в наш час дуже бурхливо розвивається і може знайти застосування в багатьох областях.

Аналіз додатків показав, що зараз вони мають, в основному, або розважальний, або вузькоспрямований характер. Основна увага в таких додатках приділяється їх можливостям при взаємодії з навколишнім середовищем і користувачем. Зважаючи на це, можна зробити висновок про актуальність розробки системи з розпізнаванням жестів, для взаємодії з віртуальними об'єктами.

Для створення функціонуючої системи необхідна досить потужна платформа, якою можуть бути сучасні мобільні пристрої, так як вони широко поширені і їх можливості постійно зростають, також з'являються окуляри доповненої реальності які в майбутньому будуть переважно використовуватись для цих цілей.

## 4 ВИКОРИСТАННЯ АЛГОРИТМІВ КОМП'ЮТЕРНОГО ЗОРУ В СИСТЕМАХ ДОПОВНЕНОЇ РЕАЛЬНОСТІ

### 4.1 Feature detection

Концепція feature detection виявлення в комп'ютерному зорі відноситься до методів, які націлені на обчислення абстракцій зображення і виділення на ньому ключових особливостей. Дані особливості можуть бути як у вигляді ізольованих точок, так і кривих або пов'язаних областей. Не існує строгого визначення того, що таке ключова особливість зображення. Кожен алгоритм розуміє під цим своє (кути, грані, області тощо).[42]

Найчастіше для пошуку маркерів використовуються алгоритми, які виконують пошук і порівняння зображень по ключових точках. Ключова точка - це деяка ділянка картинки, яка є відмітним для заданого зображення. Що саме приймається за дану точку - безпосередньо залежить від використовуваного алгоритму.

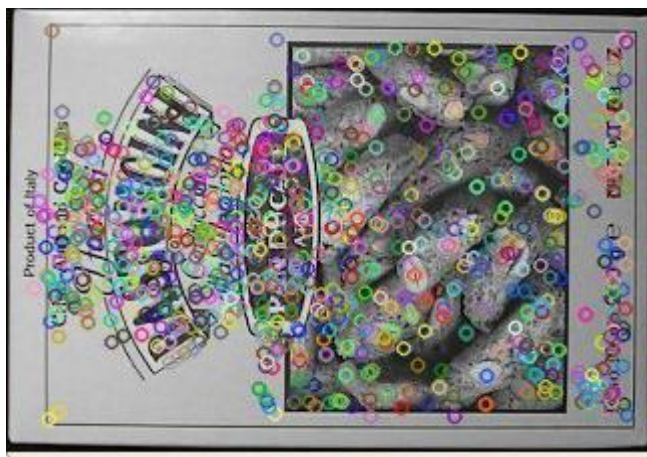


Рисунок 4.1 Приклад ключових точок на зображенні

Для їх знаходження та подальшого порівняння використовуються три складові:

- Детектор (англ. feature detector) — здійснює пошук ключових точок на зображенні.
- Дескриптор (англ. descriptor extractor) — виробляє опис знайдених ключових точок, оцінюючи їх позиції через опис навколишніх областей.
- Матчер (англ. matcher) — здійснює побудову відповідностей між двома наборами точок.

Спочатку за допомогою детектора проводиться пошук ключових точок шаблонного (шуканого) зображення. Отримані точки потім описуються за допомогою дескриптора. Дана інформація зберігається в окремий файл (або базу даних), щоб не виконувати цей процес повторно. При обробці відеопотоку з метою пошуку заданого шаблону описаний процес виконується для кожного кадру (за винятком збереження даних). Для встановлення відповідності між ключовими точками і дескрипторами застосовується матчер.[43]

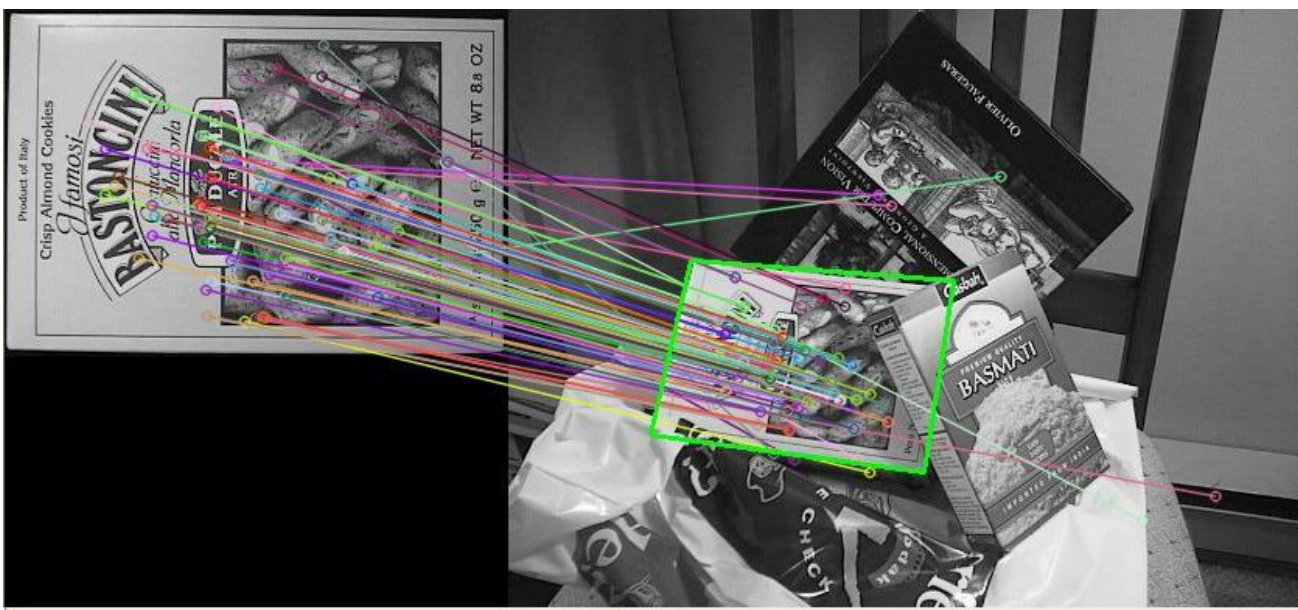


Рисунок 4.2 Відповідності між точками шаблону і тестованого зображення

Природно припустити, що різні алгоритми працюють з різною швидкістю та ефективністю. В умовах застосування їх для побудови доповненої реальності необхідно використовувати тільки ті, які показують високу швидкість роботи при досить хорошій якості відстеження позицій ключових точок. В іншому випадку ми можемо отримати помітні відставання у знімаємих відеоданих.

Для підвищення швидкості роботи алгоритмів feature points detection застосовуються різні способи фільтрації точок, щоб мінімізувати їх число і відсіяти зовсім погані поєднання. Таким чином, можна домогтися не тільки підвищення швидкості роботи алгоритмів, але і якості трекінгу маркерів.[44]

## **4.2 Генетичні алгоритми**

Генетичні алгоритми - це евристичні алгоритми пошуку, використовувані для вирішення завдань оптимізації та моделювання шляхом випадкового підбору, комбінування і варіації шуканих параметрів з використанням механізмів, що нагадують біологічну еволюцію.[47]

У комп'ютерному зорі вони використовуються для пошуку об'єкта деякого заданого класу на статичному зображенні або відеопотоці. Спочатку необхідно провести навчання алгоритму за допомогою двох різних наборів зображень:

- «Хороші» - містять потрібний об'єкт.
- «Погані» - помилкові зображення без шуканого об'єкта.

При цьому для навчання використовується велике число зображень, і чим їх більше - тим краще буде працювати сам алгоритм. Для кожної картинки виробляється виділення різних ключових особливостей: межі, лінії, центральні елементи.

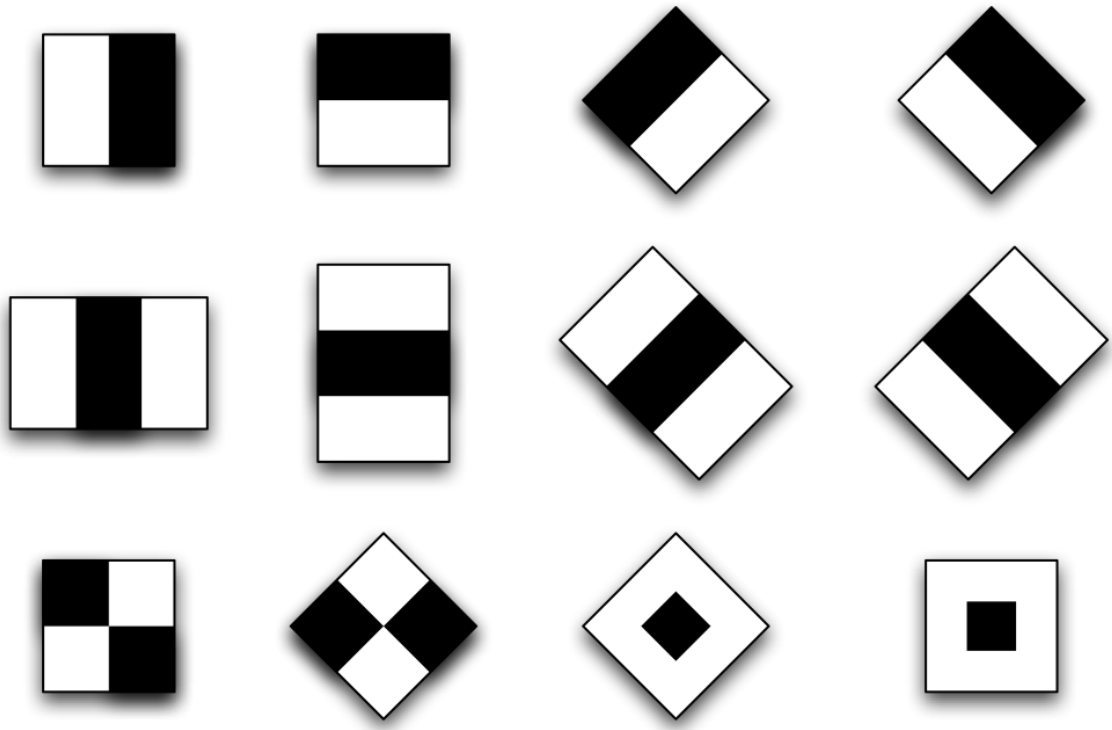


Рисунок 4.3 Примітиви Хаара, використовувані в алгоритмі

По ним проводиться побудова статистичної моделі, яка потім і використовується для пошуку об'єкта на зображенні.

Прикладом використання даного підходу може служити алгоритм розпізнання осіб і очей на відеопотоці. Поступово навчаючи алгоритм, можна домогтися високих результатів знаходження заданого класу об'єктів. Однак необхідність навчання якраз і робить використання генетичних алгоритмів досить проблематичним. Для їх гарної роботи потрібне істотне число різних зображень (як «хороших», так і «поганих»), і час побудови класифікатора для кожного об'єкта може займати тривалий час.[48]

### 4.3 Узагальнений алгоритм розпізнавання маркера

- а) Приводимо в градації сірого.
- б) Бінаризація зображення (поріг).

- c) Визначення замкнутих областей.
- d) Виділяємо контури.
- e) Виділяємо кути маркера.
- f) Перетворимо координати.

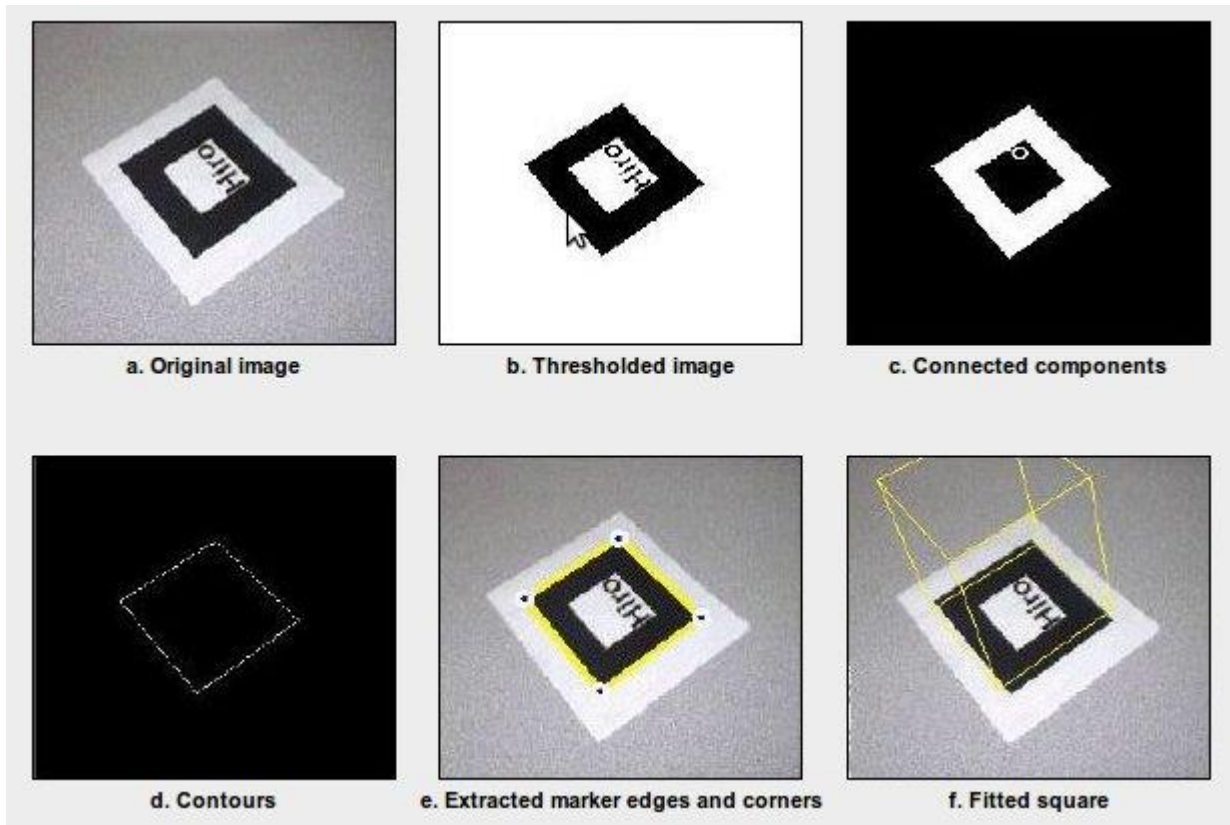


Рисунок 4.4 Узагальнений алгоритм розпізнавання маркера

#### 4.4 Переклад кольорового зображення в градації сірого

Наведемо три алгоритми перекладу кольорового зображення в градації сірого.

1. Світлота (Lightness)

$$GS = (\max(R,G,B) + \min(R,G,B))/2$$

2. Світимість (Luminosity)

$$GS = 0.21 \times R + 0.72 \times G + 0.07 \times B$$

3. Середнє (Average)

$$GS = (R + G + B) / 3$$

Ось приклад, як виглядають ці три способи. Як правило використовують “Світимість”. [49]

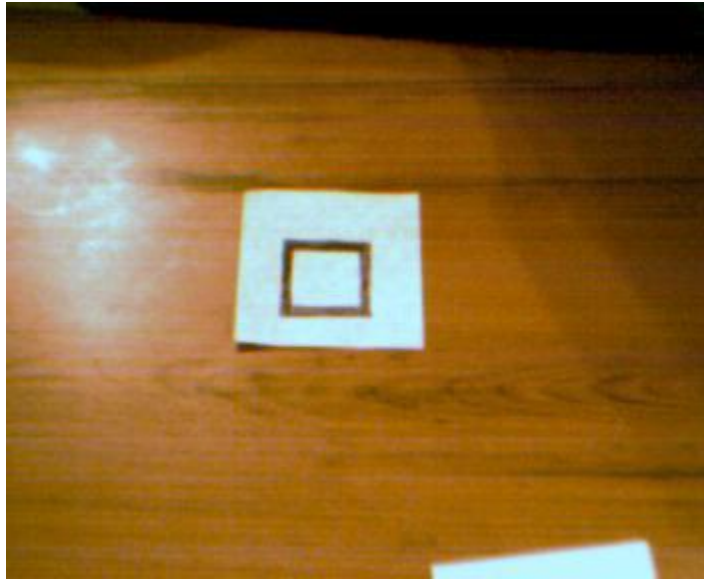


Рисунок 4.5 Початкове зображення

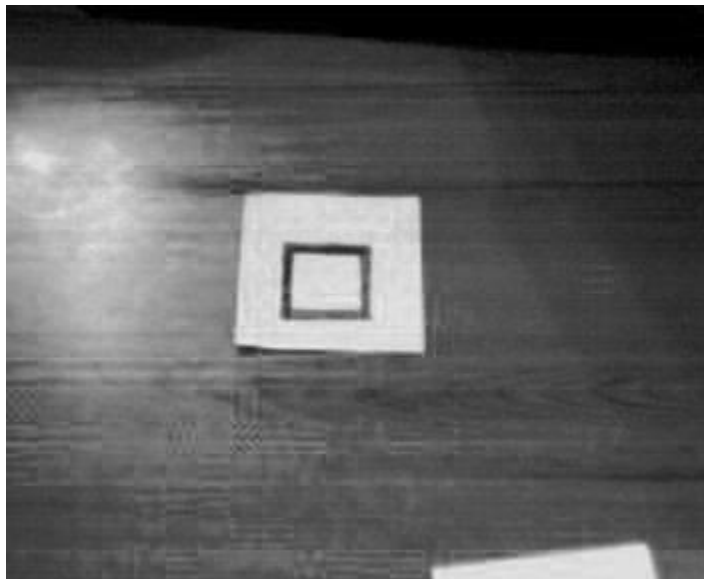


Рисунок 4.6 Світлота



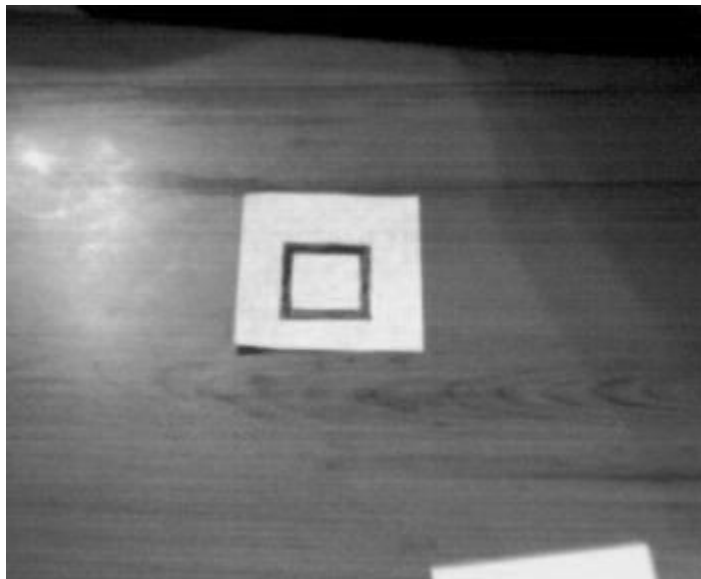


Рисунок 4.7 Світимість

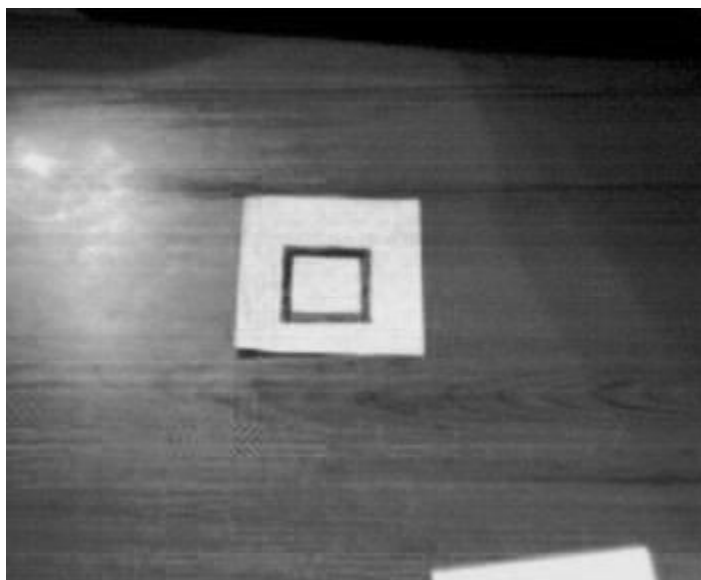


Рисунок 4.8 Середнє

## 4.5 Бінаризація зображення

Для перекладу зображення в двокольоровий стан використовується певний поріг. Питання в тому, як і до чого цей поріг застосовувати. Найпростіший спосіб - задати поріг. Наприклад якщо у нас 256 кольорів, то можемо задати поріг 128.

Якщо трохи ускладнити - то можемо вибрати поріг користуючись гистограммою кольору. Взагалі всі методи перетворення зображення в ч \ б вигляд можна розділити на шість великих груп які перераховані і класифіковані далі.[40]

- 1) Методи засновані на "формі" гістограми.
- 2) Методи на основі кластеризації.
- 3) Методи на основі вивчення ентропії.
- 4) Методи що базуються на пошуку подібності між сірим і ч / б зображенням.
- 5) Методи що використовують кореляційні залежності та особливості статистичного розподілу між пікселями в областях зображення.
- 6) Методи засновані на локальній адаптації порога для кожного пікселя зображення.

Найчастіше використовуються методи на основі локальної адаптації. Методів бінаризації близько сорока, якщо їх порівнювати, то вийде ще одна повноцінна стаття, так що наведемо приклад бінаризації зображення із заданим порогом, адаптивного, а також методу Оцу.[39]



Рисунок 4.9 Жорстко заданий поріг

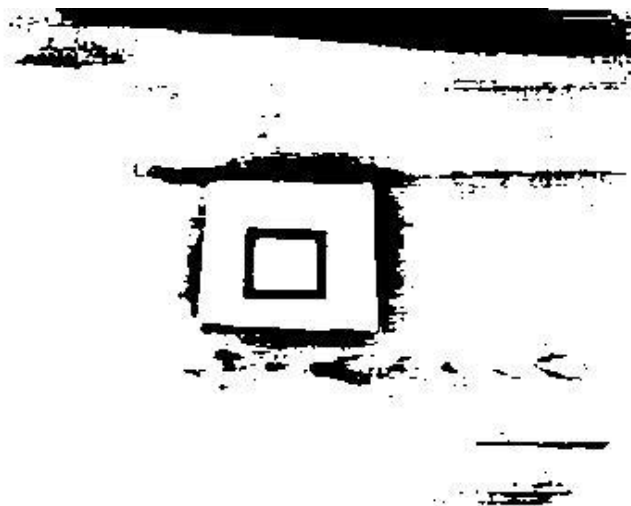


Рисунок 4.10 Адаптивный

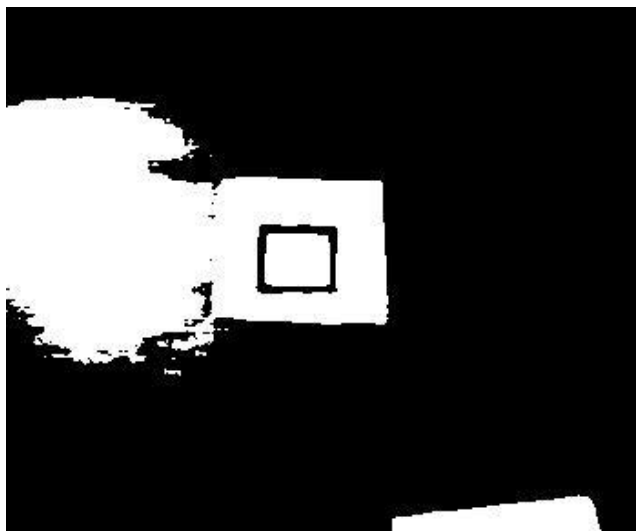


Рисунок 4.11 Метод Оцу

#### 4.6 Визначення замкнутих областей

Необхідно визначити замкнуті області на білому фоні. Як правило тут йде комбінація алгоритмів, але в загальному випадку - застосовують алгоритми "заливки" білих областей і виділяють замкнуті області. Спочатку виділяються контури, а потім перевіряються на замкнутість.[39]

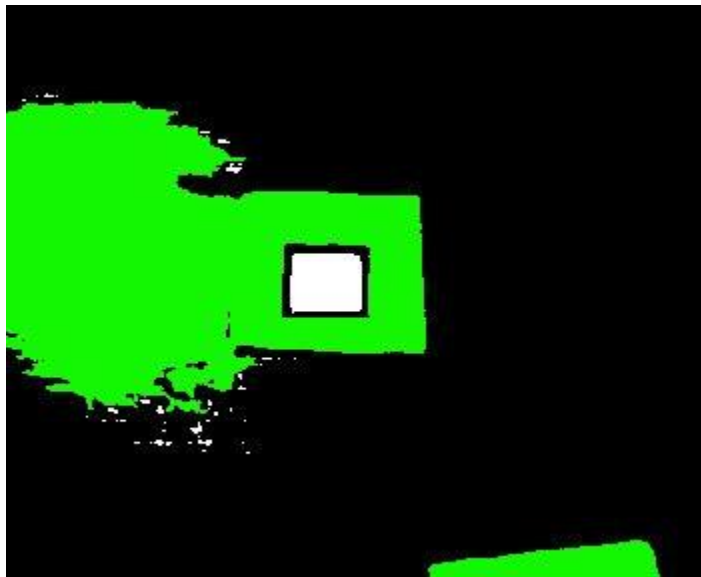


Рисунок 4.12 Перевірили на замкнутість



Рисунок 4.13 Спочатку виділили контури, потім перевірили на замкнутість

## 4.7 Виділення контуру

Існує кілька підходів до виділення контурів на зображенні. Зазначимо основні:

- 1) Marr-Hildreth Edge Detector
- 2) Canny Edge Detector
- 3) Boolean function based Edge Detector
- 4) Euclidian distance and Vector Angle based Edge Detector
- 5) Depth Edge Detection using Multi-Flash Imaging
- 6) Sobel Edge Detector

В основному використовується алгоритм Канні і Собеля. Всередині білих замкнутих областей шукаємо контури. В результаті застосування, отримуємо таке зображення: [38]

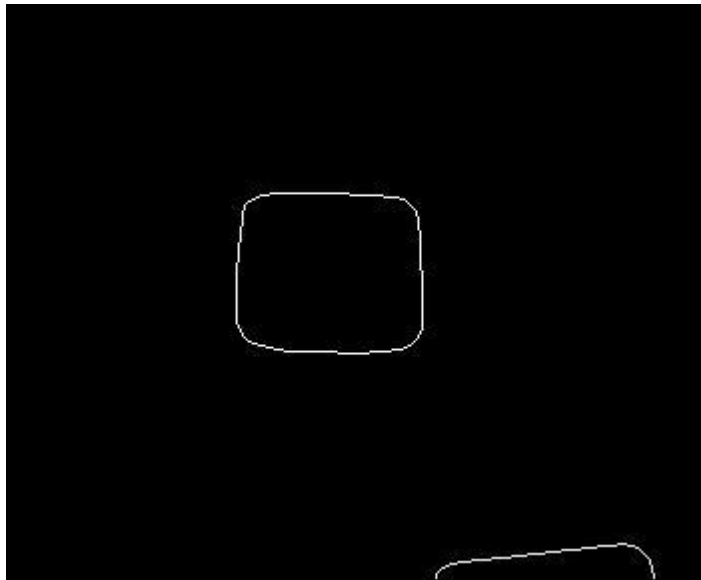


Рисунок 4.14 Канні

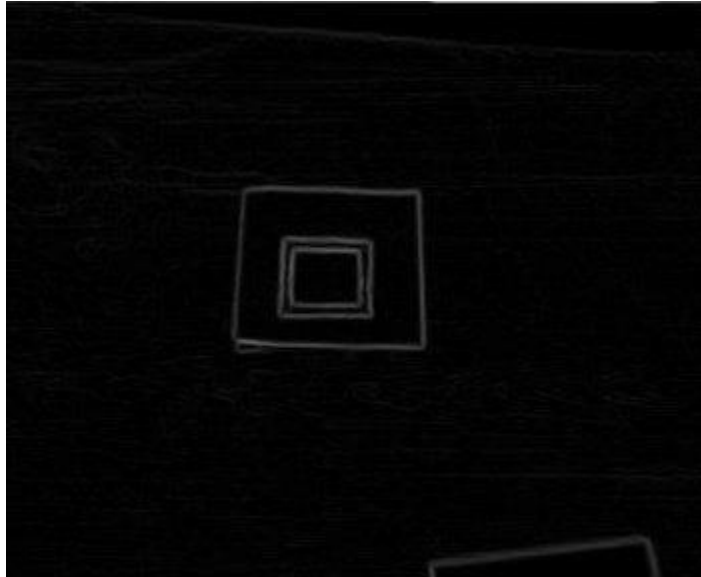


Рисунок 4.15 Собел

## 4.8 Виділення кутів маркера

Виділивши контур, його необхідно зіставити з нашим маркером. Може бути виділено багато різних контурів на зображенні, нам необхідно знайти щось «схоже» на чотирикутник.[32]

Для вирішення цього завдання можна застосувати алгоритм Дугласа-Пекера (він же алгоритм Рамера-Дугласа-Пекера, алгоритм ітеративної найближчої точки, алгоритм розбиття і злиття), що дозволяє зменшити кількість точок кривої, апроксимованої більшою серією точок.

Наприклад, в OpenCV є функція `approxPolyDP`, яка вже це робить. Якщо її результат додатково обробити, то вийде цілком підходящий результат.

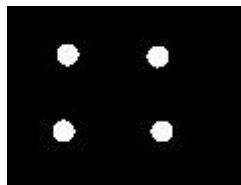


Рисунок 4.16 Координати кутів маркера

### 4.9 Перетворення координат

Тепер у нас є координати кутів маркера, які по суті, в ідеалі є перпендикулярними, а в реальності розташовані під іншим кутом. Крім того, і в ідеалі і в реальності, сторони квадрата є осями координат. Таким чином, ми можемо визначити положення «камери» щодо нашого об'єкта, і точку відліку початку координат.[34]

На чому заснований метод визначення координат можна показати графічно так:

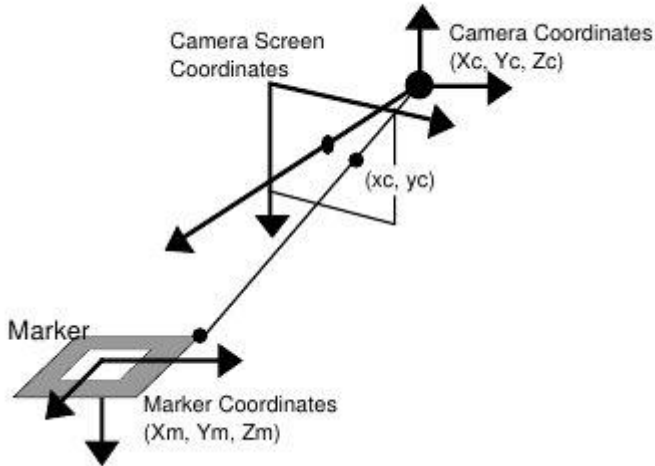


Рисунок 4.17

Тобто ідея полягає в тому, що при зміні кута з якого дивиться камера, змінюється розмір проекції, наприклад:

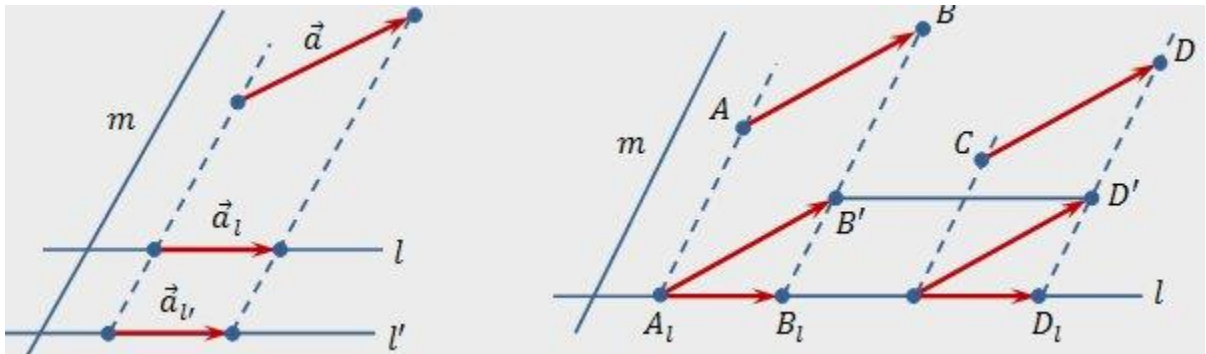


Рисунок 4.18

Знаючи положення камери і точки відліку, ми можемо намалювати проєкцію для нашої 3D моделі. В даному випадку куба.[36]

Як бачимо, якщо ми використовуємо квадрат з чистим полем в якості маркера, то він симетричний, і розпізнати обертання ми можемо тільки лише частково. Іноді цього достатньо. Якщо ж ні - то ми можемо внести всередину квадрата додатковий маркер, він тоді виглядатиме наприклад так і отримати в результаті ще й кут повороту, використовуючи перетворення Хафа.



Рисунок 4.19



## 4.10 Висновки

Існують два принципових підходи для створення доповненої реальності: з використанням заздалегідь приготованого маркера, який потрібно роздруковувати та без такого. Обидва підходи використовують алгоритми "комп'ютерного зору" розпізнають об'єкти в кадрі і доповнюють їх.

## 5 РОЗРОБКА ПРОТОТИПУ

### 5.1 Вибір бібліотеки для розробки

Перед початком розробки необхідно провести вибір бібліотеки.

#### 5.1.1 Огляд альтернатив

OpenCV - бібліотека алгоритмів комп'ютерного зору, обробки зображень та чисельних алгоритмів загального призначення з відкритим кодом. Реалізована на C / C ++, також розробляється для Python, Java, Ruby, MATLAB, Lua та інших мов. Може вільно використовуватися в академічних та комерційних цілях - поширюється в умовах ліцензії BSD.

Vuforia SDK - це програмне забезпечення для мобільних пристроїв, яке дозволяє створювати додатки доповненої реальності. Воно використовує технологію комп'ютерного зору для того, щоб розпізнавати і відстежувати плоскі зображення і прості 3D-об'єкти в режимі реального часу. Ця можливість реєстрації зображень дозволяє визначати розташування й орієнтації віртуальних об'єктів, таких як 3D-моделі, в реальному світу, коли вони розглядаються через камеру мобільного пристрою. Положення і орієнтація віртуального об'єкта відстежується в реальному часі, так що точки зору глядача на об'єкт співвідноситься з їх точкою зору на зображення, так що здається, що віртуальний об'єкт є частиною реальної сцени світу.

Vuforia SDK підтримує різні 2D і 3D цільові типи, включаючи безмаркерні мітки. Додаткові можливості SDK включає локалізовані виявлення оклюзії за допомогою "віртуальних кнопок, зображень виконання цільового відбору, а також можливість створити і змінити цільові набори програмно під час виконання.

Vuforia забезпечує API для C ++, Java, Objective-C. Є розширення ігрового движка Unity. Таким чином, SDK підтримує як рідні для IOS і Android-мови, так і

одночасно дозволяє розробляти додатки доповненої реальності в Unity, які легко портуються на обидві платформи. Додатків, розроблених з використанням Vuforia сумісні з широким спектром мобільних пристроїв, включаючи iPhone, iPad, Android і телефонів і планшетів під управлінням ОС Android версії 2.2 або вище і ARMv6 або 7 процесор з FPU.

Metaio SDK - готова бібліотека для створення мобільних додатків доповненої реальності. Використовує OpenGL використовує SLAM методи для більш точної роботи.

String – бібліотека для створення мобільних додатків орієнтована на IOS пристрої.

## 5.2 Проектування прототипу

Для роботи була вибрана бібліотека бібліотека JSARToolKit з WebRTC API getUserMedia для створення веб-додатків доповненої реальності. Для обробки було використано технологію WebGL через її високу ефективність. Кінцевим результатом є створення демонстраційної програми, яка розміщує тривимірну модель на маркері доповненої реальності у відео, що знімається веб-камерою.

JSARToolKit - це бібліотека доповненої реальності для мови JavaScript з відкритим вихідним кодом, випущена за ліцензією GPL у вигляді безпосереднього порту Flash-пакета FLARToolKit. Сама бібліотека FLARToolKit є портом Java-пакета NyARToolKit, який, у свою чергу, являє собою порт бібліотеки ARToolKit, написаної мовою C.

JSARToolKit працює з елементами canvas. Оскільки бібліотеці потрібно считувати зображення з об'єкта canvas, воно повинно знаходитися там же, де і сторінка.

Коли елемент canvas передається на аналіз у бібліотеку JSARToolKit, вона повертає список маркерів доповненої реальності, знайдених в зображенні, а також відповідних матриць перетворення. Щоб намалювати об'єкт поверх маркера, необхідно передати матрицю перетворення в будь-яку бібліотеку, використовувану для обробки тривимірних моделей(в нашому випадку це three.js, використовую WebGL), щоб з її допомогою трансформувати вибраний об'єкт. Потім достатньо намалювати відеокадр в сцені WebGL і розташувати об'єкт поверх нього.

### 5.2.1 Архітектура

Для функціонування даної системи необхідні наступні модулі:

#### 1) Камера

Компонент камери гарантує, що кожен перегляд кадрів фіксується і передається ефективно на трекер. Розробник тільки ініціалізує камеру, щоб почати і зупинити транслявання. Кадр камери автоматично конвертується в апаратно-залежний формат і задає потрібний розмір зображення.

#### 2) Конвертер зображення.

Конвертор форматів перетворює кадри з камери формату у формат, придатний для рендеринга WebGL і для відстеження. Це перетворення також включає в себе зменшення зображення з камери в різних дозволах доступних в перетвореному стеку кадрів.

#### 3) Трекер

Компонент трекер містить алгоритми комп'ютерного зору, для того щоб виявляти і відстежувати об'єкти реального світу в рамках відеокамери. На основі зображення з камери, різні алгоритми піклуються про виявлення нових цілей, або маркерів. Трекер може завантажити декілька наборів даних одночасно і активувати їх.

#### 4) Відрисовка текстури

Цей модуль візуалізації створює зображення, що зберігається в об'єкті.

#### 5) Код додатку

У коді повинні бути ініціалізовані всі перераховані вище компоненти і виконані три умови. Для кожного обробленого кадру об'єкт оновлюється і викликається метод відтворення. необхідно:

- Визначати об'єкти для знову виявлених цілей, маркерів або оновлених станів цих елементів;
- Оновлювати логіку програми з новими вхідними даними;
- Малювати фігуру квадрат з накладеною на неї текстурою зображенням.

#### 6) База маркерів

Створення об'єкту маркерів, в якому вказати маркери які буде розпізнавати додаток, в нашому випадку їх 3.

На рисунку 15 представлена архітектура прототипу.

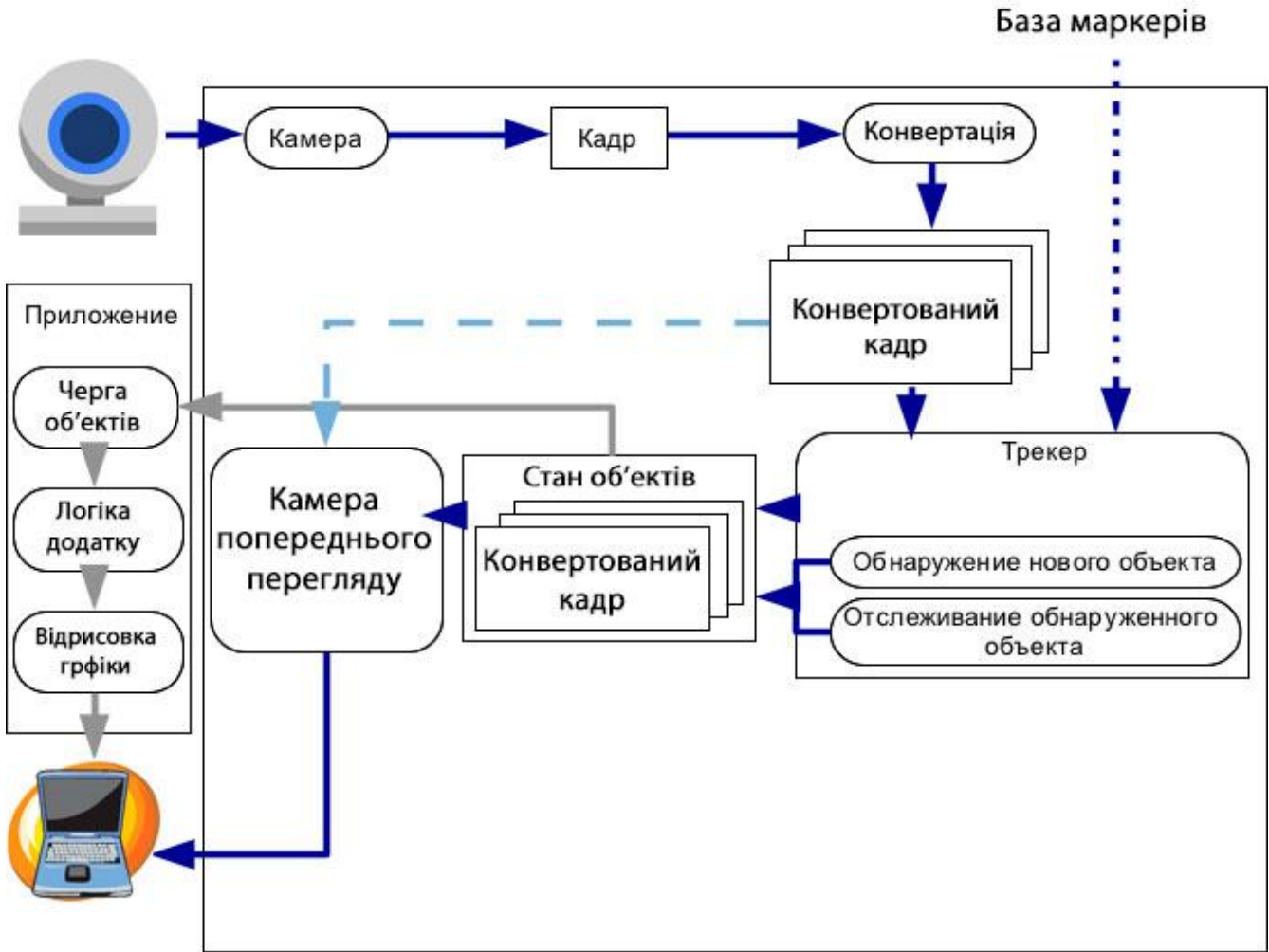


Рисунок 5.1 Архітектура прототипу системи

### 5.2.2 Розробка системи

#### 1) Визначення маркера

JSARToolkit - движок для доповненої реальності. Він визначає і описує становище координатних маркерів в спеціальних зображеннях. Кожен маркер прив'язаний до унікального номеру (ID) в движку. В нашому випадку були взяті маркери 2,3 та 5.

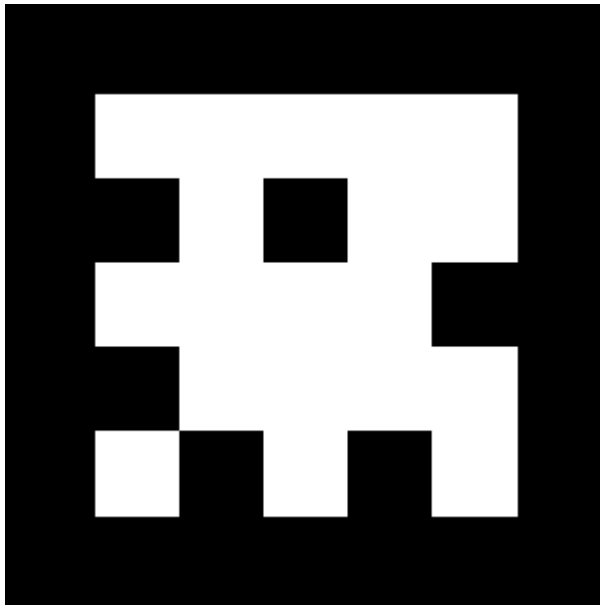


Рисунок 5.2 Маркер в бібліотеці JSARToolKit з id 2

Функціонал визначення маркера:

// Об'єкт raster - це canvas, в який ми копіюємо кадри відео.

```
var JSARRaster = NyARRgbRaster_Canvas2D(canvas);
```

// Об'єкт parameters позначає розміри вхідної відеотрансляції в пікселях.

```
var JSARParameters = new FLARParam(canvas.width, canvas.height);
```

// MultiMarkerDetector - движок для розпізнавання маркера

```
var JSARDetector = new FLARMultiIdMarkerDetector(FLARParameters, 120);
```

```
JSARDetector.setContinueMode(true);
```

2) Запуск детектора для кадру, який повертає кількість розпізнаних маркерів

Після обробки кадру за допомогою JSARDetector.detectMarkerLite (), об'єкт JSARDetector буде містити список розпізнаних маркерів. JSARDetector.getIdMarkerData (index) повертає номер ID, а JSARDetector.getTransformMatrix

(index) повертає положення в просторі. Користуватися цими методами незручно, тому ми обернєм їх у допоміжні методи і викличемо з циклу таким чином:

```
var markerCount = JSARDetector.detectMarkerLite (JSARRaster, 90);

for (var index = 0; index <markerCount; index ++) {
    // Отримання унікального номера розпізнаного маркера.
    var id = getMarkerNumber (index);

    // Отримання матриці переходу для розпізнаного маркера.
    var matrix = getTransformMatrix (index);
}
```

Оскільки детектор оперує кожним кадром окремо, відповідальність за стан маркера в проміжках між кадрами лягає на нас. Наприклад, між двома послідовними кадрами може трапитися що-небудь з нижченаведеного:

- може бути розпізнаний новий маркер;
- існуючий маркер може змінити розташування;
- існуючий маркер може зникнути з потоку.

Відстеження стану реалізується за допомогою ardetector.js. Для цього ми створюємо копію canvas, в який передаються кадри з відео:

```
// Створюємо детектор маркера AR, використовуючи canvas як джерело даних

var detector = ardetector.create (canvas);
```

З кожним запитом tick () зображення в canvas сканується детектором і при необхідності запускається callback-функція:



// Розпорядження для детектора виконати розпізнання.

detector.detect (onMarkerCreated, onMarkerUpdated, onMarkerDestroyed);

Тепер наш додаток розпізнає маркери і виводить результати розпізнавання в консоль.

### 3) Результат

В результаті був створений веб-додаток, що може розпізнавати маркери, та виводити замість них додаткову інформацію про об'єкт (рисунок 5.3).

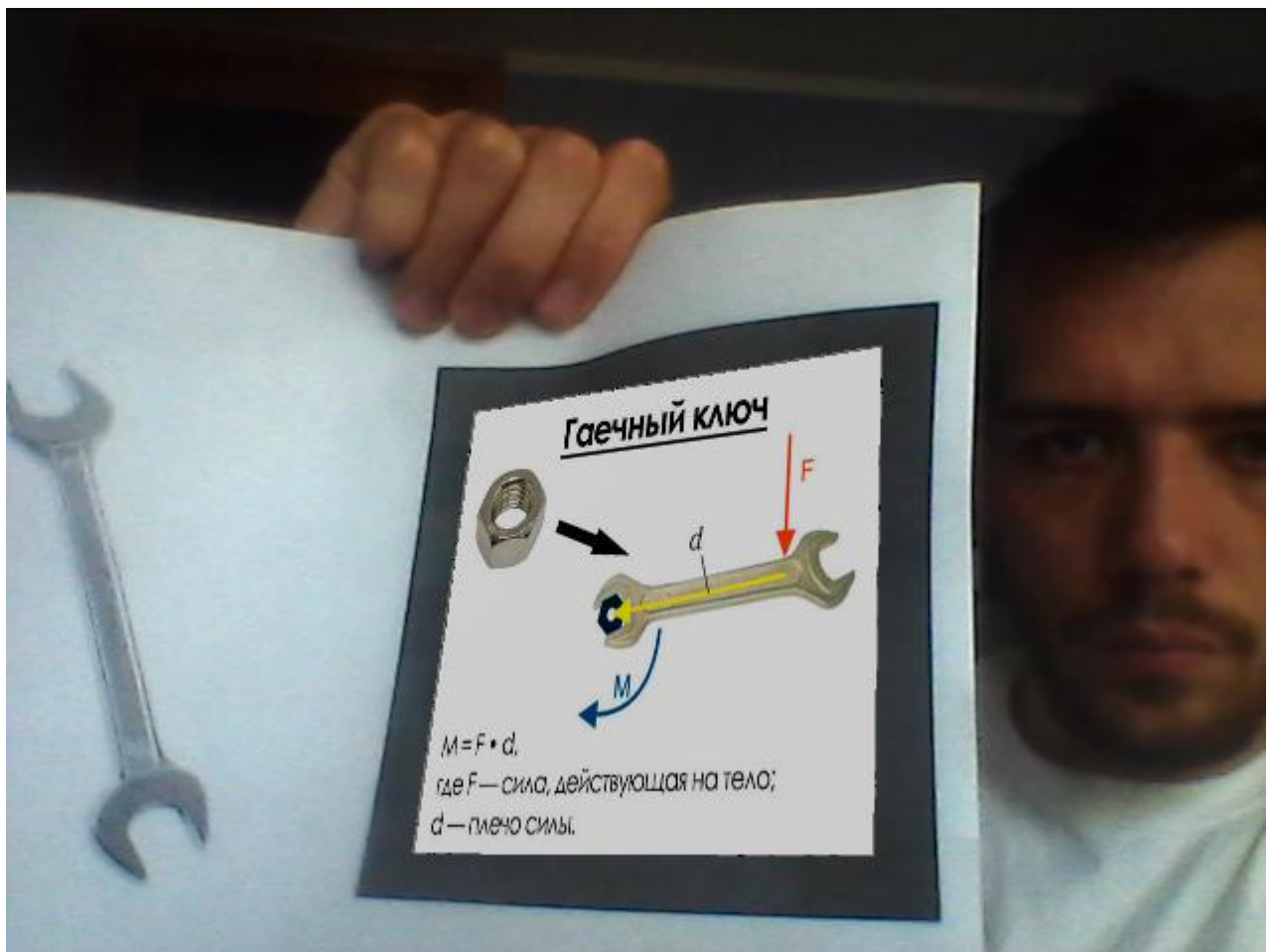


Рисунок 5.3 Розпізнаний маркер

### 5.2.3 Надійність

Надійність - властивість об'єкта зберігати в часі у встановлених межах значення всіх параметрів, що характеризують здатність виконувати необхідні функції в заданих режимах і умовах застосування, технічного обслуговування, зберігання і транспортування. Надійність є комплексною властивістю, що у залежності від призначення об'єкта та умов його застосування може включати безвідмовність, довговічність, ремонтпридатність і збереженість або певні поєднання цих властивостей.[41]

Надійність ПО визначається в першу чергу його безвідмовністю. Безвідмовність ПО - це властивість зберігати працездатність при використанні його для обробки інформації. Безвідмовністю програмного забезпечення оцінюється ймовірність його роботи без відмов при певних умовах зовнішнього середовища протягом заданого періоду спостереження.

У наведеному визначенні під відмовою ПО розуміється недопустиме відхилення параметрів функціонування цього забезпечення від пропонованих вимог. Заданий період спостереження відповідає часу, необхідного для виконання на ЕОМ розв'язуваної задачі.[41]

Безвідмовність ПО може характеризуватися середнім часом виникнення відмов при функціонуванні програми. При цьому передбачається, що апаратні засоби ЕОМ знаходяться в справному стані. З точки зору надійності, принципова відмінність ПЗ від апаратних засобів полягає в тому, що програми не зношуються і їх вихід з ладу через несправність неможливий. Отже, характеристики функціонування ПО залежать тільки від його якості, зумовлює процесом розробки. Це означає, що безвідмовність ПО визначається його коректністю і залежить від наявності в ньому помилок, внесених на етапі його створення. Крім того, прояв помилок ПО пов'язано ще і з тим, що в деякі моменти часу на обробку

можуть надходити не звичні сукупності даних, які програма не в змозі коректно обробити. Тому вхідні дані певною мірою впливають на функціонування ПЗ.

Таким чином, основними показниками надійності ПЗ є:

ймовірність безвідмовної роботи програми  $P(t)$ , що представляє собою ймовірність того, що помилки програми не проявляться в інтервалі часу  $(0, t)$ ;

ймовірність відмови програми  $Q(t)$  або ймовірність події відмови ПО до моменту часу  $t$ ;

середнє напрацювання програми на відмову  $T$ .

При визначенні характеристик надійності ПЗ враховується той факт, що виникаючі при роботі програм помилки усуваються, кількість помилок зменшується і, отже, їх інтенсивність знижується, а напрацювання на відмову програми збільшується.

Для програмного забезпечення відсутня фаза переробки та зносу, і воно завжди функціонує в періоді нормальної експлуатації.

Експоненціальна модель надійності ПЗ заснована на припущенні про експоненційному характері зміни числа помилок в часі. У період нормальної експлуатації інтенсивність відмов не змінюється.

Середнє напрацювання на відмову розраховується наступним чином:

$$T_0 = \frac{1}{\lambda_{ПО}},$$

де  $\lambda_{ПО}$  - інтенсивність помилок програмного забезпечення.

Інтенсивність помилок розроблюваного програмного забезпечення розраховується за формулами:

$$\lambda_{ПО} = \lambda_0 * e^{-\alpha t},$$

$$\lambda_0 = \frac{N_0}{N \cdot t} \cdot K_{ТП} = \frac{N_0}{N \cdot t} \cdot K_{ТПі} \cdot K_{ЯЗі} \cdot K_{ПЛі},$$

де  $t$  - фактичний час налагодження;

$\alpha$  – коефіцієнт крутизни лінії, що характеризує швидкість росту надійності;

$N_0$  – число виявлених помилок за час налагодження  $t$ ;

$N$  – загальне число рядків;

$K_{ТП}$  – коефіцієнт, що враховує вплив методології програмування на надійність ПЗ;

$K_{ТПi}$  – коефіцієнт, що враховує вплив методології програмування на Надійність ПЗ;

$K_{язi}$  – коефіцієнт, що враховує використання  $i$ -ого мови програмування;

$K_{ПЛi}$  – коефіцієнт, що враховує використання  $i$ -ой платформи програмування.

У даному програмному забезпеченні використана об'єктно-орієнтована технологія програмування ( $K_{ТПi} = 0,1$ ), мова JavaScript ( $K_{ВСi} = 0,74$ ) на 64-розрядній платформі (для платформ, що не є .NET,  $K_{ПЛi} = 2$ ).

$$K_{ТП} = 0,1 \cdot 0,74 \cdot 2 = 0,15$$

Початкове число рядків коду  $N = 5243$ .

Результати налагодження ПО представлені в таблиці 5.1.

Таблиця 5.1 Результати налагодження ПО

Число помилок	Час відладки $t$ , години	Інтенсивність помилок, $\lambda_0$ 1/година
23	4	0,00006
17	4	0,00004
14	4	0,00003
5	4	0,00002
4	4	0,00001
2	4	0,00001
1	4	0,00000

На основі отриманих даних можна побудувати криву залежності інтенсивності помилок від часу налагодження (рисунок 5.4).

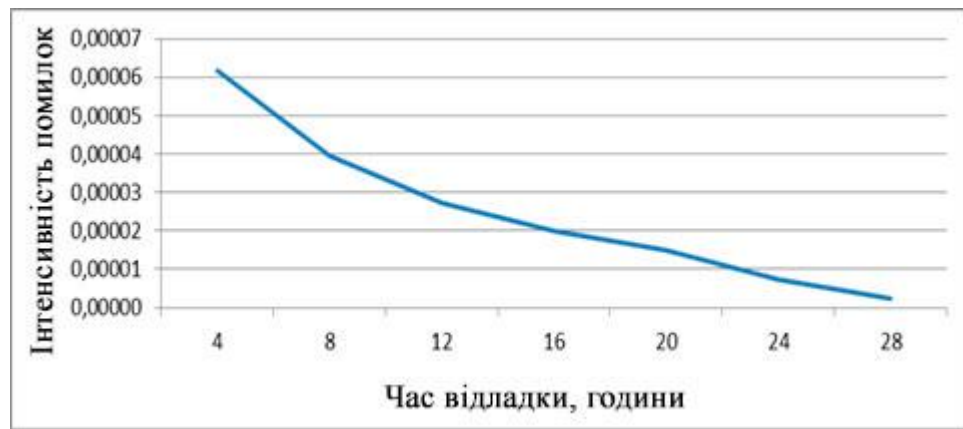


Рисунок 5.4 Залежність інтенсивності помилок від часу налагодження

Як показано вище, функціональна залежність інтенсивності помилок ПЗ від часу налагодження описується експоненціальним законом і залежить від коефіцієнта крутизни лінії, що характеризує швидкість росту надійності  $\alpha$ , і від фактичного часу налагодження ПЗ. Аналіз результатів тестування ПЗ дозволив визначити  $\alpha = 0,15$ .

Таким чином, інтенсивність помилок розроблювального ПЗ становить:

$$\lambda_0 = \frac{N_0}{N \cdot t} \cdot K_{\text{ТП}} = \frac{23}{5243 \cdot 4} \cdot 0.15 = 0.0006 \quad (1/\text{час})$$

$$\lambda_{\text{ПО}} = \lambda_0 \cdot e^{-\alpha t} = 0,0006 \cdot e^{-0,15 \cdot 4} = 0,00033 \quad (1/\text{час})$$

За експоненціальним законом ймовірність безвідмовної роботи:

$$P(t) = e^{-\lambda \cdot t}$$

Напрацювання на відмову програмного забезпечення становить:

$$T_0 = \frac{1}{\lambda_{\text{ПО}}} = \frac{1}{0,00033} = 3030 \quad (\text{ч})$$

Імовірність безвідмовної роботи становить:

$P = 0.9987$ .

### **5.3 Висновки**

Технології доповненої реальності можна застосовувати у різних областях, в даній роботі були приведені можливості застосування доповненої реальності в медицині, навчанні та в e-commerce. Також був проведений аналіз надійності робот веб-додатку.

## **6 ОХОРОНА ПРАЦІ**

### **6.1 Вступ**

У даному розділі вирішується питання безпечної життєдіяльності середовища в якому розроблювалася підсистема на основі санітарних норм України.

На робочому місці працівника можуть виникнути шкідливі та небезпечні ситуації, наприклад: висока напруга електричної мережі, шкідливі речовини в повітрі, поганий рівень освітлення, високий рівень шуму та інше. Робота з комп'ютером також супроводжується високим рівнем напруженості трудового процесу. Також важливо правильно організувати робоче місце, так як неправильна організація сприяє напрузі верхніх кінцівок, розвитку остеохондрозу, та погіршенню самопочуття користувача в цілому.

У дипломній роботі на тему: “Алгоритми комп'ютерного зору в системах доповненої реальності” питання охорони праці розглядаються стосовно до роботи користувача ПЕОМ.

### **6.2 Характеристика приміщення**

- Кількість працюючих: 3.
- Довжина приміщення: 7 м.
- Ширина приміщення: 4 м.
- Висота приміщення: 3 м.
- Загальна площа приміщення: 28 м<sup>2</sup>.
- Площа на одне робоче місце: 9.3 м<sup>2</sup>.
- Об'єм приміщення: 84 м<sup>3</sup>.
- Об'єм на одне робоче місце: 28 м<sup>3</sup>.

Вимоги, яким повинні задовольняти робочі приміщення, приведені в ДСанПіН 3.3.2.007-98:

- обсяг приміщення не менше  $20 \text{ м}^3$  на людину;
- площа приміщення не менш  $6 \text{ м}^2$  на людину.

Отже, приміщення відповідає вимогам.

### **6.3 Мікроклімат робочого місця користувача ПК**

Для постійного робочого місця працівника користувача ПК, є оптимальні параметри мікроклімату яких слід дотримуватись, при неможливості дотримання цих параметрів використовуються допустимі норми.

Мікроклімат виробничих приміщень - це сукупність параметрів повітря у виробничому приміщенні, які діють на людину у процесі праці, на його робочому місці, у робочій зоні.

Робоче місце - територія постійного або тимчасового знаходження людини у процесі праці.

Робоча зона - частина простору робочого місця, обмежене по висоті 2 м від рівня підлоги.

Параметри мікроклімату:

- 1) температура повітря  $T$ ,  $^{\circ}\text{C}$ ;
- 2) відносна вологість  $Y$ , %;
- 3) швидкість руху повітря  $V$ ,  $\text{м}/\text{с}$ .

Метеорологічні умови роботи на обчислювальному центрі вибираються відповідно до вимог ГОСТ 12.1.005-88 з урахуванням категорії робіт за енерговитратами для теплого й холодного періодів року.

У таблиці 6.1 представлені оптимальні значення метеорологічних умов роботи оператора ПЕОМ з урахуванням категорії важкості робіт -  $I_a$



(енерговитрати організму до 139 Вт, здійснюється сидячи й не потребуючої систематичної фізичної напруги або підняття й перенесення ваг).

Таблиця 6.1 - Значення оптимальних параметрів метеорологічних умов

Період року	Категорія ваги робіт	Температура, °C	Відносна вологість, %	Швидкість руху повітря, м/с, не більше
Холодний	Легка, I <sub>a</sub>	22-24	40-60	0,1
Теплий	Легка, I <sub>a</sub>	23-25	40-60	0,1

Для забезпечення оптимальних параметрів мікроклімату в холодний період року здійснюються опалення від центральної тепломережі, у теплий період року використовуються вентилятори та проводиться постійне провітрювання.

В кімнаті не встановлено кондиціонер. Проведемо аналіз потужності за охолодженням з урахуванням необхідної продуктивності за об'ємом повітря, що подається у приміщення. Розрахунок продуктивності кондиціонера в робочих приміщеннях можна провести за кількістю працюючих у приміщенні з урахуванням нормативних вимог (ДСанПіН 3.3.2.007 – 98).

Потужність (точніше, потужність охолодження) є основною характеристикою будь-якого кондиціонера. Від цієї величини залежить площа, на яку він розрахований. Розрахуємо надлишки явного тепла. Спочатку слід розрахувати теплопритоки.

Розрахунок будемо проводити за формулою

$$Q = Q_{\text{сонця}} + Q_{\text{комп'ютерів}} + Q_{\text{людей}} = q * V_{\text{приміщення}} + 0.3 \text{ кВт} * n_{\text{комп'ютерів}} + 0.1 \text{ кВт} * n_{\text{людей}} \quad (6.1)$$

У розрахунках враховується тепло, яке виділяється людьми і електроприладами. Вважається, що у спокійному стані людина виділяє 0,1 кВт тепла; комп'ютер або копіювальний апарат – 0,3 кВт; для інших приладів можна

вважати, що вони виділяють кількість тепла, що дорівнює 1/3 паспортної потужності [49]. Просумувавши всі виділення тепла і теплопритоки, ми отримаємо потрібну потужність охолодження. Так як вікна в кімнаті виходять на схід (величина  $q$  становить 30 Вт/м. куб.), то загальна величина теплопритоку буде обчислюватися за формулою (6.1):

$$Q = 30 \cdot 84 + 300 \cdot 4 + 100 \cdot 4 = 4120 \text{ Вт},$$

Кондиціонер повинен мати охолоджувальну потужність не менше:

$$Q_{\text{сум}} = 1.2 \cdot Q \quad (6.2)$$

У нашому випадку за формулою (4.2)

$$Q = 1.2 \cdot 4120 = 4944 \text{ Вт}.$$

Таким чином, для забезпечення належного охолодження в приміщенні рекомендовано встановити кондиціонер потужністю не менш, як 4,94 кВт.

В робочому приміщенні знаходилося три робочих місця. З обладнанням в кімнаті розміщено три комп'ютери на базі процесора *Intel Core 2 Duo*.

#### **Характеристики приміщення:**

- Кількість працюючих: 3.
- Довжина приміщення: 7 м.
- Ширина приміщення: 4 м.
- Висота приміщення: 3 м.
- Загальна площа приміщення: 28 м<sup>2</sup>.
- Площа на одне робоче місце: 9.3 м<sup>2</sup>.
- Об'єм приміщення: 84 м<sup>3</sup>.
- Об'єм на одне робоче місце: 28 м<sup>3</sup>.

Схема-модель приміщення приведена на рис. 6.1.



Рисунок 6.1 План приміщення

Вимоги, яким повинні задовольняти робочі приміщення, приведені в НПАОП 0.00-1.28-10 “Правила охорони праці під час експлуатації ЕОМ” – Держгірпромнагляд, № 65 від 26 березня 2010 р. [50]:

- обсяг приміщення не менше  $20 \text{ м}^3$  на людину;
- площа приміщення не менш  $6 \text{ м}^2$  на людину.

Отже, приміщення відповідає вимогам.

Дані три робочих місця обладнані спеціальними комп'ютерними столами без столешниць під клавіатуру, та звичайними офісними стільцями.

Порівняння фактичних і нормативних характеристики робочого місця приведено в таблиці 6.1.

Використовується клавіатура *4TECH*, що повністю відповідає вимогам ергономіки клавіатур, а саме:

- зміна нахилу поверхні клавіатури повинне лежати в межах від  $5^{\circ}$  до  $15^{\circ}$ ;
- висота середнього ряду клавіш - не більш 30 мм;
- вільний простір від нижнього ряду кнопок до передньої крайки клавіатури повинний мати ширину 80-100 мм тоді, коли крайка піднімається більше чим на 20 мм (для великих рук);
- вільний простір між крайкою клавіатури і краєм столу повинен мати ширину 80-100 мм у тому випадку, якщо висота передньої крайки клавіатури менше 20 мм (для маленьких рук);
- розмір контактної площини клавіш, розрахований на антропометричні характеристики вітчизняного користувача, по горизонталі повинний бути не менш 13 мм, по вертикалі - 15 мм;
- відстань між контактними площинами клавіш не може бути менше 3 мм, що визначається точністю позиціонування пальців;
- рівний для всіх клавіш робочий хід - 1,0-5,0 мм;
- до всіх клавіш повинне прикладатися однакове зусилля натискання 0,25-1,5 Н;
- клавіатура повинна мати можливість переміщатися щодо монітора в межах 0,5-1,0 м.

Щодо миші, то використовується миша марки *4TECH*, параметри якої також відповідають нормативам:

- зручний хват забезпечується завдяки глибокій виїмці для великого пальця;

- кнопки для великого пальця легко доступні, те ж стосується і центральної кнопки.

Враховуючи вище описане, ергономіка робочого місця майже повністю відповідає нормам.

Характеристики робочого місця відповідають нормативним вимогам у всьому (та ДСанПіН 3.3.2-007-98) [51].

Таблиця 6.2 - Порівняння фактичних і нормативних характеристик робочого місця

Характеристика	Норматив	Фактичне значення
Висота робочої поверхні	680-800 мм	750 мм
Глибина робочої поверхні	800-1000мм	950мм
Висота сидіння над рівнем підлоги	400-500 мм	460 мм
Висота спинки стільця	300+/- 20 мм	310 мм
Регулювання нахилу спинки крісла	1-30°	0°
Глибина сидіння	400 мм та більше	470 мм
Висота простору для ніг	600 мм	700 мм
Ширина простору для ніг	500 мм та більше	520 мм
Глибина простору для ніг	650 мм та більше	830 мм

Відстань від екрану до очей	600-700	620
-----------------------------	---------	-----

## 6.4 Освітлення

У приміщенні використовується природне й штучне освітлення. Природне освітлення здійснюється за допомогою віконних прорізів. Приміщення знаходиться на підвищенні, у відносній віддаленості від інших споруд, через це будь-які перешкоди природному освітленню в даному випадку відсутні. В середині приміщення стеля і стіни пофарбовані в білий колір.

Штучне освітлення в кімнаті здійснюється системою загального рівномірного освітлення, що реалізована на основі люмінесцентних ламп типу ЛБ-40, що мають наступні достоїнства :

- висока світлова віддача;
- тривалий термін служби;
- мала яскравість поверхні, що світиться;
- близькість спеціального складу до природного висвітлення.

Робота за дисплеєм ПЕОМ за розрядом зорових робіт відноситься до III розряду. При загальному висвітленні освітленість робочого місця повинна становити від 200 до 500 лк [52].

При штучному освітленні нормуються наступні параметри:

- $E$  (лк) - найменша припустима освітленість;
- $M$  - показник дискомфорту;
- $K_p$  (%) - коефіцієнт пульсації освітленості;

Перевіримо, чи відповідають нормам фактичні параметри штучного освітлення в приміщенні. Номінальний світловий потік лампи білого світіння ЛБ 40.

$$F_{л} = 3000 \text{ лм.}$$

У приміщенні застосовується світильник, у якому встановлено дві лампи.

Висоту підвісу світильника визначимо з формули :

$$h = H - h_c - h_p - h_n, \text{ де}$$

$H$  - висота приміщення, м;

$h_c$  - висота світильника, м;

$h_n$  - відстань від стелі до підвісу, м;

$h_p$  - висота робочої поверхні, м.

Для розглянутого приміщення :

$$H = 3 \text{ м; } h_c = 0.15 \text{ м; } h_n = 0.2 \text{ м; } h_p = 0.7 \text{ м.}$$

звідси :

$$h = 3 - 0.15 - 0.7 - 0.2 = 1.95 \text{ м.}$$

Світильник розташований в центрі кімнати. Висота підвісу світильника становить 1.95 метра. Приміщення має наступні габарити:

довжина  $A = 7$  метрів,

ширина  $B = 4$  метрів.

Визначимо освітленість у робочій точці. Для розрахунку загальної рівномірної освітленості при горизонтальній робочій поверхні використаємо метод коефіцієнта використання світлового потоку.

Розрахункова формула для світлового потоку світильника має вигляд:

$$F_{л} = \frac{E * K * S * Z}{N * n * \eta_B}, \quad (6.3)$$

Де  $N$  - число світильників у приміщенні,  $N = 3 * 2 = 6$ ;

$n$  - кількість ламп у світильнику;

$F_l$  - світловий потік ламп, лм (для ламп люмінесцентних від 780 до 4960 лм., на досліджуваному об'єкті використовуються лампи потужністю 40 Вт., типу ЛБ-40, світловий потік яких дорівнює 3000 лм.);

$K$  - коефіцієнт запасу,  $K = 1.5$ ;

$Z$  - коефіцієнт нерівномірності ( $Z=1,1$ );

$S$  - площа приміщення;

$E$  - освітленість, створювана всіма світильниками.

$\eta_B$  - коефіцієнт використання світлового потоку ( $\eta_B = 0,4 \div 0,6$ );

$$\eta_B = \frac{0.4 + 0.6}{2} = 0.5; \quad (6.4)$$

Звідси одержуємо формулу для розрахунку освітленості на робочому місці :

$$E_\phi = \frac{F_l * \eta_B * N * n}{S * K * Z}; \quad (6.5)$$

Коефіцієнт використання світлового потоку залежить від:

- ККД, кривій розподілу сили світла світильника;
- коефіцієнта відбиття стелі  $R_c$  і стін  $R_c$ ;
- висоти підвісу світильників  $h_n$ .

Показник приміщення і обчислимо за формулою:

$$i = \frac{A * B}{h * (A + B)}; \quad (6.6)$$

$$i = (6 * 5) / (1,95 * (6 + 5)) = 1,4.$$

$$E_\phi = \frac{3000 * 0.5 * 3 * 2}{28 * 1,5 * 1.1} = 194,8 \text{ лк.}$$

Виходячи з того, що по розряду зорової роботи робота за дисплеєм ПЕОМ відноситься до III розряду, тому при загальному освітленні освітленість робочого місця повинна становити від 200 до 400 лк [53]. Фактична освітленість на



робочому місці становить 194,8 лк. У такий спосіб для роботи з дисплеєм майже достатньо існуючих джерел світла.

Вивчення показало, що для роботи з дисплеєм майже достатньо існуючих джерел світла, виходячи з того, що по розряду зорової роботи робота за дисплеєм ПЕОМ відноситься до III розряду.

## **6.5 Висновки**

Був проведений аналіз державних норм і правил облаштування робочого середовища користувача ЕОМ.

У ході розрахунків було визначено, що дотримуючись Державних санітарних правил і норм роботи з візуальними дисплейними терміналами ЕОМ природного освітлення та штучного освітлення цілком достатньо.

При дотриманні правил електробезпечності виключені випадки ураження працівника струмом.

Якщо дотримуватись правил пожежної безпеки та використовувати рекомендації щодо освітлення приміщення, можна досягнути максимальної працездатності персоналу.

## 7 ВИСНОВКИ

В рамках виконання дипломної роботи був проведений огляд технологій доповненої реальності, розглянуто та проаналізовано можливі типи взаємодії користувача з віртуальними об'єктами в таких системах. За результатами аналізу визначено їх основні недоліки запропоновані види взаємодії дозволяють максимально розкрити потенціал доповненої реальності.

Розроблено прототип системи доповненої реальності з можливістю визначити дані про відстежуваний об'єкт та вивід їх на дисплей. Розглянуті області де це можна застосувати. Отриманий додаток дозволяє користувачеві отримувати інформацію про об'єкт.

Традиційні обчислювальні машини комп'ютери, останнім часом відходять на задній план, їм на заміну приходять мобільні пристрої та окуляри доповненої реальності. Тому більшість розробок для комп'ютерів вигідно портувати на мобільну техніку. Це стосується бібліотек комп'ютерного зору в тому числі. Це дозволяє використовувати функції комп'ютерного зору на мобільних пристроях, але широкого застосування ці бібліотеки не зазнали на мобільних платформах так як поки не вистачає обчислювальних ресурсів, розробники вирішили розробити OpenVX для того щоб оптимізувати роботу комп'ютерного зору та доповненої реальності на мобільних пристроях.

Комп'ютерний зір та доповнена реальність має великий спектр пристосування у багатьох галузях. Використання на мобільних пристроях поки що не вирішує промислових проблем, для вирішення складних задач використовують комп'ютери. Широко використовується комп'ютерний зір та доповнена реальність у рекламі, відеоспостереженні, у системах виявлення і супроводу рухомих об'єктів за ознакою їх руху, біометрії, системах виявлення і розпізнавання облич та інших.

Існує багато алгоритмів комп'ютерного зору, деякі з них були розглянуті у цій роботі усі вони мають своє застосування на мобільних платформах.

Для мобільних операційних систем існує багато додатків з використанням комп'ютерного зору та доповненої реальності, але в багатьох випадках ці додатки не мають практичного застосування в широкому доступі. У цій області є місце для розробки інноваційного дійсно корисного додатка з використанням комп'ютерного зору, один з таких проектів розробляється в Україні, його назва Araped, це приклад застосування доповненої реальності в медицині та освіті, додаток можна буде використовувати для вивчення анатомії.

Поки що мобільні пристрої поступаються в швидкості більшості комп'ютерів, але найближчим часом ця ситуація зміниться, тоді можливо буде вистачати ресурсів для обробки великих об'ємів інформації на мобільному пристрої.

## 8 СПИСОК ПОСИЛАНЬ

1. Ronald T. Azuma A Survey of Augmented Reality // In Presence: Teleoperators and Virtual Environments. – 1997. – No 4. – P. 355–385.
2. Miika Tikander Development and evaluation of augmented reality audio systems: Abstract of dissertation for the degree of Doctor of Science in Technology. – Helsinki, 2009. – 70 p.
3. Hear&There: An Augmented Reality System of Linked Audio / Joseph Rozier, Karrie Karahalios, Judith Donath // Online Proceedings of the ICAD – Режим доступа:  
<http://www.icad.org/websiteV2.0/Conferences/ICAD2000/ICAD2000.html>, Дата доступа: 05.04.2015.
4. Global Positioning System. – Режим доступа: <http://www.gps.gov/>, Дата доступа: 18.05.2015
5. Глобальная спутниковая навигационная система ГЛОНАСС. Интерфейсный контрольный документ. – Редакция 5.1. – М.: КНИЦ МО России, 2008. – 74 с.
6. GaitAid Virtual Walker for Movement disorder patients - Режим доступа:  
<http://www.medigait.com/index.html>, Дата доступа: 17.04.2015.
7. PHANTOM Premium 6DOF – Режим доступа: <http://www.sensable.com/haptic-phantom-premium-6dof.htm>, Дата доступа: 17.05.2015.
8. Exploring Visuo-Haptic Mixed Reality/ Christian Sandor, Tsuyoshi Kuroki, Shinji Uchiyama, Hiroyuki Yamamoto // IEIC Technical Report (Institute of Electronics, Information and Communication Engineers). –2007. – Vol. 106, No470. – P. 31–36.
9. Visuohaptic Simulation of Bone Surgery for Training and Evaluation / Dan Morris, Christopher Sewell, Federico Barbagli // IEEE Computer Graphics and

- Applications. – 2006. – Vol. 26, No 6. – P. 48–57.
10. Augmented Reality Browser: Layar – Режим доступа: <http://www2.layar.com/>,  
Дата доступа: 04.05.2015.
  11. Daniel Wagner and Dieter Schmalstieg, “Handheld Augmented Reality Displays”, Graz University of Technology, Austria.
  12. Oliver Bimber, Ramesh Raskar, Masahiko Inami, “Spatial Augmented Reality”, *SIGGRAPH 2007 Course 17 Notes*, 2007.
  13. Assaf Feldman, Emmanuel Munguia Tapia, Sajid Sadi, Pattie Maes, Chris Schmandt, “ReachMedia: On-the-move interaction with everyday objects”, iswc, pp.52–59, *Ninth IEEE International Symposium on Wearable Computers (ISWC’05)*, 2005.
  14. Li Yi-bo; Kang Shao-peng; Qiao Zhi-hua; Zhu Qiong; “Development Actuality and Appli- cation of Registration Technology in Augmented Reality”, *Computational Intelligence and Design, 2008. ISCID ‘08. International Symposium on*, Vol.2, No., pp.69–74, 17–18 Oct. 2008
  15. H. Kato, M. Billinghurst, I. Poupyrev, K. Imamoto, K. Tachibana, “Virtual Object Manipulation on a Table-Top AR Environment”, *ISAR’00*, 111–119, 2000.
  16. Pranav Mistry, Tsuyoshi Kuroki, and Chaochi Chuang, “TaPuMa: Tangible Public Map for Information Acquirement through the Things We Carry”, MIT Media Lab, *Ambi-sys’08*, February 2008.
  17. Dieter Schmalstieg, Anton Fuhrmann, Gerd Hesina, “Bridging Multiple User Interface Dimensions with Augmented Reality”, *IEEE*, 2000.
  18. Istvan Barakonyi, Tamer Fahmy, Dieter Schmalstieg, “Remote collaboration using Augmented Reality Videoconferencing”, *Proceedings of Graphics Interface*

- 2004, p.89–96, May 17–19, 2004, London, Ontario, Canada
19. C. Sandor, A. Olwal, B. Bell and S. Feiner, “Immersive mixed-reality configuration of hybrid user interfaces”, In *ISMAR '05*, pp. 110–113, 2005
  20. Pranav Mistry, Pattie Maes, Liyan Chang, “WUW – Wear Ur World – A Wearable Gestural Interface”, *ACM, CHI 2009*, Boston, April 4–9, 2009.
  21. Jae-Young Lee; Seok-Han Lee; Hyung-Min Park; Sang-Keun Lee; Jong-Soo Choi; Jun-Sik Kwon; “Design and implementation of a wearable AR annotation system using gaze interaction,” *Consumer Electronics (ICCE), 2010 Digest of Technical Papers International Conference on*, vol., no., pp.185–186, 9–13 Jan. 2010
  22. Режим доступа: <http://www.google.com/mobile/goggles/#text>, Google Goggles, 2010 - Дата доступа: 08.05.2015
  23. Режим доступа: [www.mini.com](http://www.mini.com) MINI - Дата доступа: 02.05.2015
  24. Режим доступа: <http://www.geekologie.com/2008/12/14-week>, Cool: Augmented Reality Advertisements, Dec. 19 2008 - Дата доступа: 02.05.2015
  25. Режим доступа: <http://www.augmented-reality-games.com> Beyond Reality. - Дата доступа: 02.05.2015
  26. Marco Sacco, Stefano Mottura, Luca Greci, Giampaolo Vigan, *Institute of Industrial Technologies and Automation*, National Research Council, Italy.
  27. Yetao Huang, Yue Liu, Yongtian Wang, “AR-View: and Augmented Reality Device for Digital Reconstruction of Yuangmingyuan”, *IEEE International Symposium on Mixed and Augmented Reality*, 2009.
  28. Rainer Malaka, Kerstin Schneider, Ursula Kretschmer, “Stage-Based Augmented Edutainment”, *Smart Graphics 2004*, pp.54–65

29. Bruns, E.; Brombach, B.; Zeidler, T.; Bimber, O., “Enabling Mobile Phones To Support Large-Scale Museum Guidance”, *Multimedia, IEEE*, vol.14, no.2, pp.16–25, April-June 2007
30. Mark Billinghurst, “The MagicBook: A Transitional AR Interface”, Ivan Poupyrev, 2001.
31. Malaka, R., Schneider, K., and Kretschmer, U. Stage-Based Augmented Edutainment. In LCNS 3031 (2004), 54–65
32. Visuohaptic Simulation of Bone Surgery for Training and Evaluation / Dan Morris, Christopher Sewell, Federico Barbagli // *IEEE Computer Graphics and Applications*. – 2006. – Vol. 26, No 6. – P. 48–57.],
33. Режим доступа:  
[http://www.bmw.com/com/en/owners/service/augmented\\_reality\\_workshop\\_1.html](http://www.bmw.com/com/en/owners/service/augmented_reality_workshop_1.html)  
 ml - Дата доступа: 09.05.2015
34. Augmented Reality Browser: Layar. – Режим доступа: <http://www2.layar.com/>,  
 - Дата доступа: 04.05.2015
35. Режим доступа: <http://www.wikitude.org/dewikitude-drive-eyes-road-again>wikitude-drive-eyes- road, WikitudeDrive: Never take your eyes off the road again, 2009. - Дата доступа: 04.05.2015
36. Режим доступа: [http://www.readwriteweb.com/archives/how\\_iphone\\_4\\_could\\_change\\_augmented\\_reality.php](http://www.readwriteweb.com/archives/how_iphone_4_could_change_augmented_reality.php), How iPhone 4 Could Change Augmented Reality, Chris Cameron, June 10, 2010 - Дата доступа: 16.05.2015
37. Kumar A, Waldron K (1981) The workspace of a mechanical manipulator. *Trans ASME J Mech Des* 103:665–672
38. Gesture-based interaction via finger tracking for mobile augmented reality Wolfgang Hürst & Casper van Wezel *Multimed Tools Appl* (2013) 62:233–258

39. Шапиро Л. Компьютерное зрение / Л. Шапиро, Дж. Стокман; Пер. с англ. – М.: БИНОМ. Лаборатория знаний, 2006.
40. Azuma R. A Survey of Augmented Reality // Presence: Teleoperators and Virtual Environments, 1997. P. 355–385.
41. Amit Y. 2D Object Detection and Recognition: Models, Algorithms and Networks. The MIT Press, 2002.
42. Bay H., Tuytelaars T., L. Van. Gool. Surf: Speed up robust features // European Conference on Computer Vision, 2006. P. 404–417.
43. Hartley R., Zisserman A. Multiple view geometry in computer vision. 2nd edition. Cambridge University Press, 2004.
44. Lepetit V., Fua P., Pilet J. Point Matching as a Classification Problem for Fast and Robust Object Pose Estimation // Conference on Computer Vision and Pattern Recognition, 2004.
45. Mikolajczyk K., Schmid C. An Affine Invariant Interest Point Point Detector // European Conference on Computer Vision, 2002. P. 128–142.
46. Milgram P., Kishino A. F. Taxonomy of Mixed Reality Visual Displays // IEICE Transactions on Information and Systems. 1994. E77-D(12). P. 1321–1329.
47. Moreels P., Perona P. Evaluation of Features Detectors and Descriptors based on 3D objects // IJCV, 2006.
48. Obeysekera M. Affine Reconstruction from multiple views using Singular Value Decomposition / School of Computer Science and Software Engineering, The University of Western Australia, 2003. 138 Моделирование и анализ информационных систем Т. 20, № 2 (2013) 11. Wuest H., Vial F., Stricker D. Adaptive line tracking with multiple hypotheses for augmented reality // Proceedings of the Fourth IEEE and ACM International Symposium on Mixed and Augmented Reality, 2005. P. 62–69.
49. Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин ДСАНПІН 3.3.2.007-98



50. Правила охорони праці під час експлуатації ЕОМ. Наказ від 26 березня 2010 року N 65
51. Санітарні норми мікроклімата Постанова. № 42 ДСН 3.3.6.042-99
52. Шум, ультразвук, інфразвук. ДСН 3.3.6.037-99
53. Норми визначення катерогий. НАПБ Б.03.002-2007.