

ДОДАТОК А

Перелік джерел посилання за науковими напрямками керівника та науковців
кафедри програмної інженерії

2. Sus B., Revenchuk I., Tmienova N., Bauzha, O., Chaikivskyi T. Software System for Virtual Laboratory Works//2020 IEEE 15th International Scientific and 59 Technical Conference on Computer Sciences and Information Technologies. 2020. P. 396–399.

3. Sus B., Tmienova N., Revenchuk I., Vialkova V. Development of Virtual Laboratory Works for Technical and Computer Sciences//ICIST – International Conference on Information and Software Technologies. 2019. P. 383-394

5. Sus B., Tmienova N., Revenchuk I., Bauzha O., Stirenko S. Principles of gamification for Virtual laboratory works and educational courses//Матеріали IX Міжнародної науково-практичної конференції «Інформаційні управляючі системи і технології» (ІУСТ-ОДЕСА-2020). 2020. С. 63-65.

11. Sus B., Tmienova N., Revenchuk I., Bauzha O., Chaikivskyi T. Software System for Virtual Laboratory Works//XV International Scientific and Technical Conference Computer Science and Information Technologies. 2020. P. 396-400.

19. Ревенчук І., Агарков Є. Моделювання доповненої реальності на основі маркерів//Бионика интеллекта. 2021. № 96. С. 90–95.

23. Назаров О. С., Шураєв І. Д. Автоматизація процесу створення 3D моделей на основі зображень//Міжнародна наукова інтернет-конференція "Інформаційне суспільство: технологічні, економічні та технічні аспекти становлення"/Збірник тез доповідей: випуск 34. 2018. С. 54-56.

ДОДАТОК Б

Звіт результатів перевірки кваліфікаційної роботи на унікальність тексту



Ім'я користувача:
Олійник Олена Володимирівна каф. ПІ

Дата перевірки:
30.05.2024 19:42:40 EEST

Дата звіту:
30.05.2024 20:52:27 EEST

ID перевірки:
1016300292

Тип перевірки:
Doc vs Internet + Library

ID користувача:
100012353

Назва документа: 2024_М_ПІ_ІПЗм-22-4_Меньшикова_А_А_скорочений

Кількість сторінок: 77 Кількість слів: 16553 Кількість символів: 132358 Розмір файлу: 1.86 MB ID файлу: 1016095722

0.31%
Схожість

Найбільша схожість: 0.06% з Інтернет-джерелом (<http://www.cek.ef.uni-lj.si/magister/senk2452-B.pdf>)

0.26% Джерела з Інтернету

34

Сторінка 79

0.1% Джерела з Бібліотеки

5

Сторінка 79

0% Цитат

Вилучення цитат вимкнене

Вилучення списку бібліографічних посилань вимкнене

0%
Вилучень

Немає вилучених джерел

ДОДАТОК В

Відображення процесу програмної реалізації віртуальної примірочної

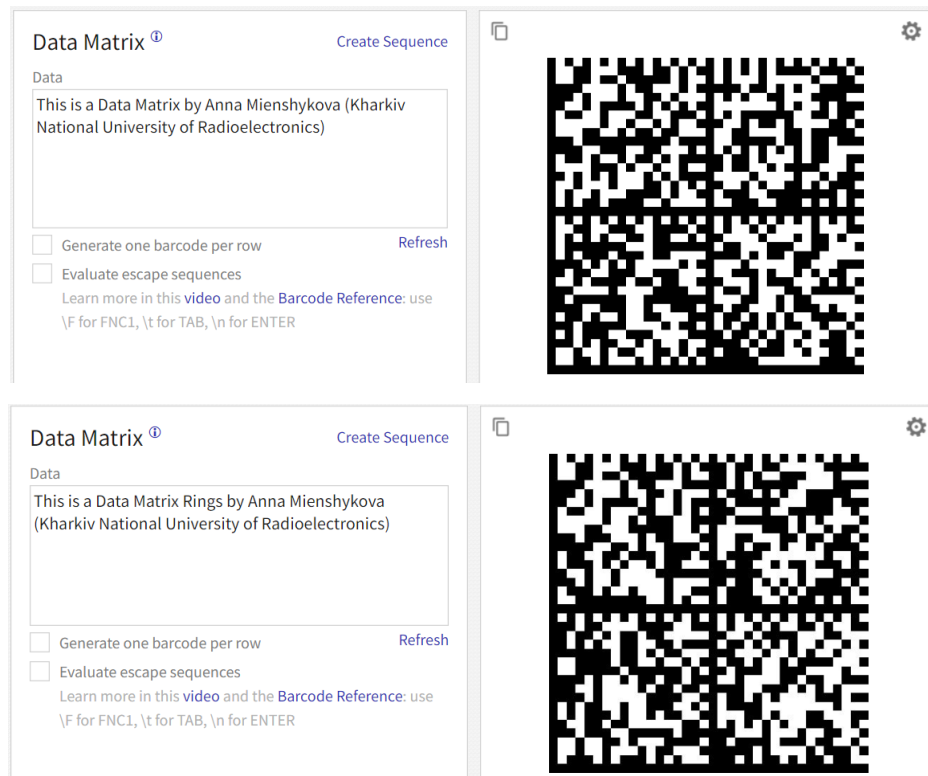


Рисунок В.1 – Згенеровані бінарні маркери: закодований текст (ліворуч) та двовимірний матричний штрих-код (праворуч)



Type: Image

Status: Active

Target ID: 7e4194f96ce14b189f15f3b6e4dde71a

Augmentable: ★ ★ ★ ★ ★

Рисунок В.2 – Ефективність розпізнавання двовимірного матричного штрих-коду



Рисунок В.3 – Готові 3D-моделі годинників та каблучок

Database [Edit Name](#)

Type: Device

Targets (2)

☐	Image	Target Name	Type	Rating ①	Status ▾
☐		barcode4	Image	★★★★★	Active
☐		barcode3	Image	★★★★★	Active
☐		barcode2	Image	★★★★★	Active
☐		barcode	Image	★★★★★	Active

Рисунок В.4 – База даних зображень штрих-коду

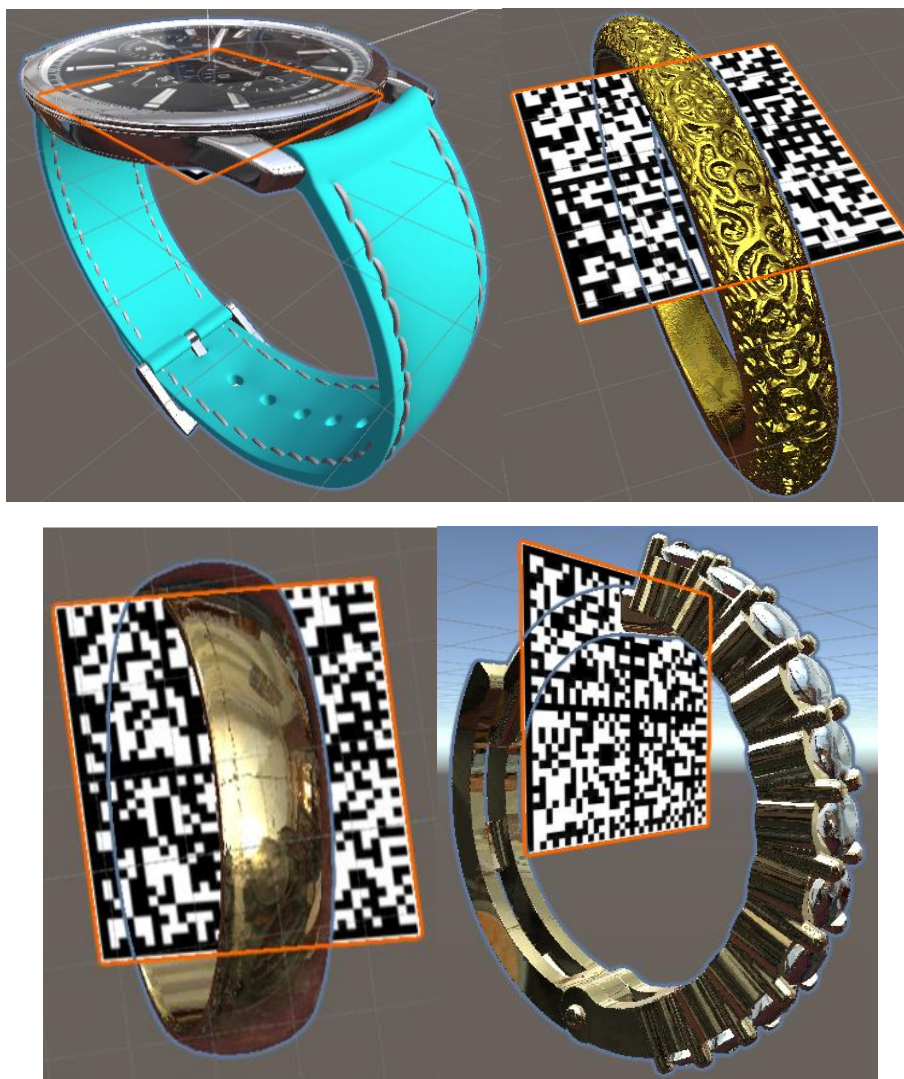


Рисунок В.5 – Накладання моделей аксесуарів на об'єкт зображення



Рисунок В.6 – Тестування накладання моделі годинника та обручки на об'єкт зображення

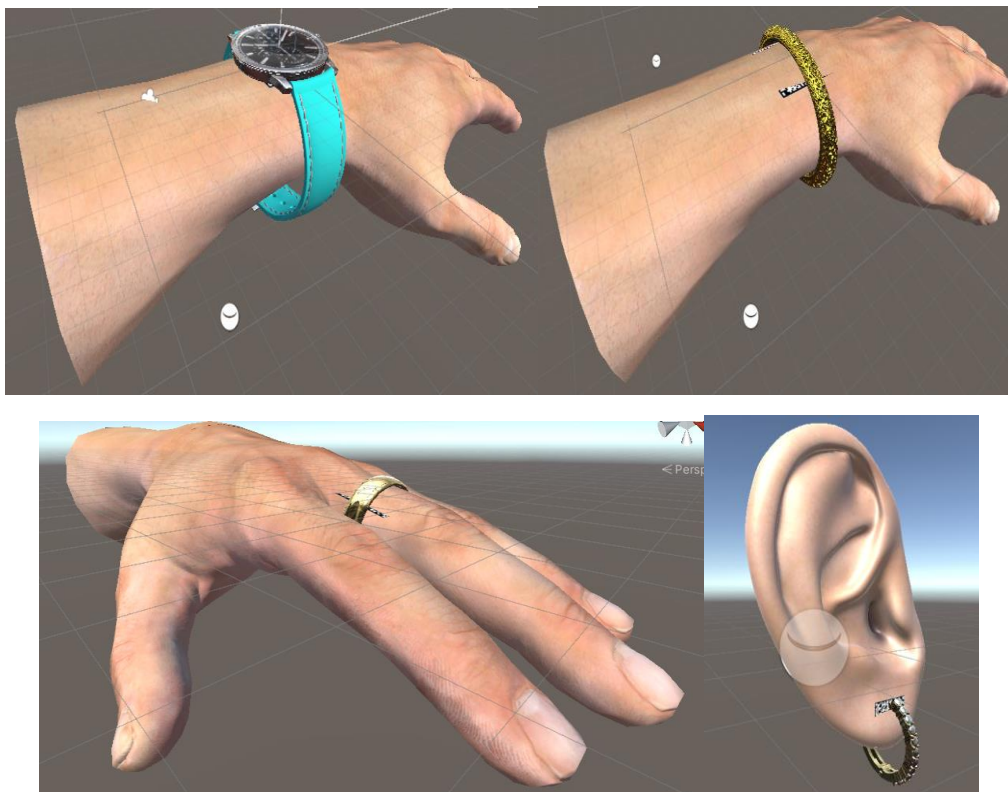


Рисунок В.7 – Обгортання моделей аксесуарів навколо моделі руки та надягання моделі сережок на вухо

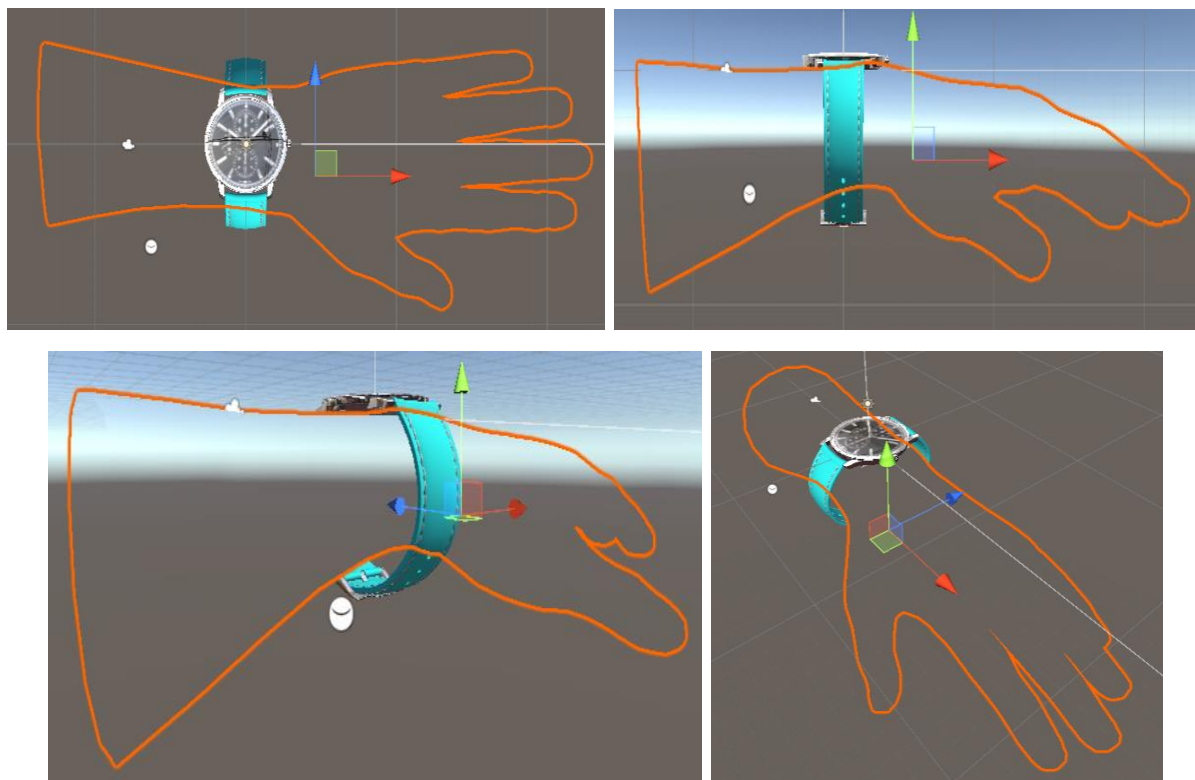


Рисунок В.8 – Застосування матеріалу оклюзії до моделі руки (на прикладі годинника з різних сторін)



Рисунок В.9 – Тестування оклюзії на мобільному пристрої

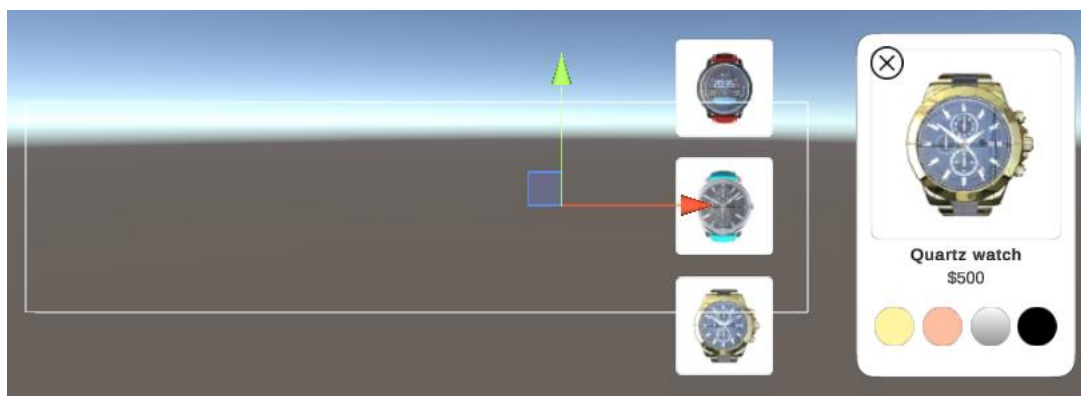


Рисунок В.10 – Користувацький інтерфейс після вибору категорії годинники

```

public class AccessoriesSelect : MonoBehaviour
{
    public GameObject watchModel1;
    public GameObject watchModel2;
    public GameObject watchModel3;
    public GameObject ringModel1;
    public GameObject ringModel2;
    public GameObject ringModel3;

    public GameObject w1Window;
    public GameObject w2Window;
    public GameObject w3Window;
    public GameObject w4Window;
    public GameObject w5Window;
    public GameObject w6Window;

    public void AccessoryOneButtonClicked()
    {
        watchModel1.SetActive(true);
        watchModel2.SetActive(false);
        watchModel3.SetActive(false);
    }

    public void AccessoryTwoButtonClicked()
    {
        watchModel1.SetActive(false);
        watchModel2.SetActive(true);
        watchModel3.SetActive(false);
    }

    public void AccessoryThreeButtonClicked()
    {
        watchModel1.SetActive(false);
        watchModel2.SetActive(false);
        watchModel3.SetActive(true);
    }
}

```

Рисунок В.11 – Код логіки взаємодії (аналогічний для трьох інших методів каблучки, браслета та сережок)

```

public class AccessoriesSelect : MonoBehaviour
{
    public GameObject watchModel1;
    public GameObject watchModel2;
    public GameObject watchModel3;
    public GameObject ringModel1;
    public GameObject ringModel2;
    public GameObject ringModel3;

    public GameObject w1Window;
    public GameObject w2Window;
    public GameObject w3Window;
    public GameObject w4Window;
    public GameObject w5Window;
    public GameObject w6Window;

    Animation w1WindowAnimation;
    Animation w2WindowAnimation;
    Animation w3WindowAnimation;
    Animation w4WindowAnimation;
    Animation w5WindowAnimation;
    Animation w6WindowAnimation;

    void Start()
    {
        w1WindowAnimation = w1Window.GetComponent<Animation>();
        w2WindowAnimation = w2Window.GetComponent<Animation>();
        w3WindowAnimation = w3Window.GetComponent<Animation>();
        w4WindowAnimation = w4Window.GetComponent<Animation>();
        w5WindowAnimation = w5Window.GetComponent<Animation>();
        w6WindowAnimation = w6Window.GetComponent<Animation>();
    }

    public void AccessoryOneButtonClicked()
    {
        watchModel1.SetActive(true);
        watchModel2.SetActive(false);
        watchModel3.SetActive(false);

        w1WindowAnimation["w1animation"].speed = 1;
        w1WindowAnimation.Play();
    }

    public void AccessoryTwoButtonClicked()
    {
        watchModel1.SetActive(false);
        watchModel2.SetActive(true);
        watchModel3.SetActive(false);

        w2WindowAnimation["w2animation"].speed = 1;
        w2WindowAnimation.Play();
    }

    public void AccessoryThreeButtonClicked()
    {
        watchModel1.SetActive(false);
        watchModel2.SetActive(false);
        watchModel3.SetActive(true);

        w3WindowAnimation["w3animation"].speed = 1;
        w3WindowAnimation.Play();
    }
}

```

Рисунок В.12 – Код керування анімацією вікна

```

public void CloseButtonClicked()
{
    string buttonName = EventSystem.current.currentSelectedGameObject.name;

    if (buttonName == "w1close")
    {
        w1WindowAnimation["w1animation"].speed = -1;
        w1WindowAnimation["w1animation"].time = w1WindowAnimation["w1animation"].length;
        w1WindowAnimation.Play();
    }

    else if (buttonName == "w2close")
    {
        w2WindowAnimation["w2animation"].speed = -1;
        w2WindowAnimation["w2animation"].time = w2WindowAnimation["w2animation"].length;
        w2WindowAnimation.Play();
    }

    else if (buttonName == "w3close")
    {
        w3WindowAnimation["w3animation"].speed = -1;
        w3WindowAnimation["w3animation"].time = w3WindowAnimation["w3animation"].length;
        w3WindowAnimation.Play();
    }
}

```

Рисунок В.13 – Код закриття для кожного вікна аксесуару

```

public class ColorSelect : MonoBehaviour
{
    public Material band1;
    public Material band2;
    public Material band3;
    public Material ring1;
    public Material ring2;
    public Material ring3;

    void Start()
    {
        // ...
    }
}

public void w1ColorSelect()
{
    string buttonName = EventSystem.current.currentSelectedGameObject.name;

    if (buttonName == "Red button")
    {
        Color myColor = new Color();
        ColorUtility.TryParseHtmlString("#800000", out myColor);
        band1.color = myColor;
    }

    else if (buttonName == "Green button")
    {
        Color myColor = new Color();
        ColorUtility.TryParseHtmlString("#00802E", out myColor);
        band1.color = myColor;
    }

    else if (buttonName == "Light Blue button")
    {
        Color myColor = new Color();
        ColorUtility.TryParseHtmlString("#90DAFF", out myColor);
        band1.color = myColor;
    }

    else if (buttonName == "Black button")
    {
        Color myColor = new Color();
        ColorUtility.TryParseHtmlString("#000000", out myColor);
        band1.color = myColor;
    }
}

```

Рисунок В.14 – Код зміни кольору

```

// Функція для обчислення нормованого значення крос-кореляційної розбіжності
public float CalculateNormalizedCrossCorrelation(float[] markerPixels, float[] templatePixels)
{
    // Обчислення середнього значення пікселів маркера та шаблону
    float markerMean = CalculateMean(markerPixels);
    float templateMean = CalculateMean(templatePixels);

    // Ініціалізація змінних для підсумовування
    float numerator = 0f;
    float denominatorMarker = 0f;
    float denominatorTemplate = 0f;

    // Обчислення чисельника та знаменника для значення розбіжності
    for (int i = 0; i < markerPixels.Length; i++)
    {
        numerator += (markerPixels[i] - markerMean) * (templatePixels[i] - templateMean);
        denominatorMarker += Mathf.Pow((markerPixels[i] - markerMean), 2);
        denominatorTemplate += Mathf.Pow((templatePixels[i] - templateMean), 2);
    }

    // Обчислення значення розбіжності
    float discrepancyValue = numerator / (Mathf.Sqrt(denominatorMarker) * Mathf.Sqrt(denominatorTemplate));

    return discrepancyValue;
}

// Функція для обчислення середнього значення масиву значень пікселів
private float CalculateMean(float[] pixels)
{
    float sum = 0f;
    for (int i = 0; i < pixels.Length; i++)
    {
        sum += pixels[i];
    }
    return sum / pixels.Length;
}

```

Рисунок В.15 – Код метрики нормалізованої крос-кореляції

ДОДАТОК Г

Скріншоти взаємодії користувача з віртуальною примірочною аксесуарів

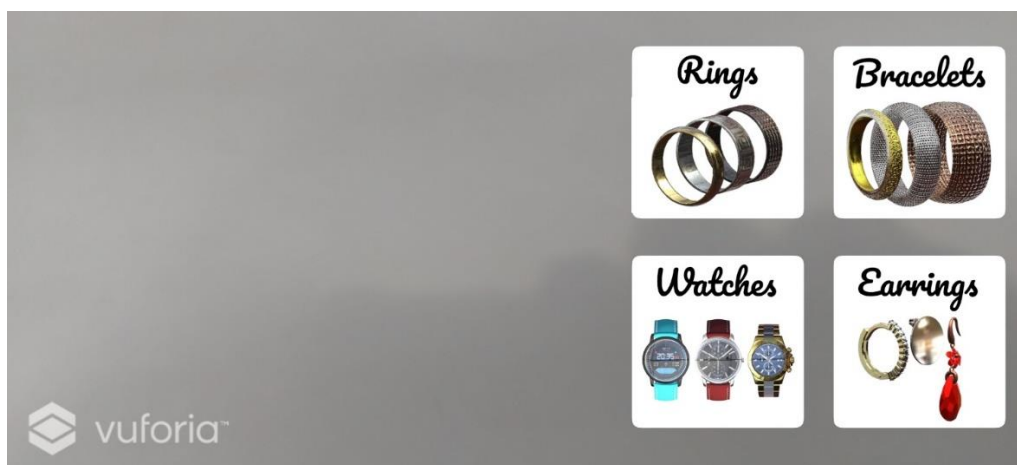


Рисунок Г.1 – Віртуальна примірочна перед вибором категорії аксесуара



Рисунок Г.2 – Віртуальна примірочна перед вибором моделі каблучки

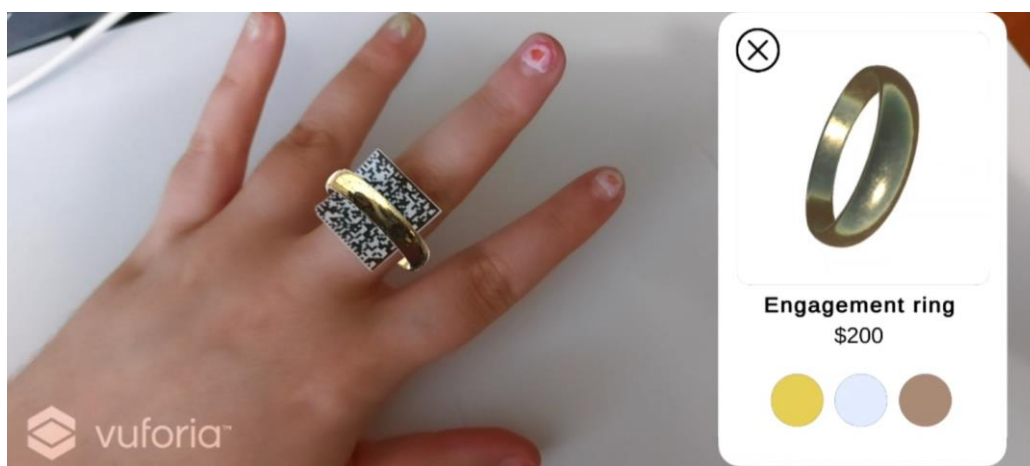


Рисунок Г.3 – Модель каблучки Engagement ring золотого кольору

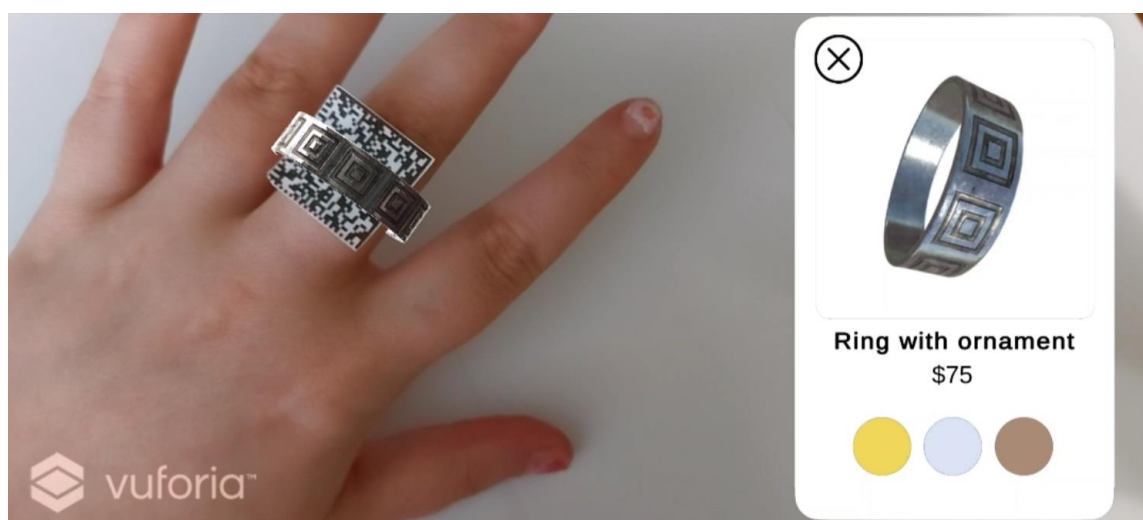


Рисунок Г.4 – Модель каблучки Ring with ornament срібного кольору

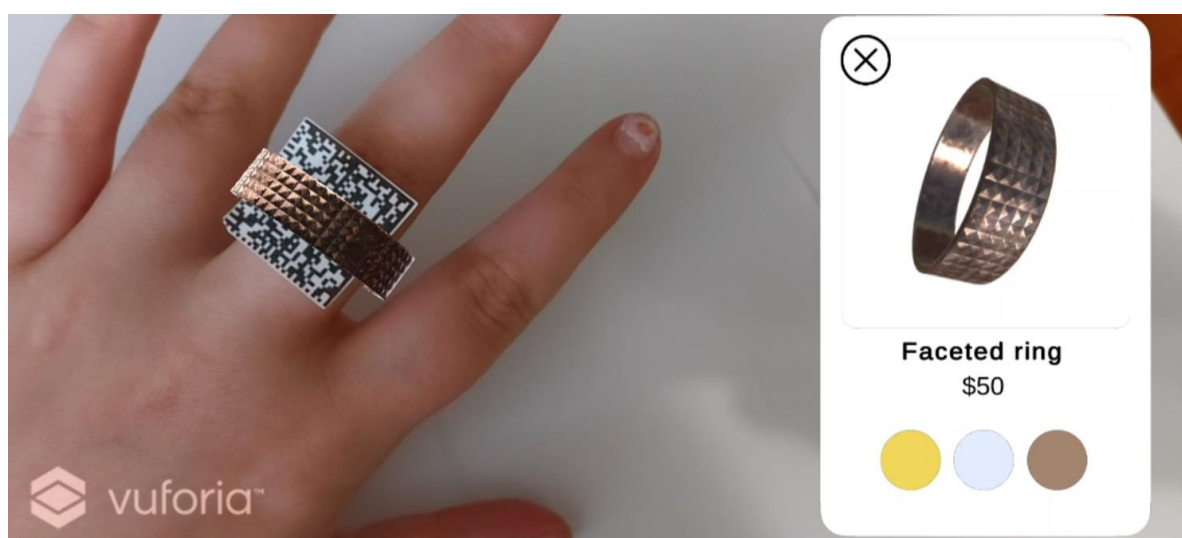


Рисунок Г.5 – Модель каблучки Faceted ring мідного кольору



Рисунок Г.6 – Віртуальна примірочна перед вибором моделі браслета



Рисунок Г.7 – Модель браслета Gold patterned bracelet золотого кольору



Рисунок Г.8 – Модель браслета Bismarck bracelet срібного кольору



Рисунок Г.9 – Модель браслета Wide bracelet червоно-золотого кольору



Рисунок Г.10 – Віртуальна примірючна перед вибором моделі годинника

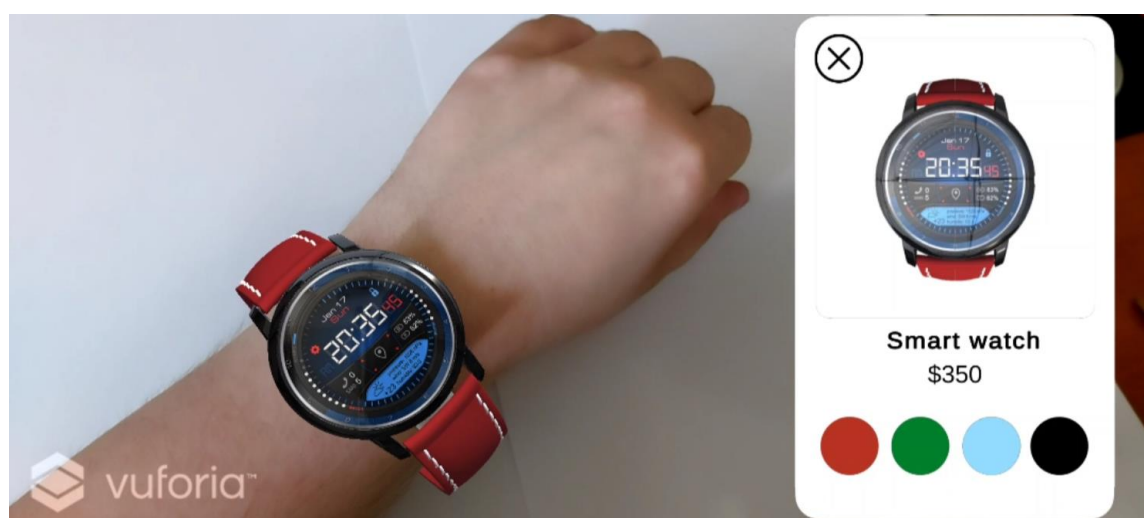


Рисунок Г.11 – Модель годинника Smart watch з червоним ремінцем

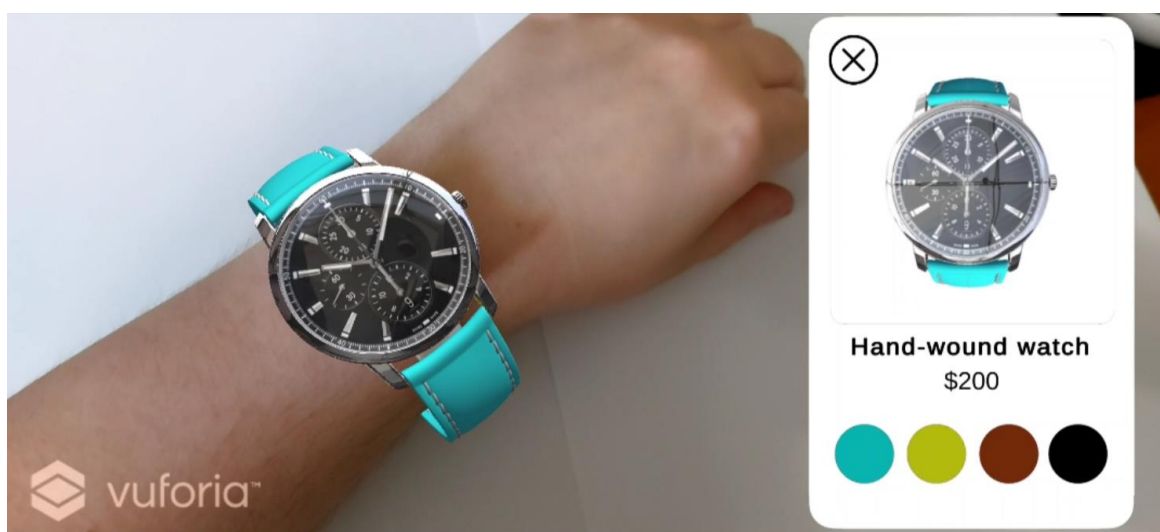


Рисунок Г.12 – Модель годинника Hand-wound watch з бірюзовим ремінцем

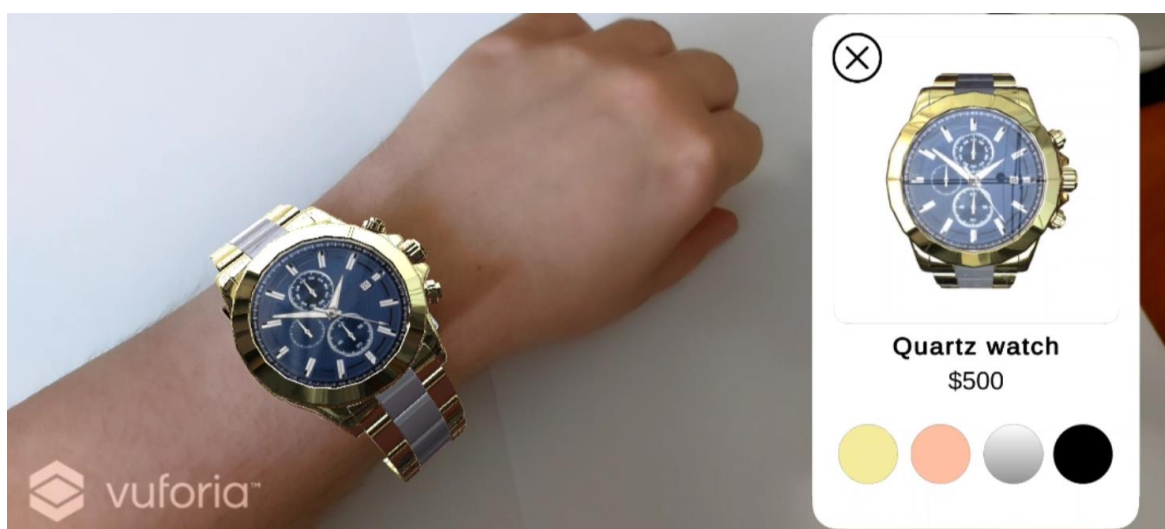


Рисунок Г.13 – Модель годинника Quartz watch з золотим ремінцем



Рисунок Г.14 – Віртуальна примірочна перед вибором моделі сережки

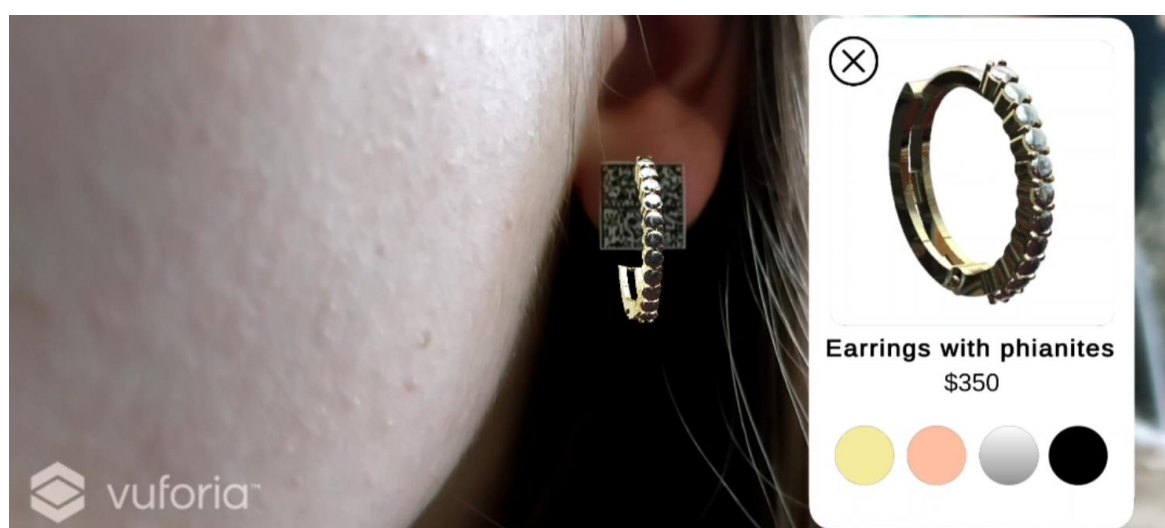


Рисунок Г.15 – Модель сережки Earrings with phianites золотого кольору

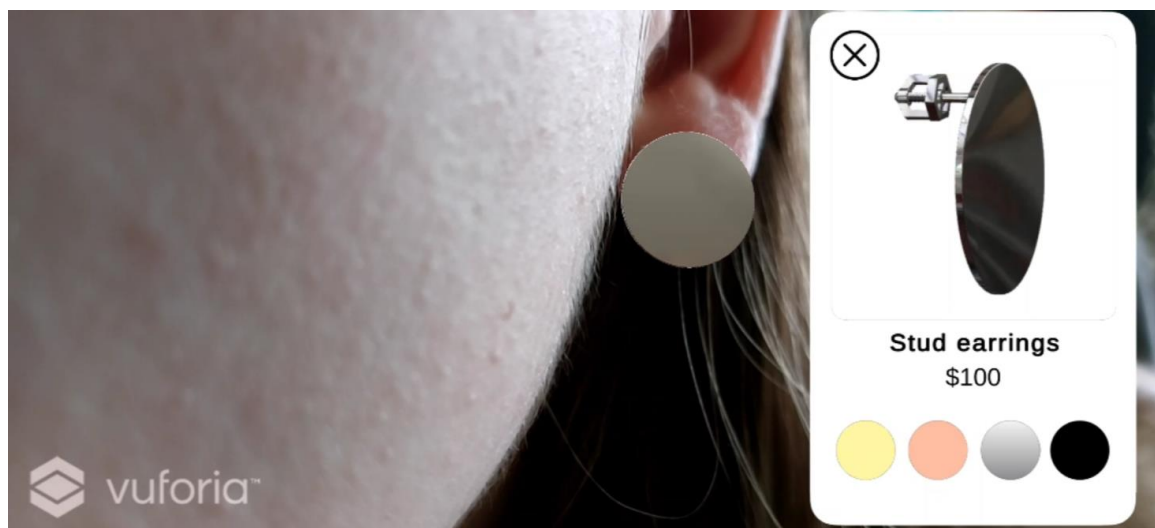


Рисунок Г.16 – Модель сережки Stud earrings срібного кольору

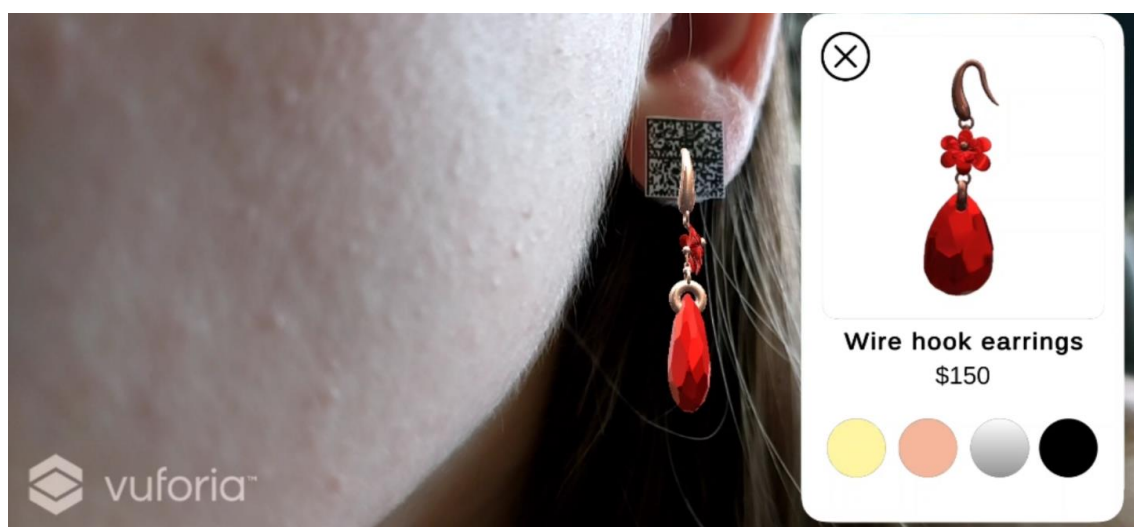


Рисунок Г.17 – Модель сережки Wire hook earrings червоно-золотого кольору

ДОДАТОК Д

Експертний висновок результатів перевірки кваліфікаційної роботи на
відповідність оформлення вимогам ДСТУ 3008:2015

Експертний висновок результатів перевірки кваліфікаційної роботи

студент
(посада)

програмної інженерії
(кафедра)

ІПЗМ-22-4
(група)

Меньшикова Анна Андріївна

(прізвище, ім'я, по батькові)

Зауваження

Пункт ДСТУ 3008-2015	Зміст пункту	Сторінка кваліфікаційної роботи
1	2	3
	7.1 Загальні положення	
	7.3 Нумерація сторінок звіту	
	7.4 Нумерація розділів, підрозділів, пунктів, підпунктів	
	7.5 Рисунки	
	7.6 Таблиці	
	7.7 Переліки	
	7.8 Примітки	
	7.9 Виноски	
	7.10 Формули та рівняння	
	7.11 Посилання	
	7.13 Список авторів	
	7.14 Скорочення та умовні позначки	
	7.15 Додатки	

Зауважень немає

Експерт

(підпис)

Олійник О.В.

(прізвище, ініціали)

01.06.2024

ДОДАТОК Е

Апробація результатів роботи

УДК 004.946:004.89

DOI: <https://doi.org/10.30837/IYF.IIS.2024.319>**ДОСЛІДЖЕННЯ ТЕХНОЛОГІЙ ДОПОВНЕНОЇ РЕАЛЬНОСТІ ДЛЯ
ПРОГРАМНОЇ РЕАЛІЗАЦІЇ ВІРТУАЛЬНОЇ ПРИМІРОЧНОЇ
АКСЕСУАРІВ ЗА ДОПОМОГОЮ ШТУЧНОГО ІНТЕЛЕКТУ**

Меньшикова А. А.

Науковий керівник – к.т.н., доцент Назаров О. С.

Харківський національний університет радіоелектроніки, каф. ПІ,
м. Харків, Українаe-mail: anna.mienshykova@nure.ua

This work is dedicated to the study of augmented reality technologies for the software implementation of a virtual accessory fitting room using artificial intelligence. Users can virtually try on different accessories in real time, which improves the online shopping experience using augmented reality technology. The study uses an artificial intelligence mechanism for marker recognition to accurately place and model accessories on the user's body. The findings contribute to the retail industry by offering insight into the potential of AI-based solutions to bridge the gap between physical and digital environments for accessory shopping.

Технології доповненої реальності у поєднанні зі штучним інтелектом змінюють різні сфери діяльності, поєднуючи цифровий та фізичний світи. Метою дослідження є вирішення проблеми обмеженого доступу до фізичної примірки аксесуарів під час онлайн-покупки шляхом визначення найбільш ефективної технології доповненої реальності та програмної реалізації віртуальної приміркової аксесуарів, використовуючи такий механізм штучного інтелекту, як розпізнавання маркерів. Пропозиція реалістичного та цікавого досвіду у віртуальному середовищі сприятиме більшій довірі до покупок аксесуарів без необхідності фізичної взаємодії.

У дослідницькій роботі порівнювалися такі технології доповненої реальності: Wikitude, ARKit, ARCore, Vuforia, Kudan та Unity AR Foundation. Платформа Wikitude фокусується на прив'язці до місцезнаходження з такими функціями, як розпізнавання зображень та 3D-відстеження. Фреймворк від Apple ARKit надає інструменти для відстеження руху, розуміння навколишнього середовища та оцінки освітленості. Платформа ARCore від Google пропонує можливості, подібні до ARKit, але адаптовані для екосистеми Android. Vuforia відома своїми потужними можливостями розпізнавання зображень та відстеження, що дозволяє розробникам створювати AR-додатки для різних цілей. Набір інструментів для розробки програмного забезпечення Kudan з функціями відстеження без маркерів, розпізнаванням 3D-об'єктів і місцезнаходження, придатний для крос-платформної розробки. Фреймворк Unity AR Foundation пропонує такі функції, як виявлення площин і відстеження об'єктів. Використовуючи метод лінійної адитивної згортки з ваговими

коефіцієнтами вдалося визначити оптимальну технологію для програмної реалізації віртуальної примірочної. За результатами обчислень, Vuforia виявилася найкращою альтернативою, маючи найвищий результат у функції згортки.

Використовуючи складні алгоритми та методи машинного навчання, розпізнавання маркерів дозволяє системам в режимі реального часу інтерпретувати та ідентифікувати візуальні підказки, наприклад, QR-коди, візерунки або фізичні маркери. У дослідженні використовувалися бінарні маркери, які дозволили системі встановлювати та підтримувати постійну точку відліку в динамічному середовищі, полегшуючи точне відтворення віртуальних елементів у реальному середовищі. У механізмі розпізнавання маркерів використовувалася метрика нормалізованої крос-кореляції, де для оцінки подібності між виявленим маркером і шаблоном обчислюється значення розбіжності:

$$D = \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_i (x_i - \bar{x})^2} \cdot \sqrt{\sum_i (y_i - \bar{y})^2}}$$

де $\bar{x}$ – середнє значення пікселів маркера, $\bar{y}$ – середнє значення пікселів шаблону, а підсумовування відбувається за відповідними значеннями пікселів.

Також були проведені експерименти, які передбачають маніпулювання такими факторами, як якість камери, умови освітлення та місцезнаходження людини, щоб оцінити їхній вплив на точність розміщення аксесуарів на людині у механізмі розпізнавання маркерів. Завдяки отриманим результатам, відповідні параметри були оптимізовані для підвищення стійкості та надійності цього механізму, призводячи до більш точної та реалістичної віртуальної примірки для користувачів.

Віртуальна примірочна була розроблена за допомогою технології доповненої реальності Vuforia та штучного інтелекту у якості мобільного додатку, яка працює на Android та iOS, забезпечуючи доступ широкому колу користувачів.

Отже, використовуючи можливості доповненої реальності та штучного інтелекту, очевидно, що віртуальні примірочні можуть запропонувати захоплюючий та інтерактивний досвід покупки аксесуарів. Результати цього дослідження мають практичне застосування в різних галузях, включаючи роздрібну торгівлю одягом, дизайн аксесуарів, електронну комерцію та маркетинг.

Список використаних джерел:

1. Ruzive Von N., Tsang P.J.H. Fashion Tech Applied. 1st edition. New York City : Apress, 2023. 281 p.
2. Liu M. Machine Learning, Animated. 1st edition. United Kingdom : Chapman and Hall/CRC, 2023. 436 p.
3. Ревенчук І., Агарков Є. Моделювання доповненої реальності на основі маркерів // Бионика интеллекта. 2021. №96. С. 90–95.

ДОДАТОК Ж

Слайди презентації



МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

**«Дослідження технологій доповненої реальності
для програмної реалізації віртуальної примірконої
аксесуарів за допомогою штучного інтелекту»**

Виконала: ст. гр. ІПЗм-22-4 Меньшикова А. А.
Науковий керівник: доц. каф. ПІ Назаров О. С.



Рисунок Ж.1 – Слайд 1

2

Постановка задачі

Дослідження технологій доповненої реальності:

- огляд і порівняння сучасних AR-технологій.

Аналіз механізму розпізнавання маркерів:

- визначення ролі маркерів та їх використання для точного відстеження;
- використання бінарних маркерів для стабільного розпізнавання в динамічному середовищі;
- реалізація механізму з адаптивним пороговим значенням та метрикою крос-кореляції;
- оптимізація та вдосконалення механізму на основі результатів експериментів.

Розробка віртуальної примірконої аксесуарів:

- проектування та розробка додатку з використанням AR та штучного інтелекту;
- акцент на користувацькому досвіді та мультиплатформеності.

Оцінка ефективності механізму розпізнавання маркерів:

- проведення експериментів для оцінки точності розташування аксесуарів;
- маніпулювання різними факторами, щоб визначити їх вплив на точність.

Рисунок Ж.2 – Слайд 2

Мета, актуальність та об'єкт дослідження

3

Визначення напрямку дослідження

Мета дослідження – вибір оптимальної технології доповненої реальності та розробка ефективного механізму розпізнавання маркерів для покращення онлайн-покупок, шляхом аналізу, порівняння технологій доповненої реальності та визначення їхньої придатності для створення віртуальної примірочної аксесуарів.

Актуальність та стан розвитку галузі

Електронна комерція в модній індустрії зіткнулася з проблемою відсутності фізичних примірочних, що перешкоджає переходу споживачів на цифрові платформи. Технології доповненої реальності та штучного інтелекту пропонують рішення цієї проблеми, дозволяючи створювати віртуальні примірочні для реалістичного взаємодії з товарами.

Об'єкт дослідження

Процес інтеграції віртуальних аксесуарів із реальним середовищем за допомогою технології доповненої реальності.

Рисунок Ж.3 – Слайд 3

Проблеми та протиріччя у вирішенні завдань

4

Точність моделювання

- x Проблеми з реалістичним відображенням аксесуарів на користувачах у реальному часі.
- ✓ Необхідність підвищення точності віртуальної примірки за допомогою алгоритмів та метрик.

Адаптація до реальних користувачів

- x Віртуальні примірочні важко адаптувати для великої кількості користувачів.
- ✓ Важливість забезпечення доступності для користувачів з різними потребами, включаючи користувачів з інвалідністю.

Персоналізація та кастомізація

- x Брак можливостей для персоналізації та індивідуальних налаштувань.
- ✓ Необхідність врахування індивідуальних нюансів фігури та стильових уподобань.

Технічні виклики інтеграції

- x Проблеми сумісності з різними пристроями та програмними системами.
- ✓ Забезпечення безперебійної роботи на різних платформах.

Конфіденційність та безпека даних

- x Важливість захисту особистих даних користувачів.
- ✓ Необхідність розробки надійних заходів для захисту конфіденційності та запобігання несанкціонованому доступу.

Рисунок Ж.4 – Слайд 4

Обґрунтування методології дослідження

5

Технології доповненої реальності

Розглядаємо платформи: Wikitude, ARKit, ARCore, Vuforia, Kudan та Unity AR Foundation.

Критерії вибору: наявність маркерного відстеження, сумісність та підтримка пристроїв, точність відстеження, простота інтеграції, продуктивність та швидкість реагування, середовище розробки та підтримка та наявність у тарифному плані безкоштовної опції.

Поєднання зі штучним інтелектом

Механізм розпізнавання маркерів у поєднанні з AR може покращити реалістичність та точність віртуальних аксесуарів.

Алгоритми штучного інтелекту дозволяють точно відстежувати та позиціонувати віртуальні об'єкти у фізичному середовищі.

Рисунок Ж.5 – Слайд 5

Огляд маркерного підходу

6

Бінарні маркери

Використовуються як візуальні підказки для точного доповнення віртуальних елементів на реальну сцену.

Полегшують постійне відслідковування точки відліку в динамічному середовищі.

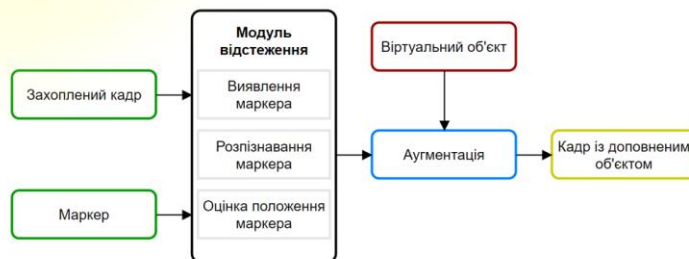


Рисунок 1 – Блок-схема системи доповненої реальності

Блок-схема системи доповненої реальності ілюструє послідовні кроки, що задіяні в процесі роботи системи. Вона включає такі етапи, як отримання зображення маркера, вилучення кадрів камери, модуль відстеження для обчислення відносного положення камери та процес додавання віртуальних компонентів до реальної сцени.

Формула нормалізованої крос-кореляції для оцінки подібності між виявленим маркером і шаблоном має вигляд:

$$D = \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_i (x_i - \bar{x})^2} \cdot \sqrt{\sum_i (y_i - \bar{y})^2}},$$

де x – середнє значення пікселів маркера, а y – середнє значення пікселів шаблону, а підсумовування відбувається за відповідними значеннями пікселів.

Рисунок Ж.6 – Слайд 6

Порівняльна оцінка технологій

Таблиця 1 – Технології доповненої реальності зі значеннями критеріїв вибору

	Наявність маркерного відстеження	Сумісність та підтримка пристроїв	Точність відстеження	Простота інтеграції	Продуктивність та швидкість реагування	Середовище розробки та підтримка	Наявність у тарифному плані безкоштовної опції
Wikitude	Без маркерів	Android та iOS	Висока	Висока	Висока	Широка	Обмежена
ARKit	На основі маркерів	iOS	Висока	Помірна	Висока	Широка	Обмежена
ARCore	Без маркерів	Android	Висока	Помірна	Висока	Широка	Обмежена
Vuforia	На основі маркерів	Android та iOS	Висока	Висока	Висока	Широка	Безкоштовна
Kudan	На основі маркерів	Android та iOS	Помірна	Висока	Висока	Помірна	Безкоштовна
Unity AR Foundation	Обидва	Android та iOS	Висока	Помірна	Висока	Широка	Обмежена

Розподіл балів за кожним критерієм

Наявність маркерного відстеження:

- обидва: 4 бали
- на основі маркерів: 3 бали
- без маркерів: 1 бал

Продуктивність та швидкість реагування:

- висока: 3 бали
- помірна: 2 бали
- достатня: 1 бал

Сумісність та підтримка пристроїв:

- Android та iOS: 3 бали
- Android або iOS: 1 бал

Середовище розробки та підтримка:

- широка: 3 бали
- помірна: 2 бали
- обмежена: 1 бал

Точність відстеження:

- висока: 3 бали
- помірна: 2 бали
- прийнятна: 1 бал

Наявність у тарифному плані безкоштовної опції:

- безкоштовна: 3 бали
- обмежена: 1 бал

Простота інтеграції:

- висока: 3 бали
- помірна: 2 бали
- складна: 1 бал

Найважливіші критерії можна визначити за найвищими отриманими балами, що відображають їхню важливість в контексті вибору технології доповненої реальності.

Рисунок Ж.7 – Слайд 7

Порівняльна оцінка технологій

Метод оцінки

Для проведення оцінки використовується метод лінійної адитивної згортки з ваговими коефіцієнтами за формулою:

$$Z^* = \max_{i=1,m} \sum_{j=1}^n \alpha_j \beta_j a_{ij},$$

де α_j – нормуючі множники, а β_j – вагові коефіцієнти.

Були пронормовані всі значення критеріїв, тобто вони були приведені із різних шкал до однієї в діапазоні [0; 1] із урахуванням min та max за формулою:

$$f = \frac{f_{\text{вимір}} - f_{\text{min}}}{f_{\text{max}} - f_{\text{min}}}$$

	Наявність маркерного відстеження	Сумісність та підтримка пристроїв	Точність відстеження	Простота інтеграції	Продуктивність та швидкість реагування	Середовище розробки та підтримка	Наявність у тарифному плані безкоштовної опції	Z*
Wikitude	0	1	1	1	1	1	0	0,6
ARKit	0,6	0	1	0,5	1	1	0	0,47
ARCore	0	0	1	0,5	1	1	0	0,35
Vuforia	0,6	1	1	1	1	1	1	0,92
Kudan	0,6	1	0,5	1	1	0,5	1	0,82
Unity AR Foundation	1	1	1	0,5	1	1	0	0,75
Вагові коефіцієнти	0,2	0,2	0,1	0,1	0,1	0,1	0,2	

Рисунок 2 – Технології доповненої реальності зі значеннями критеріїв у числовому форматі (результати)

За методом лінійної адитивної згортки, **Vuforia** виявилася **найкращою технологією** для реалізації віртуальної примірконої аксесуарів з використанням механізму розпізнавання маркерів. З **оцінкою 0.92** вона перевершує інші альтернативи за всіма оціненими критеріями.

Рисунок Ж.8 – Слайд 8

Програмна реалізація

9

Для розробки використовувалась **Vuforia Engine** у середовищі **Unity**, що значно покращує процес створення додатків доповненої реальності для різних пристроїв. Інтеграція пакету Vuforia Engine AR в Unity дозволяє розташовувати віртуальні 3D-об'єкти на різних поверхнях за допомогою комп'ютерного зору. Після створення облікового запису на **порталі Vuforia Engine Developer Portal** та отримання ліцензійного ключа, розробка AR-додатків стає більш ефективною.

У проєкті використовуються **бінарні маркери Data Matrix**, які швидко та надійно розпізнаються Vuforia в режимі реального часу. Це забезпечує точне відстеження віртуальних аксесуарів, надаючи користувачам реалістичний та безперервний досвід. Використання **Target Manager для створення бази даних зображень-цілей** покращує продуктивність алгоритмів комп'ютерного зору. Завантажені зображення імпортується до проєкту Unity для подальшої роботи.

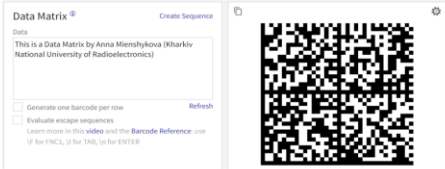


Рисунок 3 – Згенерований бінарний маркер: закодований текст (ліворуч) та двовимірний матричний штрих-код (праворуч)

Рисунок Ж.9 – Слайд 9

Програмна реалізація

10



Type: Image

Status: Active

Target ID: 7e4194f96ce14b189f15f3b6e4dde71a

Augmentable: ★★★★★

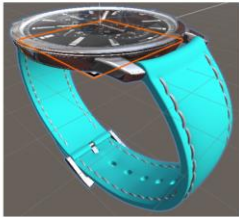


Рисунок 4 – Ефективність розпізнавання двовимірного матричного штрих-коду

Рисунок 6 – Накладання моделі годинника на об'єкт зображення (маркер)

Database

Type: Device

Targets (2)

Image	Target Name	Type	Rating (1)	Status
	barcode4	Image	★★★★★	Active
	barcode3	Image	★★★★★	Active
	barcode2	Image	★★★★★	Active
	barcode1	Image	★★★★★	Active

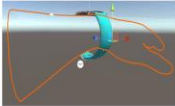
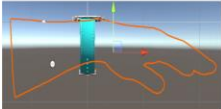


Рисунок 5 –База даних зображень штрих-коду

Рисунок 7 – Застосування матеріалу оклюзії до моделі руки (на прикладі годинника з різних сторін) та без оклюзії

Рисунок Ж.10 – Слайд 10

Проведення експериментальних досліджень

11

Експеримент 1: Вплив якості камери на розпізнавання маркерів
Мета: перевірити, як різна якість камери впливає на точність розпізнавання маркерів та час накладання аксесуара.
Метод: використовували одну камеру, накладаючи на неї різну кількість шарів клейкої стрічки для імітації різної якості зображення.

Експеримент 2: Вплив умов освітлення на розпізнавання маркерів
Мета: дослідити, як різні умови освітлення впливають на точність розпізнавання маркерів та час накладання аксесуара.
Метод: використовувалися різні освітлювальні прилади, включаючи природне світло в різний час доби (ранок, день, вечір) та штучне світло (ліхтарик телефону) з різними рівнями яскравості.

Експеримент 3: Вплив переміщення та положення користувача на розпізнавання маркерів
Мета: оцінити вплив швидкості руху руки користувача у віртуальній примірочній на точність розпізнавання маркерів та час накладання аксесуара.
Метод: рука користувача переміщалася зі сторони в сторону з різною швидкістю, щоб оцінити здатність системи точно відстежувати маркери.

Рисунок Ж.11 – Слайд 11

Аналіз результатів експериментальних досліджень

12

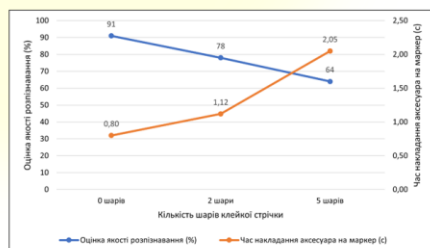


Рисунок 8 – Результати першого експерименту

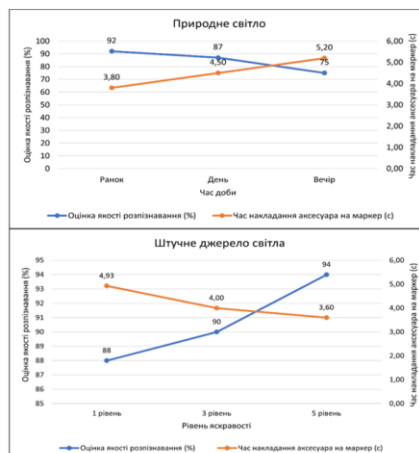


Рисунок 9 – Результати другого експерименту

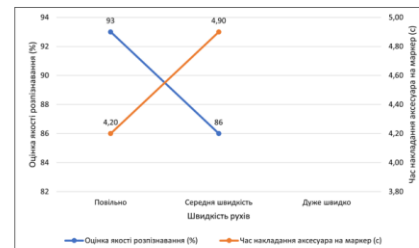


Рисунок 10 – Результати третього експерименту

Рисунок Ж.12 – Слайд 12

Підсумки

13

Реалістичність та корисність отриманих результатів

Дослідження технологій доповненої реальності для програмної реалізації віртуальної примірконої аксесуарів за допомогою штучного інтелекту принесло значні результати. За допомогою об'єктивних методів оцінки та порівняння різних технологій, була визначена оптимальна технологія для розробки додатку.

Експериментальні дані підтвердили вплив якості камери, умов освітлення та швидкості руху на точність та ефективність розпізнавання маркерів. Це має практичне значення для подальшого вдосконалення систем, оскільки ідентифікація ключових факторів дозволяє спрямувати зусилля на оптимізацію та покращення цих аспектів.

Можливий розвиток досліджень

Подальші дослідження можуть бути спрямовані на вдосконалення методів розпізнавання маркерів, використання глибокого навчання для коригування зображень у реальному часі, а також на розвиток інтерактивності через розпізнавання жестів та голосу. Персоналізовані рекомендації та інтеграція датчиків навколишнього середовища можуть додатково покращити користувацький досвід. Оптимізація для роботи в різних умовах освітлення та швидкостей руху також є важливим напрямком розвитку.

Висновок

Використання технології доповненої реальності Vuforia та штучного інтелекту для розробки віртуальної примірконої показало високий рівень ефективності. Завдяки ретельному аналізу, реалізації методології та експериментальним дослідженням були отримані практичні рекомендації для покращення функціональності та реалістичності віртуальної примірконої.

Рисунок Ж.13 – Слайд 13