

Вінницький національний технічний університет

Факультет інформаційних технологій та комп'ютерної інженерії

Кафедра програмного забезпечення

Пояснювальна записка

до магістерської кваліфікаційної роботи

магістр

(освітньо-кваліфікаційний рівень)

на тему: Розробка методу інтерактивної візуалізації тривимірних об'єктів у
реальному середовищі

Виконав: студент II курсу

групи 2ПІ-18М

спеціальності

121 – Інженерія програмного забезпечення

(шифр і назва напрямку підготовки, спеціальності)

Білик В. О.

(прізвище та ініціали)

Керівник: к.т.н., доц. каф. ПЗ

Бабюк Н. П.

(прізвище та ініціали)

Рецензент: к.т.н., доц. каф. КН

Іванчук Я. В.

(прізвище та ініціали)

Вінницький національний технічний університет
Факультет інформаційних технологій та комп'ютерної інженерії
Кафедра програмного забезпечення
Освітньо-кваліфікаційний рівень – магістр
Спеціальність 121 – Інженерія програмного забезпечення

ЗАТВЕРДЖУЮ
Завідувач кафедри ПЗ
Романюк О. Н.
“ ____ ” _____ 2019 року

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

Білику Владиславу Олеговичу

1. Тема роботи – розробка методу інтерактивної візуалізації тривимірних об'єктів у реальному середовищі.

Керівник роботи: Бабюк Н. П., к.т.н., доцент кафедри ПЗ, затверджені наказом вищого навчального закладу від “ ____ ” _____ 2019 року № ____

2. Строк подання студентом роботи _____

3. Вихідні дані до роботи: операційна система Windows 7/8/10

Середовище виконання – Unity.

Мова програмування – C#.

4. Зміст розрахунково-пояснювальної записки: обґрунтування доцільності розробки; варіантний аналіз та обґрунтування вибору засобів реалізації додатку; розробка методу та структури програмного засобу для інтерактивної візуалізації тривимірних об'єктів; тестування роботи додатку; економічна частина.

5. Перелік графічного матеріалу: мета роботи, об'єкт та предмет дослідження; основні задачі дослідження; наукова новизна, практичне значення; засоби реалізації; розробка методів; приклад вигляду розробленої програми; висновки, публікації, апробації.

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1-4	Бабюк Н. П., к.т.н., доцент кафедри ПЗ		
5	Бальзан М. В., к.е.н., доцент кафедри ЕПВМ		

7. Дата видачі завдання _____

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів магістерської кваліфікаційної Роботи	Строк виконання етапів роботи	Примітка
1	Аналіз проблеми, обґрунтування актуальності розробки та постановка задачі		Вик.
2	Розробка методу інтерактивної візуалізації та алгоритмів роботи додатку		Вик.
3	Розробка програмного продукту		Вик.
4	Тестування роботи додатку		Вик.
5	Економічна частина		Вик.
6	Оформлення матеріалів до захисту магістерської кваліфікаційної роботи		Вик.

Студент _____
(підпис)

Білик В.О.
(прізвище та ініціали)

Керівник магістерської кваліфікаційної роботи _____
(підпис)

Бабюк Н. П.
(прізвище та ініціали)

АНОТАЦІЯ

У магістерській кваліфікаційній роботі проведено детальний аналіз методів і засобів візуалізації тривимірних об'єктів в реальному середовищі. Сформульовано мету досліджень – покращення працездатності та швидкодії інтерактивної візуалізації тривимірних об'єктів в реальному часі та в реальному середовищі.

Розроблені методи підвищення реалістичності при візуалізації тривимірних об'єктів засобами доповненої реальності, що відрізняються використанням комбінації статичного і динамічного освітлення з попередньої візуалізацією незалежної текстури затінення і статичної падаючої тіні об'єктів. Запропоновано підхід до зміни параметрів об'єктів доповненої реальності при візуалізації, що відрізняється введенням параметра матеріалу, прискорює взаємодію, що включає можливість зміни в реальному часі і забезпечує спрощення інтерфейсу користувача. Розроблено метод підготовки та зберігання тривимірних моделей віртуальних об'єктів для їх реалістичної візуалізації, що відрізняється структурою і форматом зберігання даних. Запропоновано авторський механізм і алгоритм маніпуляції віртуальними тривимірними об'єктами в просторі з використанням жестових методів введення, що відрізняються можливістю одночасного переміщення та обертання, а також швидкістю і стабільністю роботи.

Отримані в магістерській кваліфікаційній роботі наукові та практичні результати можна використати для побудови ефективних систем візуалізації тривимірних об'єктів.

ANNOTATION

In the master's qualification work a detailed analysis of methods and means of visualization of three-dimensional objects in a real environment. Research goal is formulated to improve performance and performance by interactively rendering 3D objects in real time and in the real world.

Techniques for enhancing realism in the visualization of three-dimensional objects by means of augmented reality are developed, characterized by the use of a combination of static and dynamic illumination with the preliminary visualization of the independent texture of shading and static incident shadows of objects. An approach to changing the parameters of augmented reality objects in the visualization is proposed, which differentiates the input of a material parameter, accelerates interaction, which includes the possibility of real-time changes and provides simplification of the user interface. The technique of preparation and storage of three-dimensional models of virtual objects for their realistic visualization has been developed, which differs in structure and format of data storage. The author's mechanism and algorithm for manipulation of virtual three-dimensional objects in space with the use of gestural input methods, which differ in the possibility of simultaneous movement and rotation, as well as speed and stability of work, are offered.

The master's qualifications obtained can be used to build effective visualization systems for three-dimensional objects.

ЗМІСТ

ВСТУП.....	8
1 АНАЛІЗ СТАНУ ПИТАННЯ ТА ПОСТАНОВКА ЗАДАЧІ ДОСЛІДЖЕННЯ	12
1.1 Аналіз особливостей сучасної візуалізації тривимірних об'єктів	12
1.2 Використання засобів доповненої реальності для візуалізації об'єктів в реальному середовищі.....	18
1.3 Порівняльний аналіз аналогів	23
1.4 Постановка задачі для розробки методу інтерактивної візуалізації тривимірних об'єктів у реальному середовищі	28
1.5 Висновки	28
2 РОЗРОБКА МЕТОДУ ІНТЕРАКТИВНОЇ ВІЗУАЛІЗАЦІЇ ТРИВИМІРНИХ ОБ'ЄКТІВ У РЕАЛЬНОМУ СЕРЕДОВИЩІ	29
2.1 Методи та алгоритми візуалізації засобами доповненої реальності	29
2.2 Розробка методу підготовки та зберігання тривимірних моделей об'єктів.....	38
2.3 Алгоритм маніпуляції віртуальними тривимірними об'єктами в просторі з використанням жестових методів введення	41
2.4 Розробка інтерфейсу програмного додатку.....	45
2.5 Висновки	48
3 РОЗРОБКА ПРОГРАМНОГО ЗАСОБУ ДЛЯ ІНТЕРАКТИВНОЇ ВІЗУАЛІЗАЦІЇ ТРИВИМІРНИХ ОБ'ЄКТІВ У РЕАЛЬНОМУ СЕРЕДОВИЩІ... ..	49
3.1 Варіантний аналіз і обґрунтування вибору засобів для реалізації програмного засобу	49
3.2 Реалізація процесу трекінгу	57

3.3 Розробка процесу зміни параметрів об'єкта при візуалізації в реальному часі.....	63
3.4 Висновки	68
4 ТЕСТУВАННЯ ПРОГРАМИ	69
4.1 Методи тестування	69
4.2 Інструкція тестування програми	72
4.3 Висновки	78
5 ЕКОНОМІЧНА ЧАСТИНА	79
5.1 Оцінювання комерційного потенціалу розробки	79
5.2 Прогнозування витрат на виконання науково-дослідної роботи та конструкторсько–технологічної роботи.....	83
5.3 Прогнозування комерційних ефектів від реалізації результатів розробки.....	88
5.4 Розрахунок ефективності вкладених інвестицій та період їх окупності	90
5.5 Висновки	94
ВИСНОВКИ	95
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ	96
ДОДАТКИ	99
Додаток А. Технічне завдання.....	100
Додаток Б. Лістинг програмного засобу	103
Додаток В. Ілюстративний матеріал до захисту магістерської кваліфікаційної роботи	107

ВСТУП

Обґрунтування вибору теми дослідження. В сучасному світі комп'ютерна графіка використовується у великій кількості сфер людської графічної діяльності, таких як цифрова графіка, лазерна графіка, цифровий живопис і фотографія, практично завжди її використовують у фільмах, а також при створенні комп'ютерних ігор, при проектуванні приватних будинків, квартир[1].

Хоча комп'ютерна графіка слугує всього лише інструментом, її структура і методи засновані на передових досягненнях фундаментальних і прикладних наук: математики, фізики, хімії, біології, статистики, програмування і безлічі інших. Це зауваження є справедливим як для програмних, так і для апаратних засобів створення і обробки зображень на комп'ютері. Тому комп'ютерна графіка розвивається чи не найшвидше серед усіх галузей інформатики.

Завдання реалістичної візуалізації і генерації статичних і динамічних зображень об'єктів і сцен в реальному часі сьогодні, як і раніше, не втрачають своєї актуальності[2]. При цьому, в системах, що потребують вирішення цих завдань, виникає необхідність ефективного застосування і поєднання як методів реалістичної тривимірної графіки, що використовують традиційний механізм візуалізації, так і методів інтерактивної візуалізації в реальному часі. Це породжує нові напрямки прикладних досліджень в області створення ефективних архітектур програмно-апаратних комплексів для вирішення завдань реалістичної об'ємної візуалізації.

Програмі тривимірного моделювання зазвичай потрібно багато часу для створення кожного кадру, часом кілька годин для створення зображення з максимально можливою якістю. Залежно від складності сцени і параметрів, візуалізація зображення може бути виконана за кілька секунд, хвилин, годин або навіть днів. Часом можна задати таку складну сцену і параметри візуалізації, що комп'ютер ніколи не зможе її виконати.

В інтерактивному середовищі, наприклад грі або візуальному моделюванні, візуалізація декількох кадрів в секунду виконується в реальному часі. Такі ігри, як Quake або Unreal, мають дуже потужний механізм виконання, який здатний анімувати і перемальовувати безліч об'єктів в реальному часі. Основна мета такого механізму – отримати зображення максимальної якості з максимально можливою кількістю кадрів в секунду.

Візуалізований продукт має ряд переваг в порівнянні з 3D-фотографією: немає переривань зображення, можна переглядати з різних сторін, накладати будь-яку логіку. Актуальність дослідження полягає у тому, що для візуалізації об'єктів створюється візуальний конфігуратор, де користувач зможе побачити результат зібраного з різних опцій продукту, особливістю якого буде інтерактивність додавання і налаштування елементів для окремих зображень або всього продукту: від точок до полігональних областей.

Зв'язок роботи з науковими програмами, планами, темами. Робота виконувалася згідно плану виконання наукових досліджень на кафедрі програмного забезпечення.

Мета та завдання дослідження. Основною метою роботи є покращення працездатності та швидкодії інтерактивної візуалізації тривимірних об'єктів у реальному часі та в реальному середовищі за рахунок використання засобів доповненої реальності.

Основними задачами дослідження є:

- провести порівняльний аналіз методологічних, алгоритмічних і програмних засобів;
- покращити метод інтерактивної візуалізації у реальному часі;
- удосконалити метод підвищення реалістичності відображення тривимірних об'єктів при візуалізації у реальному часі;
- розробити метод зміни параметрів об'єктів при візуалізації в реальному часі;
- розробити метод підготовки та зберігання тривимірних моделей для покращення швидкодії реалістичної візуалізації на мобільних пристроях;

- розробити алгоритм маніпуляції віртуальними тривимірними об'єктами в просторі для підвищення інтерактивності візуалізації;
- сформулювати рекомендації з проектування інтерфейсу користувача для візуалізації.

Об'єкт дослідження – процес інтерактивної візуалізації тривимірних об'єктів у реальному середовищі.

Предмет дослідження – методи та засоби візуалізації тривимірних об'єктів у реальному середовищі.

Методи дослідження.

- методи проектування програмного забезпечення для виконання планування розробки додатку;
- методи теорії алгоритмів для розробки алгоритмів і розробки програмного забезпечення;
- комп'ютерне моделювання для аналізу та перевірки достовірності отриманих теоретичних положень.

Наукова новизна отриманих результатів.

1. Подальшого розвитку отримав метод підвищення реалістичності при візуалізації тривимірних об'єктів засобами доповненої реальності, що відрізняється від відомого використанням комбінації статичного та динамічного освітлення, що дозволяє виконувати попередню візуалізацію статичної падаючої тіні об'єктів.

2. Вперше розроблено метод зміни параметрів об'єктів доповненої реальності при візуалізації, особливістю якого є введення параметру матеріалу, що дає можливість змінювати параметри об'єктів в реальному часі та забезпечує спрощення інтерфейсу користувача.

Практична цінність отриманих результатів. Практична цінність одержаних результатів полягає в тому, що на основі отриманих у магістерській кваліфікаційній роботі теоретичних положень розроблено програмні засоби для інтерактивної візуалізації тривимірних об'єктів у реальному середовищі.

Особистий внесок здобувача. Усі наукові результати, викладені у магістерській кваліфікаційній роботі, отримані автором особисто. У праці [3], опублікованій у співавторстві, здобувачу належить запропонований метод візуалізації тривимірних об'єктів у реальному середовищі засобами доповненої реальності та розроблені модулі програмного продукту.

Апробація результатів. Основні положення й результати досліджень доповідалися й обговорювалися на XII міжнародній науково-практичній конференції «Інформаційні технології і автоматизація – 2019», – 17-18 жовтня 2019 р.

Публікації. Основні результати досліджень опубліковано в 1 науковій праці, що входить у матеріали конференції «Інформаційні технології і автоматизація – 2019».

1 АНАЛІЗ СТАНУ ПИТАННЯ ТА ПОСТАНОВКА ЗАДАЧІ ДОСЛІДЖЕННЯ

1.1 Аналіз особливостей сучасної візуалізації тривимірних об'єктів

Останнім часом візуалізацію часто не уявляють без комп'ютерних технологій, тому визначення візуалізації часто пов'язують саме з комп'ютерною обробкою даних і їх відображенням засобами комп'ютера.

Візуалізація – це інструмент або метод інтерпретації даних зображення, які розміщені в комп'ютері, а також генерації зображень з складними багатовимірними наборів даних. Візуалізація – це метод обчислень. Він перетворює символічне подання до геометричного, дозволяючи дослідникам спостерігати свої моделі і обчислення. Візуалізація пропонує метод, який дозволяє побачити невидиме[21].

Розвиток комп'ютерної графіки став можливим завдяки розвитку електроніки, електротехніки, появи телебачення і екранів, які використовують електронно-променеві трубки. У 1950-х роках, такі проекти як Whirlwind і SAGE стали використовувати екрани на основі променевих трубок як пристрою відображення, а також взаємодії з користувачем. Для введення графічних даних в комп'ютер застосовувалося світлове перо.

Реальні прориви в комп'ютерній графіці відбулися в 1970-і роки, змістивши акценти в комп'ютерній графіці в сторону розвитку її реалістичності.

Рендеринг – це процес створення зображення деякої моделі за допомогою комп'ютерної програми. Модель – це опис тривимірних об'єктів строго певною мовою або структурою даних[14]. Вона містить геометрію, точку огляду, інформацію про текстури, освітлення і методи розрахунку кольорів.

Основні техніки рендеринга включають:

- растеризація і порядковий рендеринг, які переводить модель з уявлення, в якому об'єкти складаються з примітивів у растрове зображення;
- рейкастінг, при якому сцена будується на основі перетину променів з поверхнею, що відображається, використовується для візуалізації твердотільних об'єктів і тривимірних скалярних полів;
- рейтрейсінг, алгоритм при якому відстежує шлях променів і враховуються ефекти впливу об'єктів на промінь, дозволяє досягти високого ступеня реалістичності зображення і змодельовати більшість оптичних ефектів;
- метод випромінюваності, що є методом вирішення рівняння рендеринга для сцен з дифузним відображенням світла об'єктами.

Порядковий рендеринг – це алгоритм визначення видимих поверхонь в тривимірній комп'ютерній графіці, який працює на основі рядів, а не полігонів або пікселів[16]. Всі багатокутники, які повинні бути відображені, упорядковано спочатку по у координаті, з якої вони вперше з'являються. Потім кожен ряд або рядок розгортки зображення обчислюється, використовуючи перетин рядка розгортки з багатокутниками на початку впорядкованого списку, в той час як упорядкований список оновлюється для того, щоб викинути з нього невидимі багатокутники, в міру того, як активний рядок розгортки просувається до низу картинки.

Головною перевагою даного методу є те, що сортування вершин уздовж нормалі скануючої площини зменшує кількість порівнянь між ребрами. Іншою перевагою є те, що немає потреби переміщувати координати всіх вершин з основної пам'яті в робочу пам'ять – тільки вершини, що визначають ребра, які перетинаються з поточним рядком розгортки, повинні бути в активній пам'яті, і кожна вершина читається тільки один раз. Основна пам'ять часто є дуже повільною в порівнянні з кеш-пам'яттю процесора, тому відсутність необхідності по кілька разів звертатися до вершин в основну пам'ять може призвести до істотного прискорення.

Термін рейкастінг був вперше використаний в комп'ютерній графіці в 1982 році в статті Скотта Рота для опису методу рендеринга конструктивної твердотільної геометрії. Проте найперше алгоритм рейкастінга був застосований в 1968 році Артуром Аппелем. Ідея, на якій заснований рейкастінг – це відслідковувати промені, випущені з камери, по одному для кожного пікселя, і знаходити найближчий об'єкт на шляху цього променя. Цей об'єкт тоді і буде тим, який видно крізь цей піксель. Використовуючи властивості матеріалу об'єкта, а також джерела світла в сцені можна розрахувати колір об'єкта. При цьому приймається спрощене припущення про те, що якщо поверхня звернена до світла, то світло досягне цієї поверхні і не буде заблоковане або затінене. Колір поверхні обчислюється за допомогою традиційних моделей затінення тривимірної комп'ютерної графіки. Однією важливою перевагою, яким володів рейкастінг в порівнянні з більш старими алгоритмами порядкового рендеринга, була його можливість легко обробляти неплоскі поверхні і твердотільні об'єкти, такі як конуси і сфери.

Взагалі, термін рейкастінг може відноситись до різних проблем і методів:

- загальна проблема знаходження першого об'єкта, пересічного з променем;
- техніка видалення невидимих поверхонь, заснована на знаходженні перетинів променів, що випускаються з камери через всі пікселі зображення;
- нерекурсивний алгоритм рейтрейсінга, який випускає тільки первинні промені;
- метод прямого об'ємного рендеринга, який також називають об'ємний рейкастінг, при якому промінь проходить всередину об'єкта і досліджуване тривимірне скалярне поле семплюється по променю всередині об'єкта. Даний метод не породжує вторинних променів.

Хоча в ранній літературі терміни «рейкастінг» і «рейтрейсінг» були взаємозамінні, останнім часом їх намагаються розрізняти. Різниця полягає в

тому, що рейкастінг ніколи не породжує вторинні промені, в той час як алгоритми рейтрейсінга можуть робити це.

Рейтрейсінг – метод візуалізації зображень, при якому відслідковуються промені світла і моделюються ефекти при взаємодії променя з віртуальними об'єктами[18]. Даний метод здатний виробляти результати з дуже високим ступенем реалізму, зазвичай більш високим, ніж методи порядкового рендеринга, але має велике обчислювальне навантаження. Тому рейтрейсінг часто застосовується в додатках, де можливо підготувати кадр заздалегідь. Рейтрейсінг здатний імітувати велике число різних оптичних ефектів, таких як відображення, заломлення, розсіювання і ефекти дисперсії світла.

Рейтрейсінг працює шляхом відстеження променів від камери крізь кожен піксель віртуального екрану і обчислення кольорів об'єкта, видимого крізь цей піксель (рисунок 1.1). На відміну від рейкастінга, рейтрейсінг може породжувати додаткові промені при зіткненні променя з об'єктом. Таким чином, можна вважати що рейкастінг є найпростішим видом рейтрейсінга.

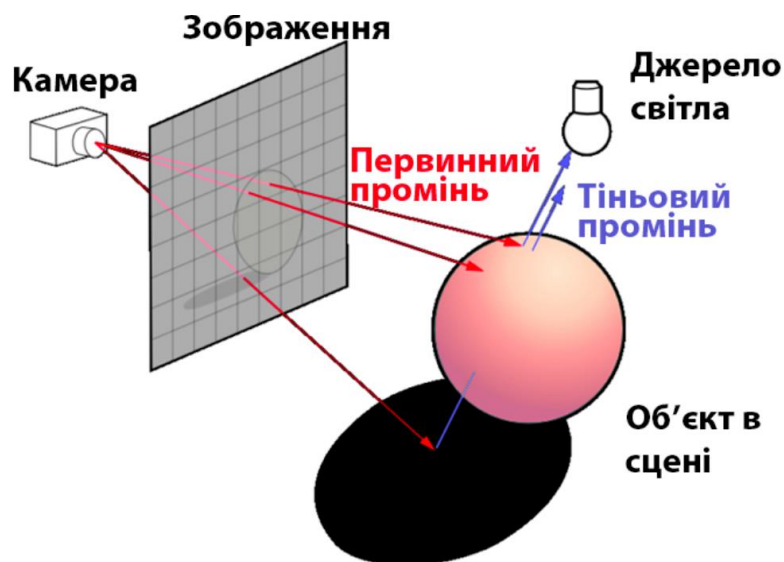


Рисунок 1.1 – Алгоритм роботи рейтрейсінгу

До 1979 року використовувався тільки рейкастінг, важливий прорив в цій області трапився завдяки роботі Тернера Уїттеда. Він запропонував наступне: коли промінь стикається з поверхнею, він може породити до трьох

нових типів променів: відображення, заломлення і тінь. Промінь відображення випускається в напрямку дзеркального відображення від поверхні. Найближчий об'єкт, який він перетне буде тим, що буде видно у відображенні. Промінь заломлення, проходить крізь прозорий матеріал, працює схожим чином, за винятком того, що він може бути вхідним або вихідним з матеріалу. Промінь тіні випускається у напрямку до кожного джерела світла. Якщо який-небудь непрозорий об'єкт знаходиться на шляху від поверхні до джерела світла, значить поверхня знаходиться в тіні і це джерело не висвітлює її. Такий рекурсивний рейтрейсінг додав більше реалістичності зображенням.

Z-буферизація – це управління координатами глибини зображення в тривимірній графіці, як правило здійснюється апаратно, але іноді і програмно. Коли об'єкт малюється, глибина згенерованого пікселя (z координата) записується в буфер (Z-буфер або буфер глибини). Як правило, цей буфер організований у вигляді двовимірного масиву, де кожен елемент відповідає пікселю екрану. Якщо інший об'єкт сцени повинен бути намальований в цьому ж пікселі, метод порівнює два значення глибини і перезаписує існуючий піксель якщо цей об'єкт знаходиться ближче до спостерігача. Вибране значення глибини потім записується в буфер, переписуючи старе. У підсумку, Z-буфер дозволить методу коректно відтворити звичайне сприйняття глибини – найближчий об'єкт закриває дальній. Винахід концепції Z-буфера часто приписують Едвіну Кетмеллу, хоча Вольфганг Штрасер описав цю ідею в своїй докторській дисертації в 1974 році за кілька місяців до винаходу Кетмелла.

Складність процесу графічного виведення зростає в міру наближення до деякого ідеалу – створення повної ілюзії природних, живих, реалістичних зображень. Зусилля багатьох вчених і інженерів усього світу спрямовані на розробку методів і засобів досягнення цієї мети. Тут найповніше відчувається зв'язок комп'ютерної графіки з природничими науками, з дисциплінами, присвяченими вивченню навколишнього світу. Наприклад, для створення реалістичних зображень потрібно брати до уваги закони оптики, які описують

світло і тінь, відображення і заломлення. Комп'ютерна графіка знаходиться на стику багатьох дисциплін і розділів науки.

Проте, для інтеграції тривимірних об'єктів в реальне середовище традиційними засобами 3D-графіки, потрібно розробити тривимірну сцену, що відтворює перспективу середовища, в яке інтегрується об'єкт [4]. Далі засобами редактора відбувається інтеграція об'єкта у фото. Все це вимагає кропіткої роботи спеціаліста та супроводжується затратами часу, коштів для розробки статичного зображення (рисунок 1.2).

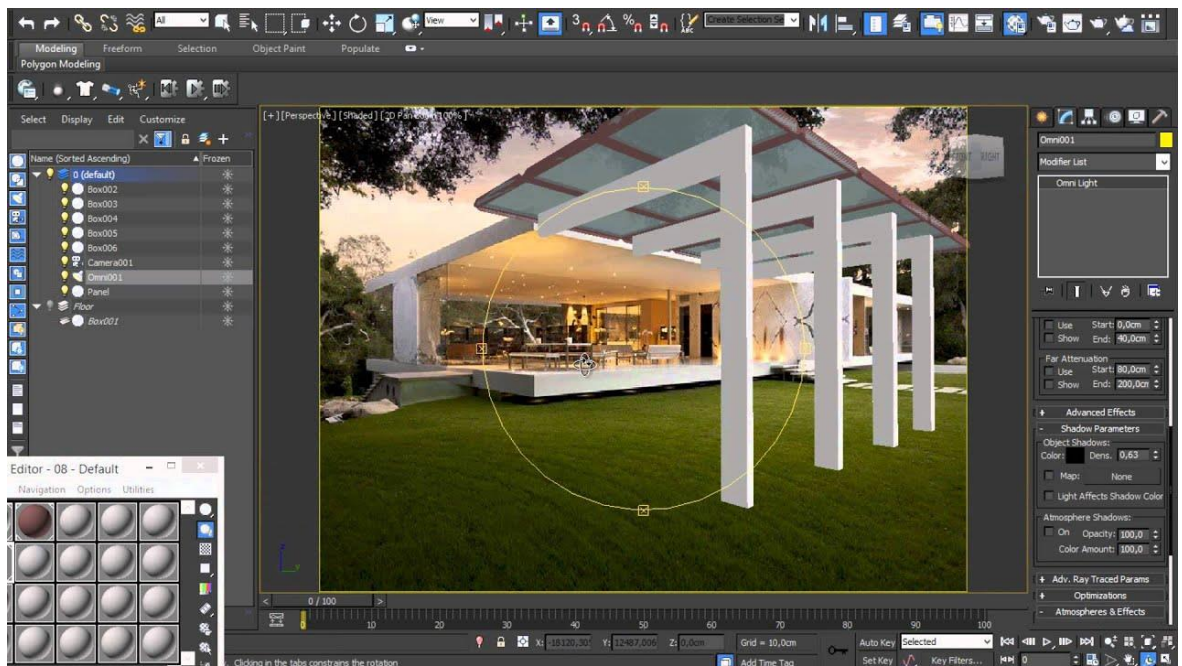


Рисунок 1.2 – Приклад розробки візуалізації для інтеграції тривимірного об'єкта

Однак, з появою технологій доповненої реальності завдання візуалізації тривимірних об'єктів в реальному середовищі може виконуватись на якісно новому рівні, що дозволяє з відносно меншими затратами праці отримувати візуалізацію в реальному часі, а не статичне зображення з певного ракурсу.

Доповнена реальність (також «розширена реальність») – це феномен просторово-часового континууму, який поєднує в собі об'єктивну і віртуальну реальність і що має низку специфічних якостей і властивостей, недоступних в об'єктивній і віртуальній реальності окремо [5]. У більш вузькому сенсі,

доповнену реальність можна визначити, як технологію інтеграції віртуальних об'єктів в реальний світ.

Концепція доповненої реальності пропонує більш досконалий, призначений для користувача інтерфейс для візуалізації за рахунок сукупності способів управління і візуалізації.

Доповнена реальність (ДР) пропонує унікальні способи відображення візуальної інформації, зокрема візуалізації тривимірних об'єктів [6]. Засобами ДР об'єкт може бути візуалізований безпосередньо в контексті його експлуатації. Прикладом може служити предмет меблів, візуалізований в інтер'єрі, рослини на присадибній ділянці, архітектурний об'єкт на місцевості, комунікації в стіні будівлі, результати ультразвукового сканування, спроектовані на пацієнта під час операції.

1.2 Використання засобів доповненої реальності для візуалізації об'єктів в реальному середовищі

Доповнена реальність (ДР) (англ. Augmented Reality, AR) – це феномен просторово-часового континууму, який поєднує в собі об'єктивну і віртуальну реальність і має низку специфічних якостей і властивостей, недоступних в об'єктивній і віртуальній реальності окремо[4].

Управління ракурсом об'єкта здійснюється природними рухами голови користувача або пристрою і є зрозумілим і ефективним. Спосіб візуалізації тривимірного об'єкту шляхом суміщення його зображення з навколишнім оточенням у відповідному ракурсі дозволяє краще сприймати об'єкт, його розміри (за умови відповідності масштабу) і, в деяких випадках, властивості матеріалів[5]. При цьому, в відміну від традиційних засобів візуалізації (ПЗ для 3d-моделювання), для візуалізації об'єктів засобами ДР не потрібно моделювання оточення.

В даний час доповнена реальність використовується в різних сферах[6]:

- в навігаційних додатках;

- в іграх: в 2004 була випущена гра для мобільних телефонів з назвою Mosquitos, що відображає на екрані телефону зображення з розташованої позаду нього камери, з накладеними на це зображення прицілом і величезними комарами, від яких «відстрілювався» гравець;

- у військовій техніці: в сучасних бойових літаках і вертольотах часто використовується нашоломна система індикації. Вона дозволяє пілоту отримувати найбільш важливу інформацію прямо на тлі спостережуваної їм обстановки, не відволікаючись на основну приладову панель;

- в медицині: в якості прикладу може розглядатися система BoneSim, що дозволяє імітувати операції на кістковій тканині;

- в проектуванні та дизайні: може використовуватися для поєднання реальних об'єктів і інструкцій по збірці і управління;

- в рекламі.

З точки зору взаємодії з користувачем додатки з використанням доповненої реальності можна розділити на наступні групи[11]:

1. Автономні (вони передбачають взаємодію з користувачем і слугують тільки для надання супровідних даних про об'єкт). Подібні додатки доповненої реальності можуть аналізувати об'єкти, що знаходяться в полі зору людини (камери пристрою), і видавати про них довідкову інформацію. Наприклад, користувач розглядає картину в музеї і за допомогою програми доповненої реальності отримує додаткові дані про художника, про долю картини, історії зображеного сюжету і т.п. Також системи цього типу застосовуються в медицині. Наприклад, система Gait Aid використовується для того, щоб шляхом використання віртуальних об'єктів давати мозку додаткову інформацію, яка допомагає координувати рухи. Дана система застосовується людьми з порушеннями опорно-рухового апарату.

2. Інтерактивні (допускають взаємодію з користувачем, який може налаштовувати тип накладання додаткового шару даних і отримувати різні відповіді з даного об'єкту). Очевидно, що такі системи передбачають наявність пристрою введення даних, в ролі якого може виступати сенсорний екран

мобільного пристрою або інші сенсорні датчики. Прикладом такого додатка доповненої реальності є «примірочні», де користувач взаємодіє з інтерфейсом, щоб вибирати одяг з наявного набору і шляхом накладення шарів отримувати власні зображення в різних вбраннях.

За ступенем мобільності можна виділити такі системи доповненої реальності:

- стаціонарні (призначені для роботи в одному місці і не передбачають якоесь переміщення);
- мобільні (їх використання має на увазі переміщення в просторі і роботу в динамічному режимі з різними об'єктами навколишнього реального світу).

За типом сенсорів можна виділити:

- геопозиційні (такі системи орієнтуються в першу чергу на сигнали систем позиціонування, наприклад GPS);
- оптичні (дані системи працюють із зображенням, отриманим з однієї або декількох камер).

Для якісного розміщення віртуального об'єкта в реальному середовищі необхідно вирішити задачу точної локалізації поточного розміщення спостерігача відносно даного місця[20]. В процесі локалізації вводиться локальна система координат і робиться її прив'язка до глобальної системи координат – визначається перетворення, зв'язує їх. У заданій локальній системі координат в подальшому здійснюється трекінг оточення – визначення поточного стану та орієнтації пристрою у реальному часі. Система трекінгу, що підходить для побудови доповненої реальності, повинна мати високу точність і бути стійкою до швидких переміщень і поворотів пристрою.

У більшості існуючих додатків для мобільних пристроїв в якості засобів, що використовуються для локалізації, використовується датчик GPS і сенсори пристрою: гіроскоп, акселерометр і магнітометр. Але їх застосування без урахування візуальної інформації про оточення не дозволяє домогтися досить точної локалізації і трекінгу. Для отримання візуальної інформації про місце використовуються як звичайні камери, так і спеціальні камери, які дають

змогу "3D-зору": камери з сенсорами глибини і стерео-камери. З розвитком технологій і підвищенням продуктивності мобільних пристроїв стає можливим застосування більш складних алгоритмів трекінгу для якісної локалізації з використанням інформації з камери пристрою.

Можна розглядати різні варіанти завдання візуальної локалізації в залежності від вихідної інформації про сцену і використовуваних ознак. Для більш простої і стійкої локалізації часто використовуються спеціальні маркери – штучні об'єкти, які легко відрізняються від решти оточення. Будемо називати природним оточення, яке не містить таких маркерів.

Трекінг по відео камері мобільного пристрою в природному оточенні є однією з найскладніших завдань в даній області і становить найбільший інтерес для дослідження, оскільки це робить можливим більш широке поширення даної технології. Як карти природного оточення найчастіше виступає розріджений хмара точок з дескрипторами або тривимірна модель оточення. техніки побудови доповненої реальності в природному оточенні можна розділити на дві основні категорії: використовують заздалегідь побудовану модель оточення і працюють без апріорної інформації про сцену, отримуючи необхідну інформацію про неї вже в процесі роботи. Також можливий варіант расширяемого трекінгу, коли є карта тільки маленького ділянки оточення (наприклад конкретний об'єкт на сцені), а інші частини добудовуються за допомогою її розширення.

Алгоритми трекінгу невідомого оточення дозволяють будувати карту сцени і на її основі виробляти трекінг в локальній системі координат. При цьому дана система координат визначається в процесі ініціалізації системи і ніяк не пов'язана зі світовими координатами. Для деяких додатків хочеться мати можливість точного геопозиціонування, тобто прив'язки побудованої локальної системи координат оточення до глобальних координатами. Прикладом такого додатка може бути проведення екскурсій по найвідоміших місцях з автоматичним додаванням додаткової інформації поруч з визначними пам'ятками.

Необхідно відзначити, що вдосконаленню піддаються не тільки методи введення інформації, а й методи її виведення. Так, спостерігається тенденція збільшення щільності точок дисплеїв: починають отримувати поширення дисплеї, на яких людське око все менше розрізняє окремі пікселі. З ростом щільності точок дисплея, очікується новий виток розвитку головних дисплеїв: шоломів віртуальної реальності і очок доповненої реальності. Дослідження по використанню таких пристроїв робляться ще з 1960-х років, проте вони ще не набули масового поширення.

Однак на сьогоднішній день є всі передумови входження концепції доповненої реальності в повсякденне життя людини. Стає зрозуміло, що її застосування не обмежується рекламними додатками, але має велике практичне значення, в тому числі в галузі вдосконалення інтерфейсу користувача шляхом більш простого і природного доступу до інформації.

Доповнена реальність почала отримувати поширення з підвищенням характеристик дисплеїв, камер, і обчислювальних можливостей мобільних пристроїв. Сучасні смартфони і планшетні комп'ютери надають можливості по застосуванню доповненої реальності в різних областях.

Першим сучасним пристроєм доповненої реальності можна назвати Microsoft HoloLens, обробка інформації відбувається за допомогою вбудованого чіпа, можливість якого здатна обробляти гігабайти даних на самому пристрої. Також ці окуляри є найпопулярнішими при згадці доповнена реальність, але, на жаль, цей пристрій до цих пір знаходиться на доопрацюванні та користувалися їм лише журналісти і розробники. Більшість відгуків про HoloLens позитивні і описують як незабутній досвід використання. Ще одним пристроєм, який є підсумком прогресу в області доповненої реальності є Magic Leap. На відміну від HoloLens, ці окуляри підключаються до невеликого комп'ютера, який, як запевняє творець, по потужності дорівнює повноцінному ігровому комп'ютера. За словами розробника, окуляри можна використовувати для ігор, інтернет-серфінгу, онлайн покупок, відображення різної інформації, віртуальне спілкування і творчість.

1.3 Порівняльний аналіз аналогів

На сьогоднішній день існує декілька аналогів, у кожного з них є певні переваги та недоліки, нижче вказані деякі з них:

JigSpace – програмний додаток для дослідження та отримання інтерактивного досвіду в різних сферах науки[7] (рисунок 1.3).



Рисунок 1.3 – Вікно додатку JigSpace

JigSpace – це як цифрова енциклопедія, яка представляє нові концепції за допомогою 3D-моделей під назвою Jigs. Користувачі обирають тему – від "Як працює акумулятор" до винащувача Lego TIE до "Що таке серцевий стент"? – і відкриває його для перегляду. Потрібно націлити камеру пристрою на рівне місце, щоб розпочати досвід доповненої реальності. Після завершення цього кроку на екрані з'явиться 3D-модель, і діти можуть прокручувати ряд інформаційних екранів, щоб дізнатися більше на цю тему. На екранах можуть відображатися перерізи або додаткові види об'єкта чи місця розташування. Також можна збільшувати та зменшувати масштаб, обертати об'єкт та дивитись на нього зверху та знизу.

Можливість збільшувати масштаб і оглядати об'єкт чи бачити поперечний переріз предмета створює чудову можливість для навчання, але

розширена реальність все ще має обмеження. JigSpace, безумовно, втілює в життя деякі теми, особливо ті, які неможливо легко переглядати, як-от переріз серця або заблокована артерія. Можна перетворювати об'єкти, збільшувати та зменшувати масштаб зображення та бачити прогресування екранів, але це все ще не є інтерактивним у повній мірі: це як перегляд 3D-слайд-шоу, а не по-справжньому взаємодія з зображеннями. AR вимагає приміщення з достатньою кількістю навколишнього світла, щоб сфотографувати поверхню, а потім він орієнтує пристрій на цю поверхню.

Houzz - Home Design & Remodel – додаток для Android, що представляє собою справжню енциклопедію інтер'єрного дизайну, завдяки якій користувач зможе з легкістю вибрати дизайн, інтер'єр і ідею ремонту власної квартири або будинку[8].

Додаток містить більше 9 мільйонів фото у високій якості, які можна сортувати за стилем (лофт, сучасний, Шеббі шик, кантрі, фьюжен, модернізм, класичний, східний, морський, ретро, рустика, скандинавський, вікторіанський, еkleктика), приміщенню (кухня, ванна, спальня, вітальня, їдальня, екстер'єр, дитяча кімната, кабінет, коридор, гараж або навіть сарай) і розташуванню. Всі вибрані фото інтер'єрів можна зберігати в "альбоми ідей" і ділитися ними з друзями (рисунок 1.4).

Особливості додатку Houzz:

- Величезний список статей, присвячених дизайнерській і ремонтній тематиці.
- Інструмент Sketch, що дозволяє наочно побачити результат.
- Доброзичливий інтерфейс, перекладений на безліч мов.
- Роботи дизайнерів зі світовим ім'ям абсолютно безкоштовно.
- Сортування за категоріями і «розумний» пошук нових ідей.
- Величезна кількість фотографій високої якості.
- Майданчик для спілкування між користувачами.

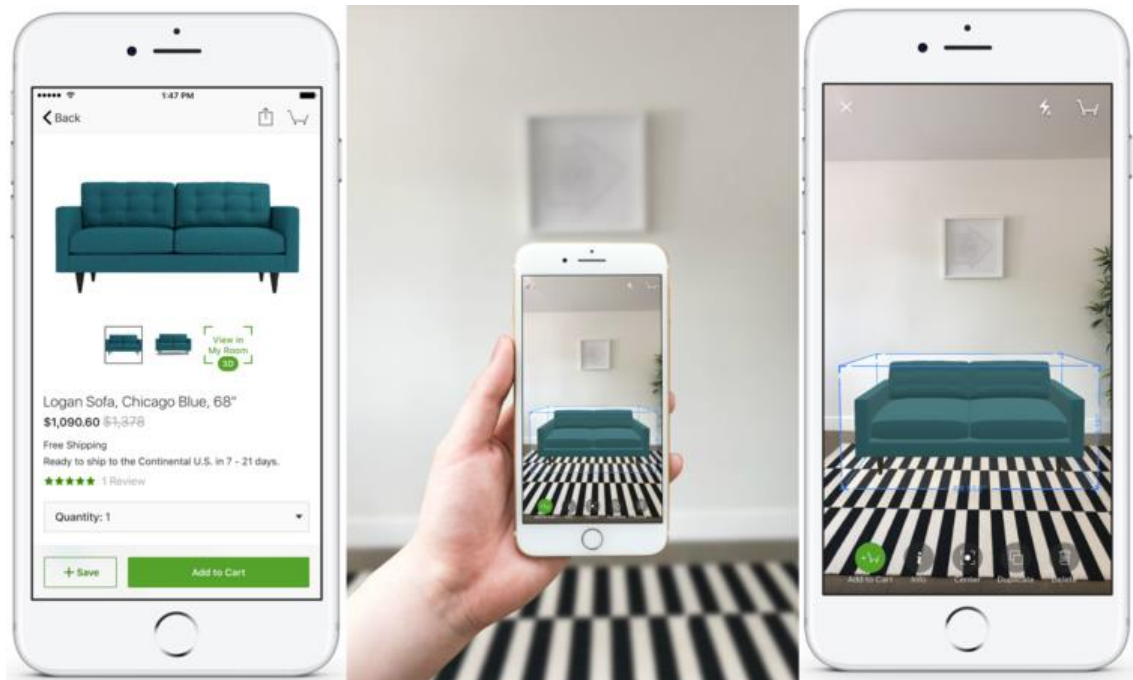


Рисунок 1.4 – Приклад роботи додатку Houzz

Holo – Holograms for Videos in Augmented Reality – мобільний додаток, за допомогою якого користувачі можуть додавати на фото і відео об'єкти доповненої реальності[9]. В основному власникам смартфонів доступні анімовані голограми людей, героїв фільмів і коміксів і тварин.

Функціонал Ноло пропонує анімовані голограми людей (знаменитості, спортсмени, танцюристи, мандрівники, моделі), тварин (тигри, горили, танцюючий Кінг Конг), а також героїв фільмів і коміксів. Кожна фігура представлена в реальному розмірі – тобто так, як би вона виглядала в цьому світі. При цьому можна змінювати їх розміри і переміщати навіть під час зйомки. Розробники відзначають, що технології Ноло дозволяють легко створювати високоточні, деталізовані і фотореалістичні голограми, які можна переглядати з будь-якого кута і відчувати небачене відчуття присутності у віртуальній, доповненої або змішаної реальності (рисунок 1.5).

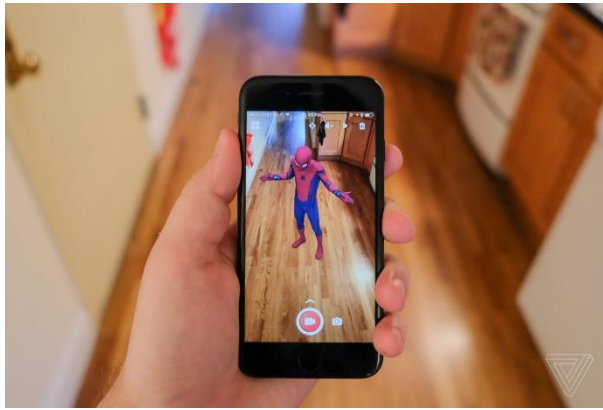


Рисунок 1.5 – Приклад роботи додатку Halo

Можливості додатку:

- Розміщення голограми улюбленого персонажа у свій реальний світ.
- Навігація навколо персонажа, ніби він знаходиться поруч.
- Голограми мають натуральний розмір, але їх можна змінити чи перемістити для налаштування сцени.
- Можливість зняти відео чи фотографію в портреті та пейзажі, ділитися з друзями та зберігати у своїй галереї.
- Вибір різних голограм, включаючи знакових персонажів, а також веселих оригінальних персонажів та тварин.
- Можливість завантажити нові голограми на свій телефон.

ІКЕА Place – мобільний додаток побудований на ARCore і дозволяє практично «розмістити» продукти ІКЕА у вашому просторі[10] (рисунок 1.6).



Рисунок 1.6 – Приклад роботи додатку ІКЕА Place

Додаток включає тривимірні моделі в реальних масштабах всього – від диванів та крісел до підніжок та журнальних столиків. IKEA Place дає точне враження про розмір, дизайн та функціональність меблів у вашому будинку, щоб користувач міг оцінити об'єкти, які планує придбати.

Проаналізувавши усі аналоги, визначено їхні переваги та недоліки, які враховувались при створенні власного програмного засобу «Interactive Visualization» (табл. 1.1).

Таблиця 1.1 – Порівняльні характеристики програмних продуктів

Критерій	JigSpace	Houzz	Holo	IKEA Place	Interactive Visualization
Зручний користувацький інтерфейс	-	+	+	+	+
Можливість маніпулювати об'єктами, що візуалізуються	-	+	+	+	+
Зміна властивостей матеріалів тривимірних об'єктів	-	-	-	-	+
Безкоштовний доступ до всіх функцій додатку	-	+	-	-	+

Таблиця порівняльних характеристик показала, що розробка програмного засобу є доцільною. В результаті отримуємо продукт, що покриває недоліки існуючих рішень.

1.4 Постановка задачі для розробки методу інтерактивної візуалізації тривимірних об'єктів у реальному середовищі

Після аналізу стану сучасної візуалізації тривимірних об'єктів та порівняння з аналогами, було визначено наступні завдання, які необхідно виконати для розробки методу:

- визначити найбільш ефективний спосіб візуалізації тривимірних об'єктів в реальному середовищі;
- розробити метод інтерактивної візуалізації тривимірних об'єктів;
- розробити інтерфейс програмного засобу, який буде легким для розуміння та інтуїтивним користувачу;
- розробити структурну схему інтерфейсу додатку;
- провести тестування програмного засобу.

1.5 Висновки

В даному розділі було розглянуто аналіз стану сучасної візуалізації тривимірних об'єктів. Також було проаналізовано стан вирішення даного питання шляхом розгляду аналогів та їх порівняння між собою, та розроблюваним методом. В результаті порівняння було відображено доцільність розробки магістерської кваліфікаційної роботи та розроблено основні завдання, які необхідні для розробки методу.

2 РОЗРОБКА МЕТОДУ ІНТЕРАКТИВНОЇ ВІЗУАЛІЗАЦІЇ ТРИВИМІРНИХ ОБ'ЄКТІВ У РЕАЛЬНОМУ СЕРЕДОВИЩІ

2.1 Методи та алгоритми візуалізації засобами доповненої реальності

Найважливішим компонентом доповненої реальності поряд з трекінгом є візуалізація об'єктів, яка зазвичай здійснюється засобами тривимірної комп'ютерної графіки. Розглянемо основні поняття даної дисципліни.

Тривимірна графіка (3D, 3 Dimensions) – розділ комп'ютерної графіки, сукупність прийомів і інструментів (як програмних, так і апаратних), покликаних забезпечити просторово-часову безперервність одержуваних зображень.

Поверхні тривимірних об'єктів в комп'ютерній графіці в більшості випадків зображуються з допомогою деякої множини трикутників, з'єднаних один з одним своїми вершинами і загальними сторонами. Такі множини називаються трикутними розбиттями (triangular meshes).

Трикутне розбиття (або мережа) може бути задано перерахуванням трикутників (трійок вершин), з яких воно складається, і координатами вершин в тривимірному просторі. Багато програм, що працюють з тривимірною графікою, зберігають додаткову інформацію, пов'язану з вершинами, сторонами і поверхнями трикутників, для підтримки текстур, тіней, анімації та інших операцій. Найчастіше для прив'язки параметрів поверхні використовують вершини – на них задають властивості матеріалу поверхні, координати текстури, інтенсивність випромінювання. Ці параметри потім лінійно інтерполюються на кожному трикутнику для задавання безперервної функції на всій моделюється поверхні.

Для передачі зовнішнього вигляду об'єктів, які моделюються, для поверхонь застосовують текстури. Під текстурою розуміють специфічні зміни тону (кольори) в зображенні об'єкта або його певної частини. До текстур відносять також наявність на зображенні характерних ліній.

Текстура може представляти впорядковану зміну тону у вигляді геометрично правильних або майже правильних малюнків. Подібну текстуру мають, наприклад, цегляна кладка, кахельне облицювання, спеціально розфарбовані поверхні: шахівниця, пішохідний перехід і т.д. Інший тип – стохастична текстура. Вона властива природним об'єктам і, як правило, є наслідком шорсткості спостережуваних об'єктів.

Найважливішу роль в тривимірній графіці грає освітлення. Для його симуляції застосовують моделі джерел освітлення, моделі поверхонь і інші засоби.

Існує 4 стандартних типи джерел освітлення:

1. Розсіяне (ambient).
2. Спрямоване (directional).
3. Точкове (point).
4. Конусне (spot).

Розсіяне (або основне) світло являє собою низькоінтенсивне світло, яке обумовлюється відображенням світла, випромінюваного будь-якими джерелами освітлення, від найближчих поверхонь. Його використання дозволяє уникнути маси розрахунків відбитого світла. Розсіяне світло застосовується для завдання загальної яскравості, наприклад, сцени. Він поширюється на всіх напрямках з однаковою інтенсивністю, рівномірно висвітлюючи всі поверхні.

Направлене джерело освітлення також відоме як нескінченно віддалене. Це одне джерело, що випускає промені в певному напрямку з нескінченно віддаленої точки. Дані джерела використовуються для моделювання таких об'єктів як сонце, коли промені паралельні один одному. Оскільки направлене джерело не має позиції в просторі, його інтенсивність не зменшується в міру віддалення, і світло поширюється на необмежену відстань.

Точкове джерело освітлення випромінює світло в усіх напрямках від конкретної точки в просторі. Інтенсивність випромінювання зменшується в

залежності від відстані відповідно до зворотного квадратичного закону, досягаючи нульового значення на заданій відстані.

Джерела конусного (прожекторного) типу відрізняються від точкових тільки тим, що мають обмежений кут випромінювання i , в результаті, конусоподібну область освітлення.

Крім того, на візуалізацію впливає власне характер поверхні об'єктів. Дифузна поверхня (diffuse shading) – це поверхня, для якої світло, що поширюється в довільному напрямку, висвітлює кожну її точку. Відповідно до цього, відбитий дифузний колір поверхні відбивається рівномірно у всіх напрямках. Таке відображення називається відображенням по Ламберту. Оскільки воно рівномірно у всіх напрямках, відсутня залежність від позиції спостерігача.

Відповідно до закону Ламберта кількість променевої енергії, що випромінюється елементом поверхні dS_1 в напрямку елемента dS_2 , пропорційно добутку кількості енергії, випромінюваної по нормалі dQ_n , на величину просторового кута $d\omega$ і, $\cos \varphi$ складеного напрямком випромінювання з нормаллю:

$$d^2Q_n = dQ_n \cdot d\omega \cdot \cos \varphi. \quad (2.1)$$

Є також проста залежність між силою світла, випромінюваного плоским розсіяним майданчиком dS в будь-якому напрямку, від кута φ між цим напрямком і перпендикуляром до dS :

$$I_\alpha = I_0 \cos \varphi. \quad (2.2)$$

Від'ємне значення скалярного добутку означає, що поверхня не освітлена, тому необхідно відкидати від'ємне значення скалярного добутку.

Модель Фонга є ще однією класичною моделлю освітлення. Вона являє собою комбінацію дифузійної складової (моделі Ламберта) і дзеркальної складової і працює таким чином, що крім рівномірного освітлення на матеріалі може ще з'являтися відблиск (рисунок 2.1). Місцезнаходження відблиску на об'єкті, освітленому по моделі Фонга, визначається із закону рівності кутів падіння і відбиття. Якщо спостерігач знаходиться поблизу кутів відбиття,

яскравість відповідної точки підвищується. Якщо об'єкт є ідеальним дзеркалом, то це буде єдиний напрямок, як і яскравість.

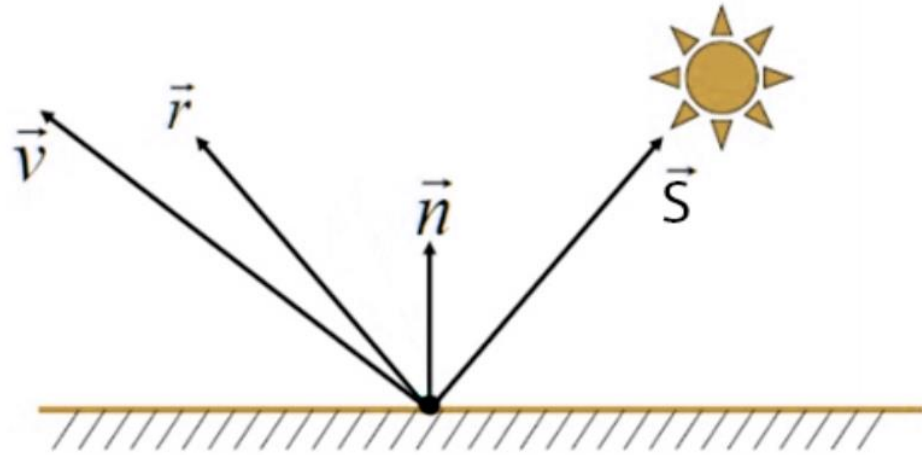


Рисунок 2.1 – Вектори для розрахунку моделі Фонга

Яскравість відбитого світла згідно моделі Фонга обчислюється за формулою:

$$L_0 = L_i k_s (r, v)^{k_l}, \quad (2.3)$$

де L_0 – вихідна яскравість, L_i – яскравість світла, k_s – дзеркальний коефіцієнт відбиття поверхні, (r, v) – кут між ідеально відбитим променем r і вектором на глядача v , k_l – експонента Фонга.

Для розрахунку об'єктів, які є просто блискучими, а не виключно дзеркальними, в дзеркальній складовій моделі Фонга необхідно розглянути не ідеально дзеркальний об'єкт, а об'єкт, що відображає світло в деякому конусі: це необхідно для отримання відблиску. Дельта функція замінюється на деяку функцію, плавно спадаючу до нуля по деякому закону при віддаленні променя від деякого заданого конуса. Формула для обчислення яскравості в точці O виглядає наступним чином:

$$L_0 = L_i (k_d (s, n) + k_s (r, v)^{k_l}), \quad (2.4)$$

де k_d – дифузний коефіцієнт відбиття поверхні, (s, n) – скалярний добуток між нормаллю в даній точці і напрямом на джерело світла.

2.1.1 Розробка методу для реалізації реалістичного освітлення об'єктів

Освітлення об'єкта грає найважливішу роль при його візуалізації. Правильна настройка освітлення вимагає спеціальних професійних навичок і досвіду. При побудові освітлення ставиться завдання максимально точно передати форму предметів і їх об'єм при перегляді на плоскому екрані пристрою.

У живописі і в фотографії при побудові освітлення натюрморту застосовуються різні типи освітлення, такі як заповнююче, контурне, моделюче світло та інші. У тривимірній комп'ютерній графіці також моделюються відповідні джерела освітлення.

Об'єкт може бути освітлений за допомогою декількох точкових джерел. Це може служити прикладом схеми динамічного освітлення, що прораховується кожен кадр. При цьому, в той час як при звичайній візуалізації застосовуються складні ресурсоємкі алгоритми розрахунку освітлення, що дозволяють отримати реалістичне зображення, візуалізація в режимі реального часу вимагає інших підходів.

При реалізації технології доповненої реальності на сучасних споживчих пристроях на сьогоднішній день не можна в реальному часі реалізувати складні алгоритми візуалізації, що враховують непряме освітлення, такі як трасування променів. Обчислювальні ресурси пристроїв не мають належного рівня продуктивності для розрахунку зображень з інтерактивною частотою кадрів.

Розроблена методика передбачає використання засобів статичного (попередньо розрахованого) освітлення: статичне затінення, і падаючу тінь. Це забезпечує істотне підвищення реалістичності і знижує складність постійних розрахунків динамічного освітлення (рисунки 2.2).

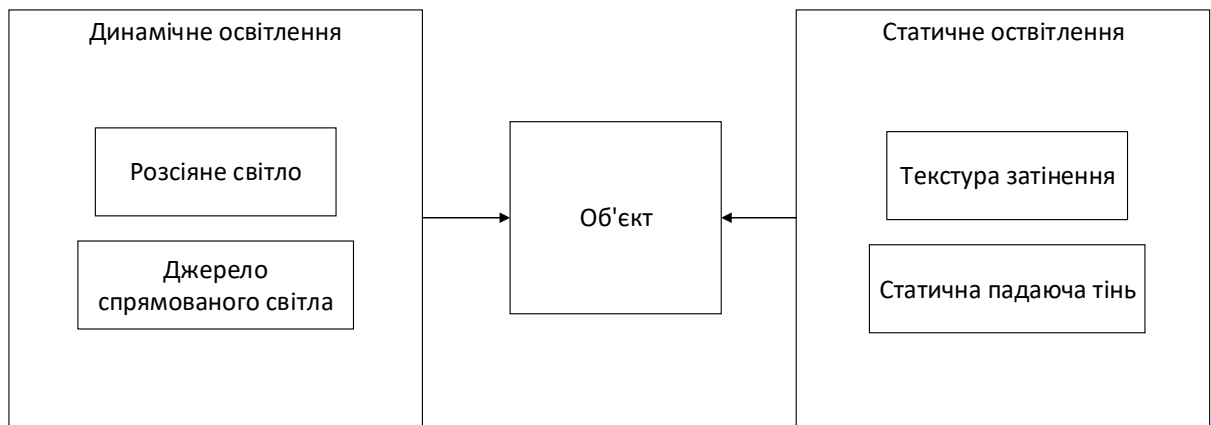


Рисунок 2.2 – Комбінація статичного і динамічного освітлення

Розроблено метод, при якому розрахунок і візуалізація освітлення поверхонь об'єкта проводиться заздалегідь на етапі підготовки тривимірної моделі і при подальшій візуалізації в реальному часі не потрібні додаткові ресурси. При такому підході на етапі розробки тривимірної моделі об'єкта до нього може бути застосовано освітлення будь-якої складності з використанням необмеженого числа джерел світла.

Велике значення має можливість попереднього розрахунку тіней, утворених в результаті дії непрямого освітлення (глобального освітлення, *global illumination*). Це істотно підвищує реалістичність візуалізації об'єкта. Розрахунок може бути виконаний із застосуванням різних складних методів візуалізації таких як трасування променів.

Окремо хотілося б зупинитися на методі *Ambient Occlusion* (рисунок 2.3). Даний метод дозволяє швидко прорахувати затінення об'єкта, використовуючи при цьому тільки інформацію про його геометрію.

Ще однією причиною застосування статичного освітлення є те, що різні об'єкти для найкращого сприйняття можуть вимагати різну кількість джерел освітлення різних типів і характеристик (наприклад, інтенсивності) по-різному розташованих. При статичному освітленні кожен об'єкт може бути налаштований незалежно, і при додаванні одночасно декількох об'єктів індивідуальне освітлення кожного не буде впливати на інші.

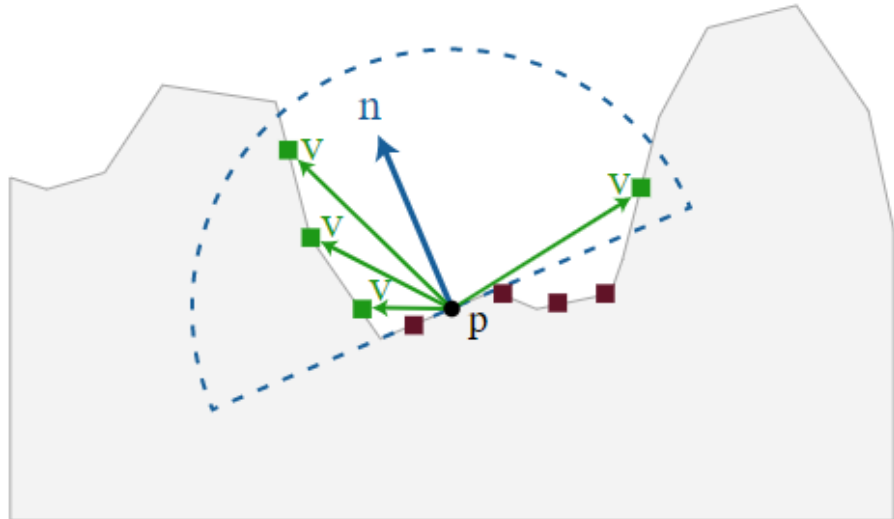


Рисунок 2.3 – Принцип роботи Ambient Occlusion

Крім простоти розрахунку перевага даного методу полягає в тому, що завдання складних методів візуалізації – максимально точно змодельовати фізичні процеси освітлення об'єкта, враховуючи віддзеркалення світла від навколишніх об'єктів. При цьому оточення зазвичай також змодельовано на комп'ютері. При візуалізації в доповненій реальності навколо об'єкта немає змодельованого оточення і сенс у використанні складних методів візуалізації втрачається.

Ambient Occlusion дозволяє отримати базову нейтральну модель затінення, яка підходить до різних типів оточення (рисунок 2.4).

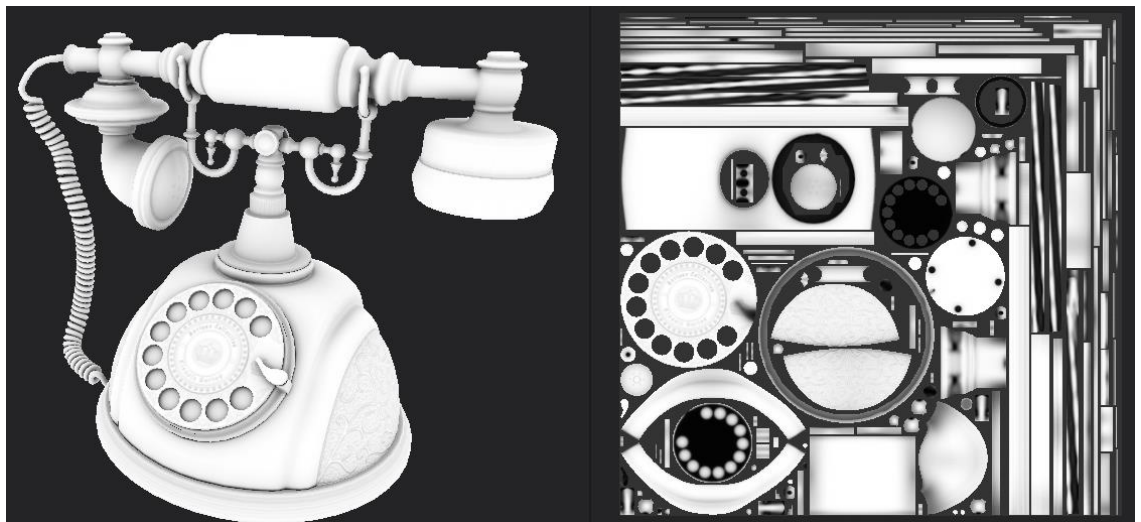


Рисунок 2.4 – Приклад текстури затінення

Відмінною особливістю застосованого методу є те, що створюється окрема текстура затінення об'єкта, яка надалі змішується з основною текстурою (рисунок 2.5). Таким чином, при створенні декількох колірних рішень об'єкта не потрібна відповідна підготовка для кожної текстури.



Рисунок 2.5 – Приклад застосування незалежної текстури затінення

Наприклад, об'єкт має n різних колірних рішень. При необхідності візуалізувати окрему текстуру для кожного колірного рішення ми змогли б отримати n текстур з повною розгорткою для всього об'єкта. Для забезпечення якісної візуалізації, слід було б використовувати текстури розміром 2048x2048 пікселів.

При такому підході будуть необхідні також n текстур кольорів і додатково карта затінення. Але при використанні окремої карти затінення, не буде необхідності дублювати різні ділянки текстури кольору з різним ступенем затіненості. Таким чином текстури кольору можуть мати набагато менший розмір (наприклад, 128x128 пікселів). При цьому немає необхідності використовувати карту затінення високої якості. Експериментально встановлено, що допустимі розміри карти затінення можуть бути розміром 512x512 пікселів і нижче.

Текстура затінення поєднується з основною текстурою в режимі накладення «Множення» (Multiply). Даний режим оптимальний для

накладення напівпрозорих тіней. На відміну від накладення сірих відтінків тіней поверх основної текстури відбувається затінення безпосередньо пікселів основної текстури на основі даних в текстурі затінення. Таким чином, текстурі затінення не потрібен додатково альфа-канал, що економить ресурси, які витрачаються на її обробку. Також чорно-біла текстура затінення дозволяє більш гнучко проводити налаштування інтенсивності тіней в порівнянні з редагуванням текстури з прозорістю.

2.1.2 Реалізація ефекту падаючої тіні об'єктів

Був також розроблений метод створення статичної падаючої тіні, що відкидається на плоску поверхню, на якій розташовуються віртуальні об'єкти.

Також, як і карта затінення, падаюча тінь прораховується заздалегідь у високій якості і зберігається у вигляді текстури. В ході експериментів з'ясувалося, що не потрібна текстура з високою роздільною здатністю, а в основному достатньо і набагато менших розмірів (наприклад, 128x128 пікселів). Далі текстура застосовується до площини, розташованої під об'єктом (рисунок 2.6). Тип шейдера площини використовує значення прозорості текстури і коректно візуалізує напівпрозорі ділянки. З метою оптимізації шейдер не використовує інформацію про джерела освітлення.



Рисунок 2.6 – Приклад реалізації ефекту падаючої тіні

Падаюча напівпрозора тінь вносить істотний внесок в реалістичність вбудовування віртуального об'єкта в реальне оточення. Відмінною особливістю представленого підходу є можливість успішного застосування в задачах реального часу.

2.2 Розробка методу підготовки та зберігання тривимірних моделей об'єктів

Для імплементації запропонованих методів щодо забезпечення реалістичності потрібна була розробка певного процесу попередньої підготовки моделей, а також особливої структури зберігання (рисунок 2.7).

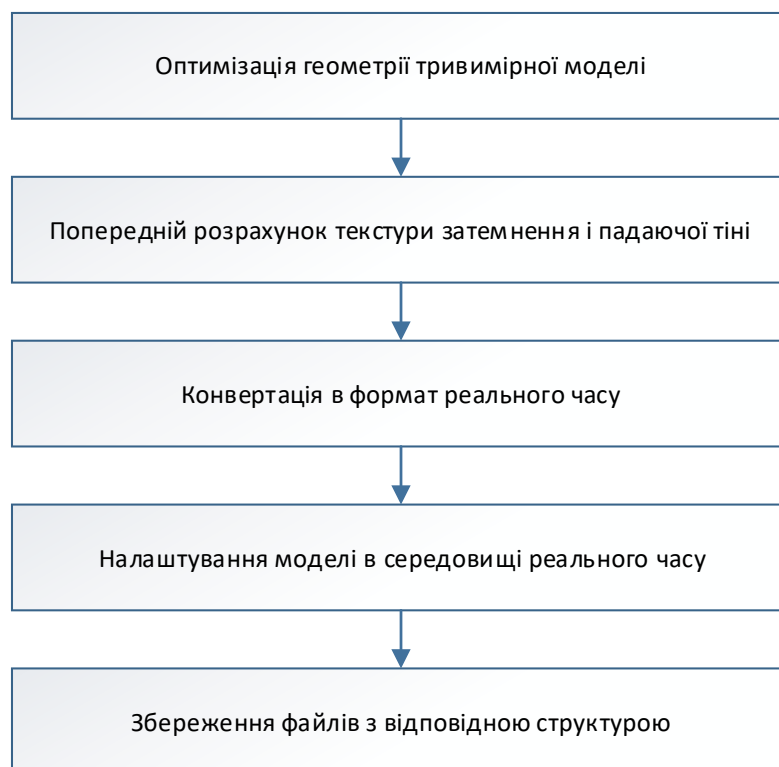


Рисунок 2.7 – Алгоритм підготовки тривимірної моделі

Початкове налаштування тривимірної моделі проводиться в тривимірному редакторі (наприклад, 3ds Max або Maya).

Сучасні масові мобільні пристрої, а також програмне забезпечення досягло вражаючих результатів щодо тривимірної візуалізації. Складна геометрія, яка містить 300 тис. полігонів може успішно оброблятися на мобільних пристроях 2009 року, тим більше сучасні смартфони.

В цілому, можна констатувати, що складність геометрії (кількість полігонів) не є першочерговим параметром для оптимізації в сучасних умовах. Більш ресурсоемними є операції прорахунку складних шейдерів.

На етапі створення текстури затінення і падаючої тіні в тривимірному редакторі виробляється настройка освітлення для найбільш вигідного представлення конкретної моделі. Далі проводиться візуалізація (рендеринг) у високій якості з висновком результату в файли зображень (формати TIFF, PNG, JPEG).

Далі проводиться експорт моделі в формат FBX за допомогою відповідних надбудов для тривимірного редактора. Даний формат використовується для забезпечення сумісності різних програм тривимірної графіки. Модель в форматі FBX імпортується в проєкт Unity, і подальша настройка проводиться в редакторі Unity.

У редакторі Unity проводиться настройка всіх матеріалів моделі. Кінцевий результат візуалізації матеріалу з тими чи іншими настройками, шейдером і текстурами необхідно тестувати на передбачуваних пристроях.

Для роботи з декількома колірними рішеннями (комбінаціями матеріалів) до об'єкта моделі (або до декількох об'єктів частин складовою моделі) додається компонент ColorManager і проводиться його настройка.

Також проводиться регулювання правильної позиції і масштабу щодо маркера, використовуваного в проєкті.

Для коректного розпізнавання торкань на об'єкті проводиться додавання та налаштування компонента Collider. Даний компонент визначає область в тривимірному просторі, в якій торкання розпізнаються на об'єкті. Область, як правило, має спрощену форму (паралелепіпед, куля і т.п.) з метою спрощення обчислень.

Типова структура зберігання об'єкта тривимірної моделі передбачає зберігання інформації про тривимірну геометрію, матеріали і текстури (рисунок 2.8).

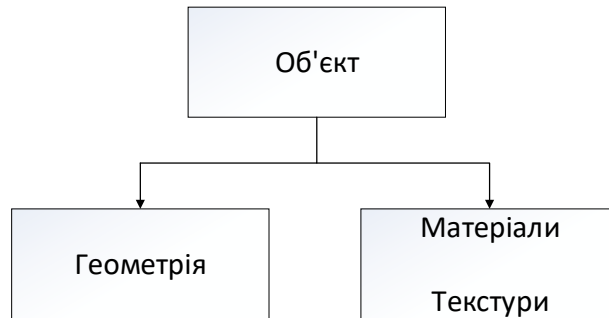


Рисунок 2.8 – Базова структура зберігання об'єкта тривимірної моделі

Об'єкт, підготовлений відповідно до запропонованого методу, матиме структуру, яка передбачає також зберігання інформації про область дотику об'єкта, матеріалів і текстур всіх кольірних рішень, а також попередньо розрахованих текстур статичного освітлення (рисунок 2.9).

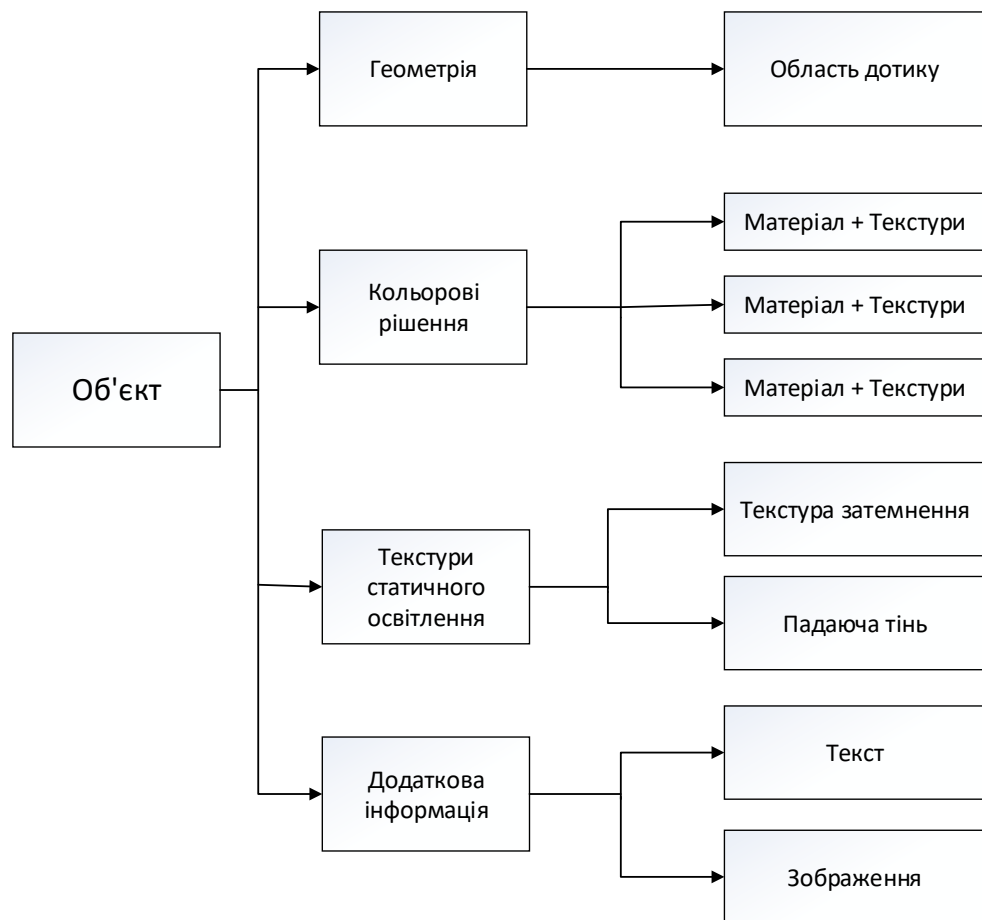


Рисунок 2.9 – Запропонована структура збереження об'єктів

Також може бути передбачено зберігання відповідних зображень моделі і використаних в ній матеріалів для використання при побудові графічного інтерфейсу користувача.

2.3 Алгоритм маніпуляції віртуальними тривимірними об'єктами в просторі з використанням жестових методів введення

Об'єкт в доповненій реальності візуалізується в положенні над зображенням-маркером (рисунок 2.10). При цьому лицьова сторона об'єкта відповідає лицьовій стороні маркера.

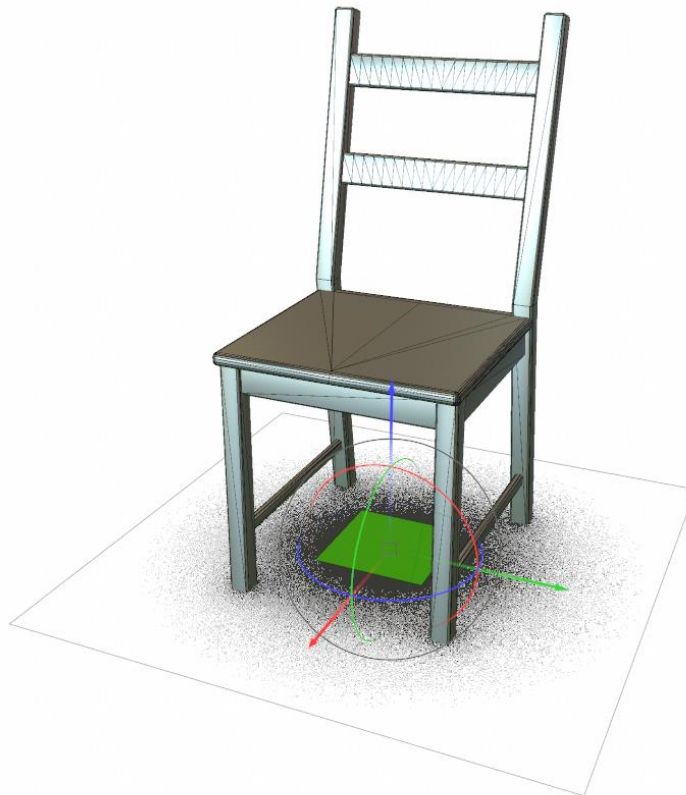


Рисунок 2.10 – Розміщення об'єкта над маркером

Для того, щоб візуалізувати об'єкт в іншому положенні, користувачеві необхідно пересунути зображення-маркер відповідним чином. Це істотно погіршує взаємодію. Наприклад, користувач відповідно до інструкції може покласти маркер на підлогу і відійти від нього на відстань до 2-5 метрів. Може

виявитися, що модель необхідно зрушити на невелику відстань або повернути (напр., Маркер і, відповідно, модель повернені до користувача не необхідною стороною). У цій ситуації користувачу необхідно пройти 2-5 метрів до маркера, поміняти його положення і пройти назад. Самим критичним є те, що користувачеві доведеться відірватися від роботи з графічним інтерфейсом, що підвищить когнітивне навантаження і ще більш збільшить час, необхідний на операцію переміщення об'єкта.

Щоб вирішити цю проблему був розроблений інтерфейс маніпуляції об'єктів в просторі.

Іншою перевагою імплементації функції переміщення об'єктів є можливість збільшення дистанції до об'єкта, компенсуючи недостатню дистанцію розпізнавання маркера[13].

Координати точок об'єкта при повороті, переміщенні і зміні масштабу об'єкта в загальному випадку змінюються у відповідності з наступною формулою:

$$\begin{bmatrix} X_1 \\ Y_1 \\ Z_1 \end{bmatrix} = M \cdot \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \cdot \begin{bmatrix} X_0 \\ Y_0 \\ Z_0 \end{bmatrix} + \begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix}, \quad (2.5)$$

де

X_0, Y_0, Z_0 – початкові координати точки об'єкта,

X_1, Y_1, Z_1 – кінцеві координати точки об'єкта,

M – коефіцієнт зміни масштаба,

матриця $\|a_{mn}\|$ – матриця перетворення повороту, елементи якої виражені через кути повороту довкола трьох просторових осей,

$\vec{r} = (r_1, r_2, r_3)$ – вектор переміщення об'єкта.

В завданні розстановки віртуальної меблів, користувачеві не потрібен повноцінний інтерфейс маніпуляції тривимірними об'єктами з шістьма ступенями свободи. Завдання переміщення предметів меблів є двомірним. І таким чином в даному випадку інтерфейс може бути спрощений. Переміщення

предметів меблів здійснюється в межах площини підлоги приміщення, по осях X і Z. Обертання здійснюється навколо своєї осі Y (рисунок 2.11).

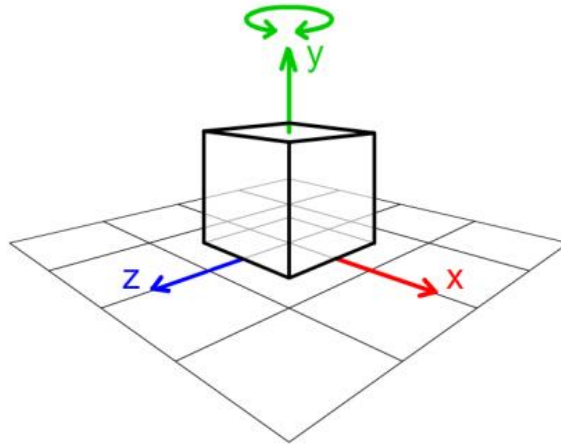


Рисунок 2.11 – Осі переміщення об'єкта (X, Z) та вісь повороту (Y)

У разі переміщення і обертання об'єкта в площині XZ без зміни масштабу наведена вище формула спрощується: координата Y залишається незмінною, а координати X і Z змінюються відповідно до виразу

$$\begin{bmatrix} Z_1 \\ X_1 \end{bmatrix} = \begin{bmatrix} \cos \varphi & -\sin \varphi \\ \sin \varphi & \cos \varphi \end{bmatrix} \cdot \begin{bmatrix} Z_0 \\ X_0 \end{bmatrix} + \begin{bmatrix} r_3 \\ r_1 \end{bmatrix}, \quad (2.6)$$

де φ – кут повороту об'єкта довкола осі Y.

Для реалізації інтерфейсу маніпуляції об'єктами були розглянуті варіанти використання кнопок і різних інших екранних елементів управління. Однак набагато більш ефективним рішенням виявилось використання жестових методів введення.

Сенсорні екрани масових мобільних пристроїв в більшості оснащені функцією розпізнавання декількох торкань одночасно (так звана функція «мультитач»). Таким чином була можливість призначити один жест (Просте торкання і переміщення одного пальця) для переміщення об'єкта (трансляцію координат), а інший (торкання одночасно двома пальцями) - для обертання.

Був розроблений алгоритм, що обробляє введення з сенсорного екрану (рисунок 2.12). При виявленні торкання екрану проводиться перевірка, чи відбулося потрапляння в область віртуального об'єкта.

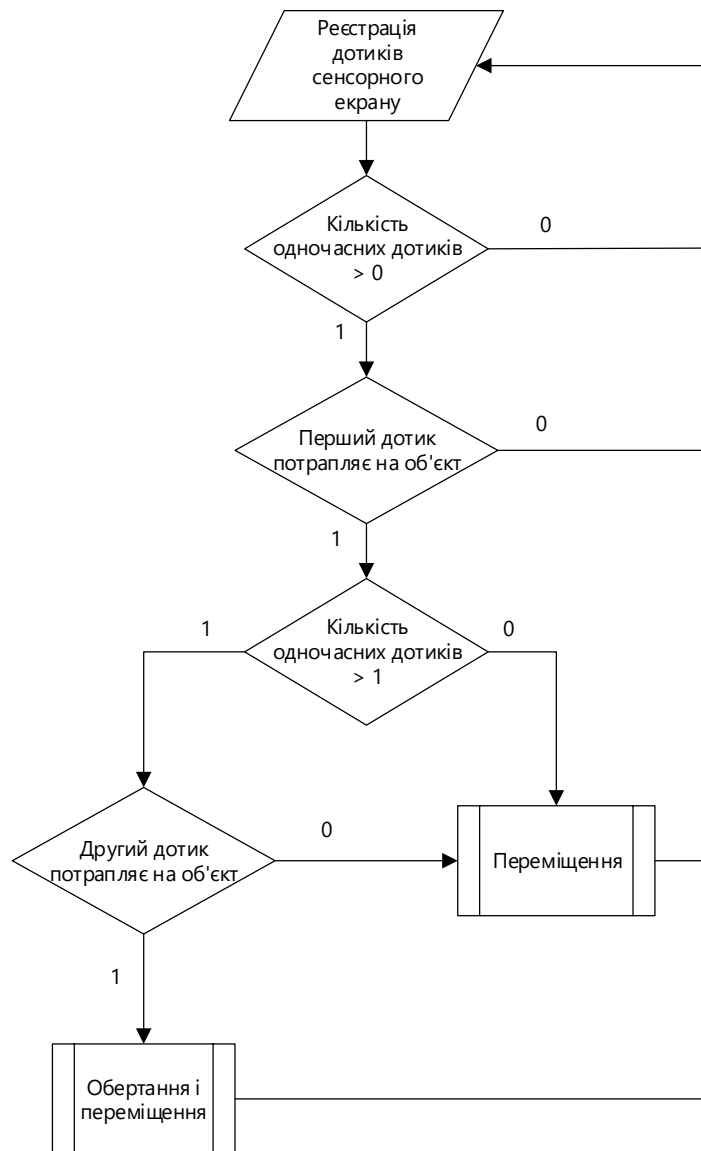


Рисунок 2.12 – Блок-схема алгоритма маніпуляції об'єктами

Віртуальний промінь прокладається від точки дотику користувачем сенсорного екрану, що знаходиться на площині екрану, знайденої за допомогою віртуальної камери Unity. У разі перетину променем об'єкта, повертається позитивний результат. При наявності на сцені декількох віртуальних об'єктів, необхідно визначити, з якими з них виробляється взаємодія.

При виявленні більш ніж одного дотику на одному об'єкті розпізнається режим обертання. Кут, на який потрібно повернути об'єкт, визначається шляхом вимірювання кута обертання прямої, проведеної через дві точки

дотику, порівняно з попереднім кадром. При виявленні більше двох торкань на об'єкті, обробляються тільки два перших.

Важливо зауважити, що в запропонованій реалізації при режимі обертання режим переміщення залишається активним. Тобто можливо, обертаючи об'єкт дотиком двох пальців, одночасно переміщати його, переміщаючи пальці відповідно. Відстежується трансляція середньої точки між двома дотиками. Неформальні випробування показали, що більшість користувачів позитивно оцінили, вважаючи за краще використовувати дотик двома пальцями для переміщення і обертання одночасно, і рідко користуються режимом переміщення одним дотиком в чистому вигляді.

Переваги розробленого механізму маніпуляції:

1. Взаємодія при візуалізації об'єктів засобами доповненої реальності покращено функцією переміщення і обертання.
2. Імплементація функції маніпуляції об'єктами може частково компенсувати недоліки технологій трекінгу і збільшити дистанцію до візуалізуються об'єктів.
3. У порівнянні з використанням елементів графічного інтерфейсу (кнопок або повзунків) запропонований механізм є більш природним для користувача.

2.4 Розробка інтерфейсу програмного додатку

При візуалізації засобами доповненої реальності графічний інтерфейс повинен забезпечувати доступ користувача до наступних основних завдань:

- перегляд інформації за доступними для вибору об'єктів для візуалізації та можливість вибору відповідного об'єкта;
- перегляд інформації про різні варіанти матеріалів поточного об'єкта і можливість вибору відповідного матеріалу.

Для масового застосування важливими є питання зрозумілості та зручності роботи для широкого кола користувачів. Внаслідок цього, в

дослідженні завжди робився наголос на даний момент, і всі питання вирішувалися з максимальною орієнтацією на користувача.

Перегляд доступних для візуалізації об'єктів пропонується організувати у вигляді двомірного меню. Пропонується кілька варіантів відображення даного меню, які можуть бути зведені до двох основних. У першому варіанті меню вибору об'єкта відображається поверх відеопотоку з камери, відеопотік при цьому не зупиняється. Таким чином користувачеві не потрібно виходити з режиму доповненої реальності. Другий варіант передбачає окремий режим меню, що займає весь простір екрану – для користувача це виглядає як вихід з режиму доповненої реальності. При цьому логічним видається рішення про припинення відеопотоку і зупинці циклу трекінгу.

До плюсів першого варіанта можна віднести більш високу швидкість виклику меню, так як при варіанті з окремим режимом витрачається час на зупинку і подальше відновлення роботи відеопотоку і трекінгу. Проте головним мінусом рішення є те, що відеопотік буде видно користувачеві і тим самим може негативно впливати на швидкість сприйняття інформації в меню. Також можна відзначити більш високе обчислювальне навантаження. Для забезпечення коректної роботи на більш слабких пристроях в меню, що відображається поверх відеопотоку, не буде рекомендовано застосування складних в обчисленні елементів (анімацій, прокручувань великих обсягів даних і ін.).

При варіанті з окремим режимом існує більше можливостей для оформлення інформації. Всі обчислювальні ресурси пристрою можуть бути спрямовані безпосередньо на роботу з меню. Вся увага користувача також буде зосереджено тільки на списку об'єктів. В даному варіанті може бути більш успішно використана складна різноманітна структура меню, а також реалізовані різноманітні фільтри по параметрам при наявності великої кількості об'єктів.

Проаналізувавши плюси і мінуси описаних варіантів було вирішено розробити меню для зміни об'єкта в режимі доповненої реальності з використанням окремого режиму, що дасть можливість не збільшувати навантаження на пристрої та розробити зручне і зрозуміле меню.

При наявності у об'єкта декількох можливих варіантів виготовлення (забарвлення, матеріалів), виникає необхідність реалізації інтерфейсу на вибір користувачем відповідного варіанту. Такий інтерфейс необхідний і при наявності у об'єкта декількох станів (наприклад, світильник включений або виключений).

Найбільш ефективним є реалізація засобів зміни параметрів об'єктів безпосередньо в режимі доповненої реальності з миттєвим відображенням результату. Причому необхідно, щоб засоби інтерфейсу зі зміни параметрів об'єкта були більш доступними, ніж в професійному програмному забезпеченні.

У прагненні до максимально простого і зручного способу зміни матеріалу об'єкта для користувача, було запропоновано наступне. У режимі доповненої реальності на екрані справа по центру відображати кнопку виклику меню зміни матеріалу (рисунок 2.13).

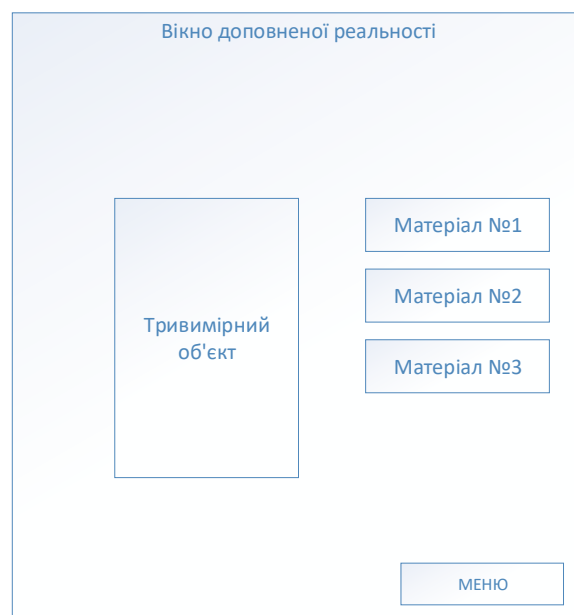


Рисунок 2.13 – Схема інтерфейсу виклику меню вибору матеріалів

Меню зміни матеріалу представляє собою список з доступних для вибору матеріалів, кожен пункт якого реалізований у вигляді кнопки. На кнопці пропонується розташовувати зображення фрагмента матеріалу і напис з назвою.

Розміри кнопок по вертикалі повинні бути зручні для роботи великим пальцем руки. Натискання в будь-якому місці на екрані поза меню закриває його. Розглядався варіант прив'язки позиції меню зміни матеріалів об'єкта безпосередньо до позиції тривимірної моделі, що було б більш логічним. Однак при такому варіанті у користувачів виникали складності. По-перше, при триманні перед собою планшетного комп'ютера виникає мимовільне тремтіння рук і текст пунктів меню виявляється мають важко навіть при реалізації алгоритмів згладжування позиції меню. По-друге, при втраті трекінгу і приховуванні об'єкта, меню також необхідно приховувати, і це також може викликати незручності у користувача.

2.5 Висновки

У другому розділі магістерської кваліфікаційної роботи було розроблено методи підвищення реалістичності при візуалізації тривимірних об'єктів засобами доповненої реальності, що відрізняються використанням комбінації статичного і динамічного освітлення з попередньою візуалізацією текстури затінення і статичної падаючої тіні об'єктів. Розроблено метод підготовки і зберігання тривимірних моделей об'єктів для їх реалістичної візуалізації, що відрізняється структурою і форматом зберігання даних. Запропоновано алгоритм маніпуляції віртуальними тривимірними об'єктами в просторі з використанням жестових методів введення, що відрізняються можливістю одночасного переміщення та обертання, а також швидкістю і стабільністю роботи. Проаналізовано проектування графічного інтерфейсу користувача для візуалізації засобами доповненої реальності, що включають побудову меню вибору об'єктів, розташування елементів управління на екрані.

3 РОЗРОБКА ПРОГРАМНОГО ЗАСОБУ ДЛЯ ІНТЕРАКТИВНОЇ ВІЗУАЛІЗАЦІЇ ТРИВИМІРНИХ ОБ'ЄКТІВ У РЕАЛЬНОМУ СЕРЕДОВИЩІ

3.1 Варіантний аналіз і обґрунтування вибору засобів для реалізації програмного засобу

Візуалізація є одним з найбільш опрацьованих аспектів в реалізації технології доповненої реальності. Для реалізації частини технології доповненої реальності, пов'язаної з тривимірною візуалізацією може бути застосовано безліч існуючих рішень.

Однак, враховуючи необхідність візуалізації в режимі реального часу, вибір середовища відбувається між ігровими рушіями.

Ігровий рушій – це середовище розробки програмного забезпечення, призначена для людей, що створюють відеоігри. Розробники використовують ігрові рушії для створення ігор для консолей, мобільних пристроїв і персональних комп'ютерів. Основні функціональні можливості, що зазвичай надаються ігровим рушієм, включають в себе механізм рендеринга («рендерер» або генерація / обробка графічного зображення) для 2D або 3D-графіки, фізичний движок або виявлення зіткнень (і реагування на зіткнення), звук, сценарії, анімацію, штучний інтелект, мережеве взаємодія, потокову передачу, пам'ять управління, багатопоточність, підтримка локалізації, графічні сцени і може включати в себе підтримку відео для кінематографії.

У багатьох випадках ігрові рушії надають набір інструментів візуальної розробки на додаток до повторно використовуваних програмним компонентам. Ці інструменти можуть надаватися в інтегрованому середовищі розробки, що дозволяє спростити і прискорити розробку ігор на основі даних. Більшість ігрових движків надають засоби, які полегшують розробку, такі як графіка, звук, фізика та функції штучного інтелекту. Ці ігрові рушії іноді називають «проміжним програмним забезпеченням», тому що, як і в діловому

сенсі цього слова, вони надають гнучку і багаторазово використовувану програмну платформу, яка надає всі основні функціональні можливості, необхідні для розробки ігрової програми, при одночасному зниженні витрат.

Сучасні ігрові або графічні рушії зазвичай надають графічні сцени, які є об'єктно-орієнтованим поданням ігрового світу 3D, який часто спрощує ігровий дизайн і може використовуватися для більш ефективного рендеринга великих віртуальних світів.

Unreal Engine 4 це четвертий за рахунком ігровий рушій компанії Epic Games[22]. Перша гра була розроблена на рушію Unreal Engine 1 ще в далекому 1998 році. На сьогоднішній день було розроблено понад 100 успішних проектів з використанням різноманітних версій рушія.

На сьогоднішній день ігровий движок поширюється безкоштовно, однак якщо прибуток від гри перевищує 3000 \$ за квартал, розробник змушений віддавати 5% прибутку.

Рушій є кросплатформовим, на ньому можна вести розробку під: PC, Mac, Xbox, Xbox360, Xbox One, PlayStation 2-4, Wii & Wii U, iOS, Android.

Крім того, рушій відмінно адаптований під мережеві ігри (підтримуються технології Xbox Live, Windows Live, GameSpy і інші) і підтримує підключення до 64 гравців в сесії.

Даний ігровий рушій відмінно підходить для проектів будь-якого розміру: від невеликого аматорського додатки, до величезного комерційного проекту з десятками розробників.

Unreal Engine 4 має високоякісної графікою, при використанні шейдерів дозволяє рендерити фотореалістичну графіку в режимі реального часу. Крім того, відмінно реалізований процес створення анімації персонажів. В останніх версіях рушія активно інтегрується система віртуальної реальності VR.

Система створення матеріалів дозволяє використовувати шейдинг на основі фізики об'єкта, який дає контроль над зовнішнім виглядом персонажів і предметів. Швидко створення широкого спектра поверхонь, які навіть при уважному розгляді виглядають реалістично.

Unreal Engine 4 підтримує повний доступ до вихідного коду C ++ через GitHub. Незалежно від розміру команди, ви можете легко координувати розвиток і дизайнерські зусилля з іншими версіями інфраструктури UE4.

Unreal Engine 4 інтегрований з продуктами більш ніж десяти великих виробників проміжного ПЗ як NVIDIA PhysX, Autodesk Gameware, Enlighten, Umbra, Oculus VR і інші.

Також движок має докладний технічний опис, а розробник постійно викладає нові пояснення і опису поточних і нововведених функцій, що істотно спрощує процес освоєння програмного забезпечення.

CryEngine – ігровий рушій, розроблений німецькою компанією Crytek і є наступником движка CryEngine 2. CryEngine 3 спочатку є кросплатформним рушієм – він з самого початку орієнтований на IBM PC-сумісні комп'ютери та ігрові консолі Microsoft Xbox 360 і Sony PlayStation 3, а також на їх наступні версії. Крім того, рушій орієнтований на розробку масових багатокористувацьких онлайн-ігор (англ. Massively Multiplayer Online Game – MMOG).

Глобальне освітлення на основі вокселей (SVOGI) дає можливість створювати сцени, схожі на життя, з фотореалістичним освітленням. Нещодавно розроблена система Voxel Global Illumination розширює існуюче освітлення на основі зображень (IBL) з великомасштабною оклюзією навколишнього простору і відображенням непрямого світла в реальному часі, з використанням системи освітлення CRYENGINE на абсолютно новий, фотореалістичний рівень. У порівнянні з традиційними попередньо дослідженими рішеннями, Voxel-based Global Illumination не вимагає яких-небудь довгих попередніх обчислень або дискового сховища, що дозволяє використовувати його для величезних відкритих світових середовищ з динамічним часом доби, а також оптимізувати рівень проектування для розробників, дозволяючи їм негайно протестувати освітлення в режимі реального часу.

За допомогою функції тіні для кожного об'єкта карти тіней можуть бути призначені будь-якому зазначеному об'єкту на рівні. Це може привести до збільшення якості тіні через більш високої щільності тіньового тесселя в світі і зменшеного діапазону глибини. Це особливо корисно для роликів або об'єктів, які потрібно включити в визначення додаткової тіні. Ця функція доступна через виділений вузол Flowgraph, який дає дизайнерам легкий доступ до кращих динамічних тіней в реальному часі.

Crytek представила Screen Space Ambient Occlusion в іграх реального часу і тепер покращує її за допомогою нової техніки Screen Directional Occlusion Occlusion (SSDO). SSDO дозволяє включати будь-яку кількість вогнів в розрахунок Ambient Occlusion і колір всіх вогнів, які необхідно враховувати. SSDO покращує точність фізичного освітлення і, в кінцевому підсумку, дозволяє скидати контактні тіні з кожного джерела світла в навколишньому середовищі.

Освітлення на основі зображень (IBL) забезпечує узгодженість освітлення і рендеринга з усіма матеріалами. Модель затінення CRYENGINE використовує розширені значення функції розподілу двонаправленого відображення і нормалізацію дзеркальних відблисків для створення дуже складних ситуацій освітлення. На відміну від більш традиційної установки, IBL використовує дзеркальне відображення і шорсткість матеріалів, щоб повернути більш фізично правильний зовнішній вигляд. Жорсткі дзеркальні відображення стають такими ж простими, як перегляд однієї текстури, в той час як розмиті відображення можуть бути ефективно апроксимовані шляхом коригування рівня міртар під час пошуку.

Unity – це кросплатформовий ігровий рушій, розроблений Unity Technologies, вперше анонсований і випущений в червні 2005 року на Всесвітній конференції розробників Apple Inc. в якості ексклюзивного ігрового рушія для OS X[23]. На 2018 року движок був розширений для підтримки 27 платформ.

Unity дає користувачам можливість створювати гри як в 2D, так і в 3D, движок пропонує основний API для написання скриптів на C # як для редактора Unity у вигляді плагінів, так і для самих ігор.

Unity підтримує C# в якості скриптової мови. Розрахунки фізики виробляє фізичний рушій PhysX від NVIDIA. У редакторі Unity є можливість створювати власні шейдери під конкретні вимоги.

Unity має візуальне середовища розробки та міжплатформенну підтримку. Крім цього, до переваг відноситься модульна система компонентів, за допомогою якої відбувається конструювання ігрових об'єктів.

До недоліків Unity відносять обмеження візуального редактора при роботі з багатокomпонентними схемами. Другим недоліком є відсутність підтримки в Unity посилань на зовнішні бібліотеки, роботу з якими програмістам доводиться налаштовувати самостійно; це також ускладнює командну роботу.

Так як програмне забезпечення Unity почало набувати свою популярність відносно недавно, команда розробників активно покращує свій продукт. Відбувається постійне поліпшення існуючого функціоналу, позбавлення від недоліків і підкреслення переваг. У зв'язку з цим постійним зміною функціоналу, розробниками підтримується актуальна версія документації, розташована на їх офіційному сайті. Також присутні відеоуроки англійською мовою які дозволяють навчитися створювати примітивні додатки навіть повністю незнайомому з Unity програмісту при наявності відповідних знань мови C # або Java.

Інструменти розробника Unity 5 підтримують мови C # і Java для реалізації функціоналу програм, а також різні види шейдерних мов з надбудовою Shader Lab. У своєму проекті я буду використовувати мову C#, активно підтримуваний Microsoft в їх серії продуктів Visual Studio.

В результаті аналізу в якості основи для програмної реалізації системи була обрана середовище розробки Unity. Це високорівневе середовище розробки для додатків та ігор. Включає потужний редактор з графічним

інтерфейсом і графічний движок, оптимізований для роботи на мобільних пристроях. Unity є кросплатформним засобом розробки і дозволяє компілювати проект з одним вихідним кодом і ресурсами під провідні мобільні платформи, такі як iOS, Android, Windows Phone.

Unity використовує формат тривимірних моделей FBX. Більшість сучасних засобів створення тривимірних моделей, таких як 3ds Max, Maya, підтримують експорт в цей формат. У більшості випадків моделі переносяться коректно.

Моделі, текстури, матеріали, скрипти можна запаковувати в формат .unityassets і передавати іншим розробникам, або викладати у вільний доступ.

Призначені для користувача скрипти можуть бути написані на будь-якому .NET мовою програмування і скомпільовані в звичайні керовані збірки (DLL). Є вбудовані компілятори для C #, UnityScript і Boo. При цьому є можливість організації інтерфейсу з зовнішнім кодом.

Окрім власне ігрового рушія, засобами якого виконується розробка додатку та візуалізація тривимірних об'єктів в режимі реального часу, необхідно обрати спеціалізований засіб розробки для роботи з доповненою реальністю. Було розглянуто декілька поширених технологій для роботи з доповненою реальністю.

ARToolKit – бібліотека програмного забезпечення для побудови додатків доповненої реальності. Це додатки, які передбачають накладення віртуальних зображень на реальний світ. Наприклад, на зображенні праворуч з'являється тривимірний віртуальний символ, що стоїть на реальній карті. Користувач може це бачити на дисплеї мобільного пристрою. Коли користувач переміщає карту, віртуальний персонаж рухається з нею і виглядає приєднаним до реального об'єкта.

ARToolKit використовує алгоритми комп'ютерного зору. Бібліотеки відео відстеження ARToolKit обчислюють реальне положення та орієнтацію камери щодо фізичних маркерів у режимі реального часу. Це дає можливість

легко розробити широкий спектр додатків доповненої реальності. Деякі функції ARToolKit включають:

- Код відстеження, який використовує прості чорні квадрати.
- Можливість використовувати будь-які шаблони квадратних маркерів.
- Простий код калібрування камери.
- Висока швидкодія для AR-програм у реальному часі.
- Поширення на платформах SGI IRIX, Linux, MacOS та Windows OS.
- Поширюється з повним вихідним кодом.

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 відслідковувати плоскі зображення і 3D-моделі в реальному світі, коли вони розглядаються через камеру мобільного пристрою.

Vuforia виявляє та відстежує плоскі зображення (маркери), а також прості тривимірні об'єкти (коробки, циліндри і т.д.) в режимі реального часу. Vuforia має свій API, що дозволяє писати код на мовах C++, Java, Objective-C++ і .Net. VuforiaSDK можна встановити в якості розширення для таких IDE як Android Studio, XCode, Visual Studio і вести розробку програми під конкретну платформу (наприклад, WindowsPhone, Android чи iOS), або використовувати розширення Vuforia для ігрового рушія Unity і розробляти кроссплатформний додаток. Додатки, створені за допомогою Vuforia, підтримуються великою кількістю мобільних пристроїв, включаючи iPhone, iPad, а також смартфони та планшети з ОС Android версії 2.2 і вище.

Metaio SDK – готова бібліотека для створення мобільних додатків доповненої реальності. Використовує OpenGL використовує SLAM методи для більш точної роботи. Дозволяє розробникам використовувати контент без попереднього шифрування для реалізації доповненої реальності в будь-яких областях застосування, що пропонує модулі для трекінгу тривимірних

об'єктів, розпізнавання і відстеження осіб, інфрачервоного і лазерного стеження, розширеної калібрування камери і прозорою оптики. Для розгортання тривимірних моделей і відстеження даних не потрібні офлайнові інструменти або шифрування на сервері. В даний час SDK підтримує висновок на платформи Windows, OS X, iOS і Android.

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

Результати аналізу занесено у таблицю 3.1.

Таблиця 3.1 – Порівняльний аналіз засобів роботи з доповненою реальністю

Критерій	ARToolKit	Vuforia	Metaio SDK	String
Кроссплатформенність	iOS, Android, Unity	iOS, Android, Unity	iOS, Android, Unity	iOS, Unity
Повнота та зрозумілість документації (оцінка від 0 до 5)	4	5	3	2
Підтримка різних мов програмування	Java, Objective-C	C++, Java, Objective-C, .Net	Java, Objective-C, .Net	Objective-C, .Net
Підтримка додаткових сервісів (оцінка від 0 до 5)	2	5	3	1
Тип ліцензування	Free + Commercial SDK option	Free + Commercial SDK option	Free + Commercial SDK option	Commercial SDK option

Отже, після аналізу існуючих засобів розробки додатку, таких як ігрові рушії та програмні засоби для роботи з доповненою реальністю, було вирішено розробку проводити в середовищі Unity з використанням спеціалізованого засобу Vuforia, що максимально відповідає поставленим задачам.

3.2 Реалізація процесу трекінгу

Станом на сьогоднішній день розробки в області трекінгу досягнуто того рівня, коли є можливість дуже точного відслідковування позицій і руху людини, однак, як правило, це досягається використанням багатьох спеціальних засобів, наприклад, трекери руху, спеціальні відбивачі або конкретна кількість окремо розташованих камер.

Існує декілька можливих підходів для реалізації трекінгу на масових мобільних пристроях.

При одному з практичних підходів функціонал трекінгу не реалізується. У даному випадку складається певна колекція із зображених об'єктів. При цьому фоном може бути представлений як відеопотік з камери, так і статичне зображення.

До плюсів можна віднести те, що від користувача не вимагається особливих попередніх дій для роботи з ПЗ.

Однак вихідне зображення виходить низько реалістичним в наслідок невідповідності ракурсу об'єкта з реальним оточенням. У користувача виникають великі труднощі підстроювання масштабу у відповідність з реальним. Як наслідок, буде відсутнє правильне сприйняття габаритів об'єкта. Також так як найчастіше використовується звичайне двомірне зображення, немає можливості оглянути об'єкт з усіх боків під різними ракурсами. В наслідок цього реалізовані таким чином програми не задовольняють одну з основних принципів доповненої реальності: роботі в тривимірному просторі. Дані аспекти зменшують споживчу цінність такого підходу.

Другим підходом є імплементація трекінгу на основі датчиків прискорення і положення пристрою. Такий спосіб дозволяє реалізувати трекінг з трьома ступенями свободи: нахил пристрою по трьох осях. Однак в цьому випадку відсутні дані про зміщення пристрою в просторі. Це означає, що відстань від камери до об'єкта залишається невідомим. На практиці це спричиняє відображення об'єкта в некоректному масштабі, що є серйозним недоліком. При переміщенні користувача, об'єкт також переміщається, як би прикріплений до камери. Але навіть якщо користувач не змінює позицію, присутня неточність в ракурсі об'єкта, пов'язана з висотою, на якій знаходиться пристрій: вона визначається або приблизно, виходячи з людського зросту, або статично задається вручну.

Третім реалізованим на масових мобільних пристроях підходом є оптичний трекінг. Він є найбільш функціональним в порівнянні з попередніми варіантами, так як надає трекінг з шістьма ступенями свободи, що дозволяє реалізувати повноцінну доповнену реальність і найбільш точно вписати віртуальний об'єкт в реальне оточення з дотриманням коректного масштабу і ракурсу. Для задовільного функціонування оптичного трекінгу достатньо однієї камери, якою оснащується більшість сучасних масових мобільних пристроїв. А алгоритми оптичного трекінгу, заснованого на маркерах, добре оптимізовані для роботи на мобільних пристроях, дозволяють домогтися інтерактивної частоти зміни кадрів. Головним мінусом є те, що в більшості реалізацій для роботи необхідний додатковий зовнішній маркер для відстеження.

Кожен з викладених методів має певні переваги і недоліки. Однак був застосований оптичний трекінг на основі маркерів, як найбільш функціональний.

Основну роль в розвитку технології доповненої реальності грає комп'ютерний зір – наукова дисципліна, що вивчає теорію і базові алгоритми аналізу зображень і сцен. Алгоритми комп'ютерного зору дозволяють виділяти ключові особливості на зображенні (кути, межі області), проводити пошук

фігур і об'єктів в реальному часі, виконувати 3D реконструкцію з кількох фотографій і багато іншого[15].

В області доповненої реальності алгоритми комп'ютерного зору використовуються для пошуку в відеопотоці спеціальних маркерів. Пошук заданого об'єкта на відео по суті зводиться до пошуку на статичному зображенні, так як відеофайл є послідовністю кадрів (зображень), лише з тією різницею, що він здійснюється багаторазово для деякої кількості кадрів.

Залежно від завдання в якості маркера можуть виступати як спеціально сформовані зображення, так і, наприклад, особи людей. Після знаходження маркера в відеопотоці і обчислення його розташування, з'являється можливість побудови матриці проєкції і позиціонування віртуальних моделей. З їх допомогою можна накласти віртуальний об'єкт на відеопотік з дотриманням ракурсу і перспективи. Основна складність полягає саме в тому, щоб знайти маркер, визначити його місце розташування в кадрі і спроектувати відповідним чином віртуальну модель.

На сьогоднішній день була створена велика теоретична база в сфері обробки зображень та пошуку на ньому різних об'єктів. Перш за все, це стосується методів контурного аналізу, зіставлення з шаблоном (template matching), визначення характерних рис (feature detection) і генетичних алгоритмів. З точки зору побудови доповненої реальності найчастіше використовуються останні два підходи.

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

Для їх знаходження і 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 побудова відповідностей між двома наборами точок.

За допомогою детектора проводиться пошук ключових точок шаблонного 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. Для встановлення відповідності між ключовими точками і дескрипторами застосовується матчер.

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 можна отримати помітні відставання у знімаються відеоданих.

Для підвищення швидкості роботи алгоритмів визначення ключових точок застосовуються різні способи фільтрації точок з метою мінімізації їх числа і відсіювання поєднань низької якості. Таким чином, можна домогтися не тільки підвищення швидкості роботи алгоритмів, але і якості трекінгу маркерів.

В якості рушія для трекінгу був обраний набір засобів розробки Qualcomm Vuforia. Використання Vuforia помітно скорочує час розробки. При цьому авторами постійно вносяться виправлення і поліпшення. Дане рішення зарекомендувало себе як надійне і ефективне.

У зв'язку з тим, що метою програми є робота з об'єктами відносно великих розмірів, такими як предмети меблів, масштаб роботи програми повинен бути збільшений. У багатьох ігрових та розважальних програмах мова

йде про масштаб столу. У даній роботі слід говорити про збільшення масштабу як мінімум до середніх розмірів кімнати.

Для реалізації цих цілей необхідно забезпечити коректне функціонування трекінгу на великій дистанції.

Розробники програмного забезпечення для трекінгу Vuforia заявляють, що співвідношення максимальної дистанції для розпізнавання і коректної роботи маркера до його розміру повинно бути приблизно 10 до 1. Відповідно, для визначення необхідного розміру маркера можливо використовувати формулу: $s = 10c$, де s - дистанція, c - діагональ маркера (рисунок 3.1).

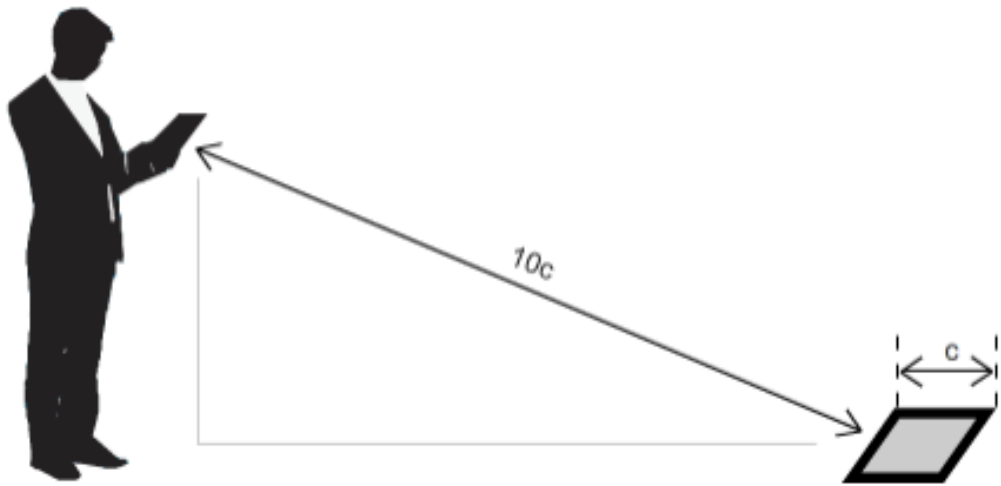


Рисунок 3.1 – Співвідношення розміру маркера та дистанції розпізнавання

У багатьох дослідженнях застосовуються підходи, при яких перед коректним функціонуванням трекінгу необхідний підготовчий етап. Зазвичай він полягає в тому, щоб заздалегідь ввести будь-яку інформацію про навколишнє оточення (приміщення) в систему. Це можуть бути тривимірні моделі приміщення або окремих об'єктів, інформація про висвітлення або інші дані. І хоча деякі розробники застосовують дані підходи в певних випадках, досвід показує, що для споживачів необхідний мінімум початкових налаштувань, особливо коли вони не знайомі з технологією і не знають, які переваги вона несе.

Таким чином, варіанти із застосуванням декількох маркерів, прикріплених спеціальним чином до підлоги та стін, були відкинуті. Безсумнівною перевагою при використанні даних підходів є збільшення кількості ракурсів, при яких в поле зору камери потрапляють маркери. Однак було укладено, що для користувача процес виготовлення і прикріплення до стін декількох маркерів буде занадто трудомісткий, що може значно вплинути на конверсію додатки.

Щоб мінімізувати дії користувача, необхідні для початку роботи з додатком, було вибрано рішення, при якому використовується один маркер, розташований горизонтально на підлозі.

Був обраний стандартний розмір маркера, відповідний формату А4 для зручності друку на простому принтері (чорно-білому або кольоровому).

При діагоналі маркера 364 мм, приблизна відстань розпізнавання дорівнює 3640 мм, що достатньо для масштабів середньої кімнати. Однак в наслідок складних умов освітлення і великого кута нахилу маркера щодо камери, відстань може зменшуватися.

Крім простоти виготовлення маркера, до переваг даного рішення відноситься мобільність. Тобто користувач має можливість оперативно змінити місце для роботи, наприклад, перейти інше приміщення.

При обраному методі трекінгу, взаємодія буде виглядати наступним чином:

- 1) користувач кладе зображення-маркер на горизонтальну поверхню підлоги;
- 2) користувач наводить камеру пристрою на маркер;
- 3) при успішному розпізнаванні маркера на планшеті відображається навколишнє оточення з впровадженим віртуальним об'єктом.

Обраний компонент для трекінгу надає можливість роботи з маркерами двох типів, які підходять під сформульовані вимоги: фрейм-маркери і маркери на основі трекінгу особливостей зображення (feature tracking)[19].

Фрейм-маркери (framemarkers) представляють собою квадратне зображення обрамлене чорно-білою рамкою з закодованим числом (від 0 до 511) (рисунок 3.2).



Рисунок 3.2 – Фрейм-маркер

У середині рамки може бути розміщено будь-яке зображення. Перевагами даного типу маркерів є висока швидкість і дистанція розпізнавання. Однак основним недоліком є те, що необхідно потрапляння маркера в поле видимості камери в повному обсязі.

3.3 Розробка процесу зміни параметрів об'єкта при візуалізації в реальному часі

Об'єкти візуалізації можуть мати різні модифікації: різні варіанти кольору, матеріалів виконання, стану, варіанти компоновки і т.д. Пропонується передбачити можливість перегляду модифікацій і зміни таких параметрів в реальному часі при візуалізації засобами доповненої реальності. Розроблений підхід забезпечує спрощення взаємодії користувача з системою.

Суть підходу полягає у введенні узагальненого параметра, який представляє собою набір змін в об'єкті.

Далі буде розглянуто приклад організації логічної і алгоритмічної реалізації зміни матеріалів об'єктів.

Як вже було викладено вище, в якості одного з орієнтирів для застосування в ході розробки була обрана область візуалізації предметів меблів. В меблевій індустрії існує специфіка, яка полягає в тому, що до однієї і тієї ж моделі меблів виробником може бути запропоновано деяка кількість варіантів використовуваних матеріалів: різні сорти деревини, металу, види і фактури тканини оббивки. При цьому в роздрібному магазині внаслідок обмеженої площі немає можливості продемонструвати всі можливі варіанти. Навіть в друкованому або електронному каталозі найчастіше не надаються зображення моделі у всіх можливих варіантах її виготовлення, так як це вимагає додаткових витрат на створення відповідних тривимірних візуалізацій.

Наприклад, у випадку з м'якими меблями фабрика зазвичай оперує якимось набором оббивок, в який може входити кілька десятків варіантів, які можуть бути застосовані до різних моделей меблів, причому в різних комбінаціях. Однак виробник не може продемонструвати потенційному покупцеві весь асортимент товару, і таким чином обмежує потенційний прибуток.

У запропонованій розробці був реалізований ефективний спосіб демонстрації таких моделей потенційному покупцеві за рахунок інтерактивного меню вибору варіантів виготовлення моделі.

В інформаційній системі була передбачена функція перегляду різних варіантів виконання моделі. Технологічно це означає, що до однієї і тієї ж геометрії (полігональної сітці або «Мешу») моделі можуть бути динамічно застосовані різні матеріали (рисунок 3.3).

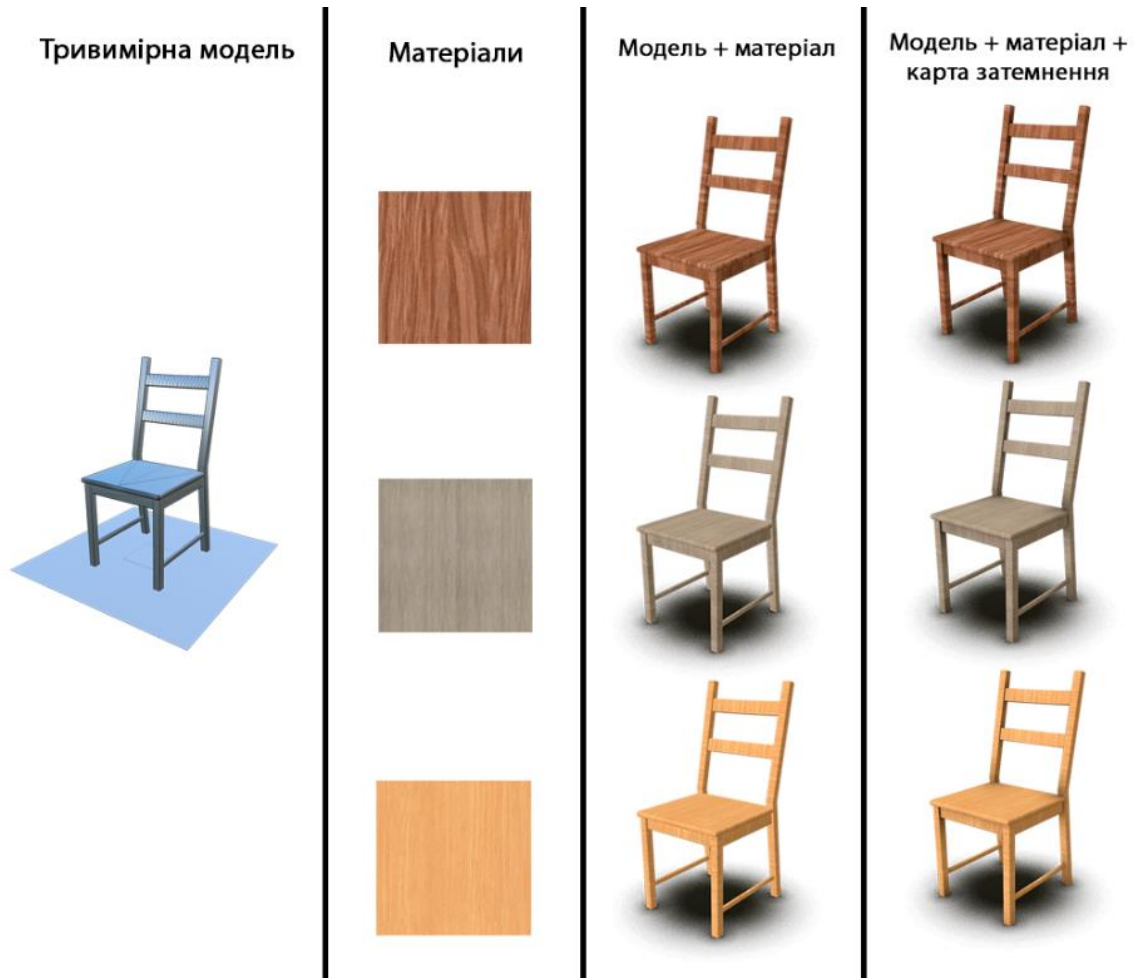


Рисунок 3.3 – Застосування різних матеріалів з текстурою затінення

Для зберігання інформації про матеріали моделі був розроблений підклас вбудованого в Unity класу MonoBehaviour, названий ColorManager. Інформація про матеріали являє собою масив назв матеріалів і масив посилань на безпосередньо об'єкти матеріалів в сховищі.

Також клас ColorManager служить і для введення інформації про матеріали моделі. При підготовці моделі об'єкта 3D-дизайнер робить налаштування окремих об'єктів матеріалів.

Після цього він за допомогою засобів графічної середовища (редактора Unity) призначає відповідні матеріали в слоти компонента ColorManager [12](рисунок 3.4).



Рисунок 3.4 – Інтерфейс компонента ColorManager в редакторі

Моделі можуть включати в себе кілька складових частин, виконаних з різних матеріалів. Наприклад, ніжки стільця можуть бути виготовлені з дерева, а сидіння оббите тканиною. У таких випадках дерев'яна частина і сидіння можуть являти собою окремі полігональні сітки з різними матеріалами.

Для таких ситуацій була спеціально передбачена можливість приєднання до окремих частин моделі індивідуальних примірників класу ColorManager.

Варіант повністю індивідуальної зміни матеріалів окремих частин має на увазі виготовлення моделі за індивідуальним замовленням і не є масовим. У масовому варіанті меблі можуть мати різні комбінації матеріалів, але вони заздалегідь підібрані і визначені фахівцем дизайнером. Внаслідок цього для спрощення логіки взаємодії вводиться узагальнений параметр – кольорове рішення.

Кольорове рішення – це сукупність матеріалів частин моделі в певній комбінації.

У графічному інтерфейсі користувача відображається інформація про доступні кольорові рішення даної моделі, що зберігається в масиві імен екземпляра класу ColorManager, прикріпленого до моделі. При виборі користувачем одного з матеріалів відбувається застосування відповідних

матеріалів до частин моделі. Алгоритм буде виглядати так, що вводиться ідентифікатор кольорового рішення, і у кожній частини об'єкта, для якої передбачено кілька матеріалів, проводиться зміна на матеріал з відповідним ідентифікатором.

Зміна колірної рішення здійснюється викликом одного з двох публічних методів класу `ColorManager`: `SetColorByName (string)` або `SetColorByIndex (int)` (рисунок 3.5). При наявності декількох окремих полігональних сіток, виклик даних методів відбувається за рахунок широкомовного повідомлення (`Component.BroadcastMessage ()`).

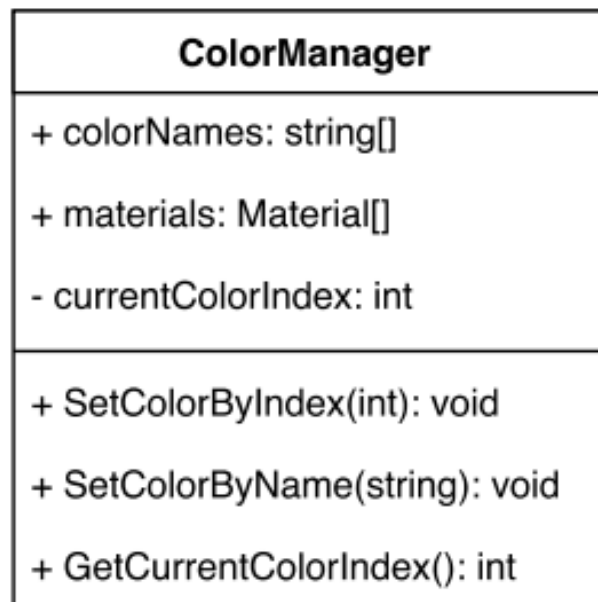


Рисунок 3.5 – UML-схема компонента зміни матеріалів

Заміна матеріалу в Unity проводиться зверненням до відповідної властивості компонента `Renderer`, прив'язаного до об'єкту: `renderer.material = newMaterial;`

Операція зміни матеріалів за часом займає частки секунди.

Крім цього, розроблено шейдер для накладання текстури затінення на головну текстуру поверхні, яку можна змінювати (рисунок 3.6).

```

Shader "Custom/Diffuse AO" {
    Properties {
        _MainTex ("Base (RGB)", 2D) = "white" {}
        _AOTex ("Ambient Occlusion (RGB)", 2D) = "white" {}
    }
    SubShader {
        Tags { "RenderType"="Opaque" }
        LOD 200

        CGPROGRAM
        #pragma surface surf Lambert

        sampler2D _MainTex;
        sampler2D _AOTex;
        struct Input {
            float2 uv_MainTex;
            float2 uv_AOTex;
        };

        void surf (Input IN, inout SurfaceOutput o) {
            half4 c = tex2D (_MainTex, IN.uv_MainTex);
            c = tex2D (_AOTex, IN.uv_AOTex) *c;
            o.Albedo = c.rgb;
            o.Alpha = c.a;
        }
        ENDCG
    }
    FallBack "Diffuse"
}

```

Рисунок 3.6 – Код шейдера для накладання текстури затінення

3.4 Висновки

У третьому розділі магістерської кваліфікаційної роботи проведено огляд та аналіз доступних засобів розробки доповненої реальності з метою оцінки їх переваг і недоліків для використання з метою дослідження. Були проаналізовані підходи для реалізації трекінгу на масових пристроях. В результаті в якості методологічної основи дослідження був обраний оптичний метод трекінгу, заснований на розпізнаванні маркерів. Запропоновано метод зміни параметрів об'єктів доповненої реальності при візуалізації, що відрізняється введенням блоку узагальненого параметра, що прискорює взаємодію, що включає можливість зміни в реальному часі і забезпечує спрощення інтерфейсу користувача. Розроблено додаток з технологією інтерактивної візуалізації тривимірних об'єктів засобами доповненої реальності в реальному часі в реальному масштабі для масових мобільних пристроїв.

4 ТЕСТУВАННЯ ПРОГРАМИ

4.1 Методи тестування

Тестування – це системний метод виявлення помилок в ПЗ шляхом виконання вихідного коду ПЗ на тестових даних, збір робочих характеристик в динаміці виконання ПЗ в конкретній операційній середовищі. Методи тестування дозволяють виявити в процесі виконання ПЗ різні помилки, дефекти і вади, викликані аномальними ситуаціями, збоями устаткування і аварійним припиненням роботи ПЗ[26].

Верифікація і атестація (валідація) – це методи, які забезпечують відповідно перевірку і аналіз правильності виконання заданих функцій і відповідності ПЗ вимогам замовника, а також заданих специфікацій ПЗ. Ці методи позначені в стандарті ISO / IEC 12207 як самостійні процеси ЖЦ і використовуються, починаючи від етапу аналізу вимог і закінчуючи перевіркою правильності функціонування програмного коду на заключному етапі – тестуванні.

Верифікація – це перевірка того, чи правильно система працює відповідно до її специфікації і заданими вимогами замовника. Цей процес ЖЦ стандарту ISO / IEC 12207 дозволяє зробити висновок про коректність розробленої системи.

Валідація є методом перевірки відповідності спроектованого ПЗ вимогам і потребам замовника і передбачає виконання на етапах ЖЦ різного роду дій для отримання коректних програм і систем.

Розглянемо наступні методи тестування: «біла скринька», «чорна скринька» та «сіра скринька».

Метод білої скриньки (white box testing, open box testing, clear box testing, glass box testing) – у тестувальника є доступ до внутрішньої структури та коду додатка, а також є достатньо знань для розуміння побаченого[29]. Виділяють навіть супутню тестування з методу білого ящика глобальну техніку –

тестування на основі дизайну (design-based testing). Для більш глибокого вивчення суті методу білого ящика рекомендується ознайомитися з техніками дослідження потоку управління або потоку даних, використання діаграм станів. Деякі автори схильні жорстко пов'язувати цей метод зі статичним тестуванням, але ніщо не заважає тестувальнику запуснути код на виконання і при цьому періодично звертатися до самого коду (а модульне тестування і зовсім передбачає запуск коду на виконання і при цьому роботу саме з кодом, а не з «Додатком повністю»).

Переваги методу білої скриньки:

- Показує приховані проблеми і спрощує їх діагностику.
- Допускає досить просту автоматизацію тест-кейсів і їх виконання на самих ранніх стадіях розвитку проекту.
- Володіє розвиненою системою метрик, збір і аналіз яких легко автоматизується.
- Стимулює розробників до написання якісного коду.
- Багато техніки цього методу є перевіреними, добре себе зарекомендували рішеннями, що базуються на суворому технічному підході.

Недоліки методу білої скриньки:

- Не може виконуватися тестувальниками, що не володіють достатніми знаннями в області програмування.
- Тестування сфокусовано на реалізованій функціональності, що підвищує ймовірність пропуску нереалізованих вимог.
- Поведінка додатки досліджується у відриві від реальної середовище програмного продукту та не враховує її вплив.
- Поведінка додатки досліджується у відриві від реальних користувальницьких сценаріїв.

Метод чорної скриньки (black box testing, closed box testing, specification-based testing) – у тестувальника або немає доступу до внутрішньої структури та коду програми, або недостатньо знань для їх розуміння, або він свідомо не звертається до них у процесі тестування[27]. При цьому абсолютна більшість

перерахованих на малюнках видів тестування працюють за методом чорного ящика, ідею якого в альтернативному визначенні можна сформулювати так: тестувальник надає на додаток впливу (і перевіряє реакцію) тим же способом, яким при реальній експлуатації додатки на нього впливали б користувачі або інші додатки[28]. В рамках тестування за методом чорного ящика основною інформацією для створення тест-кейсів виступає документація (особливо – вимоги (requirements-based testing)) і загальний здоровий сенс (для випадків, коли поведінка додатки в деякій ситуації не регламентовано явно, іноді це називають «тестуванням на основі неявних вимог», але канонічного визначення у цього підходу немає).

Переваги методу чорної скриньки:

- Тестувальник не зобов'язаний володіти (глибокими) знаннями в області програмування.
- Поведінка додатки досліджується в контексті реальної середовище програмного продукту та враховує її вплив.
- Поведінка додатки досліджується в контексті реальних користувальницьких сценаріїв.
- Тест-кейси можна створювати вже на стадії появи стабільних вимог.
- Процес створення тест-кейсів дозволяє виявити дефекти в вимогах.
- Допускає створення тест-кейсів, які можна багаторазово використовувати на різних проектах.

Недоліки методу чорної скриньки:

- Можливо повторення частини тест кейсів, вже виконаних розробниками.
- Є висока ймовірність того, що частина можливих варіантів поведінки додатки залишиться непротестованих.
- Для розробки високоефективних тест-кейсів необхідна якісна документація.
- Діагностика виявлених дефектів складніша в порівнянні з техніками методу білого ящика.

– У зв'язку з широким вибором технік і підходів важко планування і оцінка трудовитрат.

– У разі автоматизації можуть знадобитися складні дорогі інструментальні засоби.

Метод сірої скриньки (gray box testing) – комбінація методів білої скриньки і чорної скриньки, яка полягає в тому, що до частини коду і архітектури у тестувальника доступ є, а до частини – немає. Зазвичай говорять про методи білої або чорної скриньки в застосуванні до тих чи інших частин додатка, при цьому розуміючи, що «додаток повністю» тестується за методом сірої скриньки.

Методи білої і чорної скриньки не є конкуруючими або взаємовиключними – навпаки, вони гармонійно доповнюють один одного, компенсуючи таким чином наявні недоліки.

4.2 Інструкція тестування програми

Для програмного додатку «Interactive Visualization» проводиться тестування графічного інтерфейсу користувача з використанням методу «чорна скринька». Для перевірки правильності роботи програмного додатку проводимо тестування загальних функцій та процесу ігрового функціоналу кожного з модулів.

Після запуску програмного додатку відкривається сторінка авторизації. Сторінка призначена для авторизації користувача. Користувачу надається можливість ввести email, пароль та увійти в систему за допомогою відповідної кнопки «Вхід». На сторінці є «header», що складається з назви програмного додатку, зміни мови системи. При натисненні кнопки «Реєстрація» відкриється відповідне вікно реєстрації. Користувач може ввести дані у відповідні поля та натиснути «Реєстрація». Якщо email уже наявний в базі даних, то з'явиться відповідне повідомлення про дубльованість email. Після завершення реєстрації користувач повертається у вікно авторизації та вводить

відповідні email та пароль, введені при реєстрації. При відсутності введених даних, або якщо дані введені некоректно, система відправить користувачу відповідне повідомлення. Коли користувач зареєструвався і увійшов в систему з коректними email та паролем, відкривається сторінка додатку. Коли користувач авторизований, у «header» сторінки додано кнопку «Вихід», що призначена для виходу користувача з програмного комплексу. При цьому відбувається повернення на сторінку авторизації. Головна інформація про користувача відображається у головному меню програмного додатку.

Після вибору бажаної тривимірної моделі для інтеграції в реальне середовище автоматично вмикається режим доповненої реальності, запускається камера пристрою, відбувається пошук та сканування маркера та після знаходження такого в полі зору камери відбувається накладання тривимірного об'єкта (рисунок 4.1).



Рисунок 4.1 – Інтеграція тривимірної моделі в реальне середовище

Тривимірний об'єкт відображається правильно, матеріали виглядають так, як і були налаштовані в рушії. Крім того, працює метод підвищення реалістичності, а саме накладена текстура затінення та під об'єктом є падаюча тінь, що в комплексі дає набагато більше реалістичне зображення.

Наступне, що необхідно перевірити це роботу методу зміни матеріалів тривимірного об'єкта. Для цього натискаємо кнопку «Menu», з'являється підменю з вибором варіантів для 2 типів матеріалів, у даному прикладі це дерево та тканина (рисунки 4.2).



Рисунок 4.2 – Меню вибору варіантів для матеріалів

Для даної моделі заготовлено по 3 варіанта оформлення як для дерева так і для тканини. Кількість та набір матеріалів залежить від конкретної моделі та налаштовується в ігровому рушії.

Наступний крок – перевірка роботи методу зміни матеріалів. Для цього в меню вибору матеріалів обираємо пункт «Material 2» в пункті «Cloth». Після цього матеріал тканини на стільці має змінитись (рисунок 4.3).



Рисунок 4.3 – Перевірка накладання «Material 2» для тканини

Аналогічно обираємо пункт «Material 3» в пункті «Cloth» для перевірки наступного варіанту оформлення. Після цього матеріал тканини на стільці має змінитись (рисунок 4.4).



Рисунок 4.4 – Перевірка накладання «Material 3» для тканини

Також варто перевірити працездатність цієї функції для вкладки «Wood», для цього в меню вибору матеріалів обираємо пункт «Material 2» в пункті «Wood». Після цього матеріал дерева на стільці має змінитись (рисунок 4.5).



Рисунок 4.5 – Перевірка накладання «Material 2» для дерева

Отримані результати свідчать про те, що методи підвищення реалістичності тривимірних об'єктів при візуалізації в реальному середовищі

та метод зміни матеріалів об'єкта в режимі реального часу працюють в нормальному режимі.

Наступним кроком необхідно протестувати модуль маніпуляції тривимірними моделями в просторі з використанням жестових методів введення. Для цього потрібно натиснути на екран одним пальцем та протягнути вбік, після чого тривимірний об'єкт має повернутись на місці (рисунок 4.6).



Рисунок 4.6 – Перевірка методу маніпуляції віртуальних об'єктів

Після проведення тестування з використанням методу «чорна скринька» виявлено повну відповідність поставленому технічному завданню, помилок при роботі програми не виявлено, всі функції працюють. Тому методика тестування чорної скриньки підтверджує нормальний режим роботи модулю.

4.3 Висновки

Розглянуто методи тестування «чорної скриньки», «білої скриньки» та «сірої скриньки». Виявлено як переваги, так і недоліки перелічених методів.

В результаті, обрано метод тестування «чорна скринька», оскільки за даним методом тестуються лише вхідні/вихідні дані і необхідним є тестування інтерфейсу користувача, щоб виявити та виправити помилки, що виникають під час роботи програми.

Було проведено тестування інтерфейсу та всіх основних модулів програмного додатку. Протестовано роботу методів підвищення реалістичності, а саме накладення текстур затінення та падаюча тінь. Перевірено роботу методу зміни матеріалів тривимірного об'єкта та методу маніпуляції тривимірними моделями в просторі з використанням жестових методів введення.

Тестування програми показало повну її працездатність та відповідність поставленому технічному завданню.

5 ЕКОНОМІЧНА ЧАСТИНА

5.1 Оцінювання комерційного потенціалу розробки

Метою проведення технологічного аудиту є оцінювання комерційного потенціалу розробки. Для проведення технологічного аудиту було залучено 3-х незалежних експертів. Такими експертами будуть: Бабюк Наталя Петрівна (к.т.н., доц. кафедри ПЗ ВНТУ), Іванчук Ярослав Володимирович (к.т.н., доц. кафедри КН ВНТУ), Черноволик Галина Олександрівна (к.т.н., доц. кафедри ПЗ ВНТУ).

Проведено оцінювання комерційного потенціалу за критеріями, наведеними в таблиці 5.1 [30].

Таблиця 5.1 - Критерії оцінювання комерційного потенціалу розробки
бальна оцінка

Критерії оцінювання та бали (за 5-ти бальною шкалою)					
Кри-терій	0	1	2	3	4
Технічна здійсненність концепції:					
1	Достовірність концепції не підтверджується	Концепція підтверджується експертними висновками	Концепція підтверджується розрахунками	Концепція перевірена на практиці	Перевірено роботоздатність продукту в реальних умовах
Ринкові переваги (недоліки):					
2	Багато аналогів на малому ринку	Мало аналогів на малому ринку	Кілька аналогів на великому ринку	Один аналог на великому ринку	Продукт не має аналогів на великому ринку
3	Ціна продукту значно вища за ціни аналогів	Ціна продукту дещо вища за ціни аналогів	Ціна продукту приблизно дорівнює цінам аналогів	Ціна продукту дещо нижче за ціни аналогів	Ціна продукту значно нижче за ціни аналогів

Продовження таблиці 5.1 – Критерії оцінювання комерційного потенціалу розробки бальна оцінка

4	Технічні та споживчі властивості продукту значно гірші, ніж в аналогів	Технічні та споживчі властивості продукту трохи гірші, ніж в аналогів	Технічні та споживчі властивості продукту на рівні аналогів	Технічні та споживчі властивості продукту трохи кращі, ніж в аналогів	Технічні та споживчі властивості продукту значно кращі, ніж в аналогів
5	Експлуатаційні витрати значно вищі, ніж в аналогів	Експлуатаційні витрати дещо вищі, ніж в аналогів	Експлуатаційні витрати на рівні експлуатаційних витрат аналогів	Експлуатаційні витрати трохи нижчі, ніж в аналогів	Експлуатаційні витрати значно нижчі, ніж в аналогів
Ринкові перспективи					
6	Ринок малий і не має позитивної динаміки	Ринок малий, але має позитивну динаміку	Середній ринок з позитивною динамікою	Великий стабільний ринок	Великий ринок з позитивною динамікою
7	Активна Конкуренція великих компаній на ринку	Активна конкуренція	Помірна конкуренція	Незначна конкуренція	Конкуренція немає
Практична здійсненність					
8	Відсутні фахівці як з технічної, так і з комерційної реалізації ідеї	Необхідно наймати фахівців або витратити значні кошти та час на навчання наявних фахівців	Необхідне значне навчання фахівців та збільшення їх штату	Необхідне незначне навчання фахівців	Є фахівці з питань як з технічної, так і з комерційної реалізації ідеї
9	Потрібні значні фінансові ресурси, які відсутні. Джерела фінансування ідеї відсутні	Потрібні незначні фінансові ресурси. Джерела фінансування відсутні	Потрібні значні фінансові ресурси. Джерела фінансування є	Потрібні незначні фінансові ресурси. Джерела фінансування є	Не потребує додаткового фінансування

Продовження таблиці 5.1 – Критерії оцінювання комерційного потенціалу розробки бальна оцінка

Критерії оцінювання та бали (за 5-ти бальною шкалою)					
Кри-терій	0	1	2	3	4
10	Необхідна Розробка нових матеріалів	Потрібні матеріали, що використовуються у військово-промислово му комплексі	Потрібні дорогі матеріали	Потрібні досяжні та дешеві матеріали	Всі матеріали для реалізації ідеї відомі та давно використовуються у виробництві
11	Термін реалізації ідеї більший за 10 років	Термін реалізації ідеї більший за 5 років. Термін окупності інвестицій більше 10-ти років	Термін реалізації ідеї від 3-х до 5-ти років. Термін окупності інвестицій більше 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій від 3-х до 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій менше 3-х років
12	Необхідна розробка регламентних документів та отримання великої кількості дозвільних документів на виробництво та реалізацію продукту	Необхідно отримання великої кількості дозвільних документів на виробництво та реалізацію продукту, що вимагає значних коштів та часу	Процедура отримання дозвільних документів для виробництва та реалізації продукту вимагає незначних коштів та часу	Необхідно тільки повідомлення відповідним органам про виробництво та реалізацію продукту	Відсутні будь-які регламентні обмеження на виробництво та реалізацію продукту

Результати оцінювання комерційного потенціалу розробки наведено в таблиці 5.2.

Таблиця 5.2 – Результати оцінювання комерційного потенціалу розробки

Критерії	Прізвище, ініціали, посада експерта		
	1. Бабюк Н.П. (к.т.н., доц. каф. ПЗ ВНТУ)	2. Іванчук Я.В. (к.т.н., доц. каф. КН ВНТУ)	3. Черноволик Г.О. (к.т.н., доц. каф. ПЗ ВНТУ)
	Бали, виставлені експертами:		
1	4	4	4
2	3	3	3
3	3	3	3
4	4	4	3
5	3	3	3
6	4	4	4
7	2	3	3
8	4	4	4
9	4	4	4
10	4	4	4
11	4	4	4
12	4	4	4
Сума балів	СБ ₁ = 43	СБ ₂ = 44	СБ ₂ = 43
Середньоарифметична сума балів $\overline{СБ}$	$\overline{СБ} = \frac{\sum_{i=1}^3 СБ_i}{2} = 43,3$		

Отже, з отриманих даних таблиці 5.1 видно, що рівень комерційного потенціалу розробки, становить 43,3 бали, що відповідає рівню «високий».

У розділі 1.3 було проаналізовано поточний стан галузі програмних додатків для інтерактивної візуалізації тривимірних об'єктів. За результатами аналізу, розроблюваний продукт містить компоненту, відсутню у аналогів, тому є новим на ринку.

У розділі 1.3 було проаналізовано аналоги, за результатами аналізу, розроблюваний продукт поєднує або покращує функціонал усіх додатків та покриває недоліки існуючих рішень.

Жоден із аналогів не містить методу покращення реалістичності візуалізації тривимірних об'єктів в реальному середовищі, тому не задовольняють вимоги, що задовольняє розробка.

Розроблюваний продукт на ринку буде новим, оскільки існуючі аналоги не мають поєднання методів зміни параметрів об'єктів при візуалізації в реальному часі та алгоритму маніпуляції тривимірними об'єктами за допомогою жестових методів вводу. Цільова аудиторія – люди віком 10 – 40 років, які бажають відобразити віртуальний об'єкт в реальному довкіллі, магазини продажу чи виготовлення меблів, компанії, які виготовляють продукцію під замовлення з різними параметрами.

Додаток дозволяє користувачу побачити будь-який об'єкт чи створіння поруч з собою. Це може бути як об'єкт інтер'єру який користувач бажає придбати чи улюблений герой кіно чи мультфільму в кімнаті чи на подвір'ї.

В подальшому можливе вдосконалення проекту, шляхом додання нових об'єктів доповненої реальності та створення хмарної бази даних тривимірних моделей для завантаження на мобільні пристрої.

Розробка додатку не має обмежень, окрім потреби у фінансових інвестиціях, які необхідно вкласти для популяризації та публікації додатку.

Як майданчик для продажу можна використовувати Google Play Market, оскільки це найпопулярніший майданчик поширення Android додатків. Крім того на платформі Google Ads є можливість прорекламувати власний додаток для залучення більшої кількості користувачів.

5.2 Прогнозування витрат на виконання науково-дослідної роботи та конструкторсько-технологічної роботи

Для розробки нового програмного продукту необхідні такі витрати.

Основна заробітна плата для розробників визначається за формулою (5.1):

$$Z_o = \frac{M}{T_p} \cdot t, \quad (5.1)$$

де M – місячний посадовий оклад конкретного розробника;

T_p – кількість робочих днів у місяці, $T_p = 22$ дні;

t – число днів роботи розробника, $t = 45$ днів.

Розрахунки заробітних плат для керівника і програміста наведені в таблиці 5.3.

Таблиця 5.3 – Розрахунки основної заробітної плати

Працівник	Оклад M , грн.	Оплата за робочий день, грн.	Число днів роботи, t	Витрати на оплату праці, грн.
Керівник проекту	28000	1273	7	8911
Unity-розробник	24000	1091	45	49095
Всього:				58006

За формулою 5.2 обчислюється основна заробітна плата робітників Z_p , якщо вони беруть участь у виконанні даного етапу роботи і виконують роботи за робочими професіями у випадку, коли вони працюють в наукових установах бюджетної сфери.

$$Z_p = \sum_1^n t_i \cdot C_i, \quad (5.2)$$

де t_i – норма часу (трудомісткість) на виконання конкретної роботи, годин;

n – число робіт по видах та розрядах;

C_i – погодинна тарифна ставка робітника відповідного розряду, який виконує відповідну роботу, грн./год. C_i визначається за формулою 5.3.

$$C_i = \frac{M_m \cdot K_i}{T_p \cdot T_{zm}}, \quad (5.3)$$

де M_m – розмір мінімальної заробітної плати за місяць, грн, станом на грудень 2019 $M_m = 4173$ грн.;

K_i – тарифний коефіцієнт робітника відповідного розряду, для 6-го розряду тарифний коефіцієнт становить 1,45; для 7-го розряду – 1,54;

T_p – число робочих днів в місяці, $T_p = 22$ дні;

T_{zm} – тривалість зміни, $T_{zm} = 8$ годин.

Виконані розрахунки занесено до таблиці 5.3.

Додаткова заробітна плата Z_d розраховується за формулою 5.4, як 12% від основної заробітної плати робітників.

Таблиця 5.4 – Витрати на оплату праці робітників

Найменування робіт	Трудомісткість, нормо-годин	Розряд роботи	Погодинна тарифна ставка, грн.	Величина оплати, грн.
Тестування	15	6	26,36	395,5
Відлагоджування	10	7	28	280
Разом				675,5

$$Z_d = Z_o \cdot 0,12 \quad (5.4)$$

$$Z_d = (58006 + 675,5) \cdot 0,12 = 7\,041,78 \text{ (грн.)}$$

Нарахування на заробітну плату розробників та робітників обчислюються за формулою 5.5. Згідно діючого законодавства нарахування на заробітну плату складають 22% від суми основної та додаткової заробітної плати.

$$H_3 = (Z_o + Z_p + Z_d) \cdot 0,22 \quad (5.5)$$

$$H_3 = (58006 + 675,5 + 7041,78) \cdot 0,22 = 14\,459,12 \text{ (грн.)}$$

Обчислимо амортизацію обладнання, комп'ютерів та приміщень, що використовувались для розробки. В спрощеному вигляді амортизаційні відрахування розраховується за формулою 5.6.

$$A = \frac{Ц \cdot H_a \cdot T}{100 \cdot 12}, \quad (5.6)$$

де Ц – загальна балансова вартість всього обладнання, комп'ютерів, приміщень тощо, що використовувались для виконання даного етапу роботи, грн;

H_a – річна норма амортизаційних відрахувань. Приймаємо, що $H_a = 20\%$;

T – термін, використання обладнання, приміщень, місяці.

Для прикладу, наведено розрахунок амортизаційних витрат на робочий комп'ютер, балансова вартість якого становить 18000 грн., а термін його використання – 60 днів.

$$A = \frac{18000 \cdot 20}{100} \cdot \frac{2}{12} = 600 \text{ (грн.)}$$

Амортизаційні витрати на інше обладнання визначається аналогічно. Розрахунки амортизаційних відрахувань занесено в таблицю 5.5.

Таблиця 5.5 - Величина амортизаційних відрахувань

№	Найменування обладнання, приміщень	Балансова вартість, грн.	Термін використання, міс.	Величина амортизаційних відрахувань, грн.
1	Робочий комп'ютер	18000	2	600
2	Робочий комп'ютер з графічним планшетом	22000	2	733,33
Всього:				1333,33

Витрати на силову електроенергію обчислюються за формулою 5.7.

$$V_e = V \cdot \Pi \cdot \Phi \cdot K_{\Pi}, \quad (5.7)$$

де V — вартість 1 кВт - години електроенергії для підприємств, $V = 2,3$ грн./кВт – година; [32]

Π — встановлена потужність обладнання, кВт. $\Pi = 1,7$ кВт;

Φ — фактична кількість годин роботи обладнання, годин. $\Phi = 280$ годин;

K_{Π} — коефіцієнт використання потужності, $K_{\Pi} = 0,8$.

$$V_e = 2,3 \cdot 1,7 \cdot 280 \cdot 0,8 = 875,84 \text{ (грн.)}$$

Інші витрати $V_{ін}$ охоплюють: витрати на управління організацією, оплата службових відряджень, витрати на утримання, ремонт та експлуатацію основних засобів, витрати на опалення, освітлення, водопостачання, охорону праці тощо. Інші витрати приймаємо як 250% від суми основної заробітної плати розробників та робітників і обчислюються за формулою 5.8.

$$V_{ін} = (Z_o + Z_p) \cdot 2.5 \quad (5.8)$$

$$V_{ін} = (58006 + 675,5) \cdot 2.5 = 146\,703,75 \text{ (грн.)}$$

Загальні витрати на розробку проекту знаходиться як сума усіх витрат по проекту. За формулою 5.9 розраховано загальні витрати на розробку проекту.

$$V = Z_o + Z_p + Z_d + H_3 + A + V_e + V_{ін} \quad (5.9)$$

$$\begin{aligned} V &= 58006 + 675,5 + 7\,041,78 + 14\,459,12 + 1333,33 + 875,84 + 146\,703,75 = \\ &= 229\,095,32 \text{ (грн.)} \end{aligned}$$

У випадку, коли дипломник виконує лише частину роботи, то загальна вартість всієї наукової роботи $V_{\text{заг}}$ визначається за формулою 5.10:

$$V_{\text{заг}} = V/\alpha, \quad (5.10)$$

де α – частка витрат, які безпосередньо здійснює виконавець даного етапу роботи, у відносних одиницях, приймається 0,9.

$$V_{\text{заг}} = 229\,095,32 / 0,9 = 254\,550,35 \text{ (грн)}.$$

Прогнозування загальних витрат ЗВ на виконання та впровадження результатів виконаної наукової роботи виконано за формулою 5.11.

$$ЗВ = V_{\text{заг}} / \beta, \quad (5.11)$$

де β – коефіцієнт, який характеризує етап (стадію) виконання даної роботи. Так як розробка знаходиться на стадії розробки промислового зразка, то $\beta \approx 0,7$.

$$ЗВ = 254\,550,35 / 0,7 = 363\,643,36 \text{ (грн)}.$$

5.3 Прогнозування комерційних ефектів від реалізації результатів розробки

Спрогнозуємо отримання прибутку від реалізації результатів нашої розробки. Зростання чистого прибутку можна оцінити у теперішній вартості грошей. Це забезпечить підприємству (організації) надходження додаткових коштів, які дозволять покращити фінансові результати діяльності.

Оцінка зростання чистого прибутку підприємства від впровадження результатів наукової розробки. У цьому випадку збільшення чистого прибутку

підприємства $\Delta \Pi_i$ для кожного із років, протягом яких очікується отримання позитивних результатів від впровадження розробки, розраховується за формулою:

$$\Delta \Pi_i = \sum_1^n (\Delta \Pi_{\text{я}} \cdot N + \Pi_{\text{я}} \cdot \Delta N)_i, \quad (5.12)$$

де $\Delta \Pi_{\text{я}}$ – покращення основного якісного показника від впровадження результатів розробки у даному році;

N – основний кількісний показник, який визначає діяльність підприємства у даному році до впровадження результатів наукової розробки;

ΔN – покращення основного кількісного показника діяльності підприємства від впровадження результатів розробки;

$\Pi_{\text{я}}$ – основний якісний показник, який визначає діяльність підприємства у даному році після впровадження результатів наукової розробки;

n – кількість років, протягом яких очікується отримання позитивних результатів від впровадження розробки.

Припустимо, що внаслідок впровадження результатів наукової розробки чистий прибуток підприємства збільшиться на 10 грн., а кількість одиниць реалізованої послуги збільшиться протягом першого року – на 45000 од., протягом другого року – на 25000 од., протягом третього року – на 10000 од.

Орієнтовно, реалізація продукції до впровадження результатів наукової розробки складала 1 шт., а прибуток, що його отримувало підприємство на одиницю реалізованої продукції до впровадження результатів наукової розробки – 50 грн.

Потрібно спрогнозувати збільшення чистого прибутку підприємства від впровадження результатів наукової розробки у кожному році відносно базового.

Збільшення чистого прибутку підприємства $\Delta \Pi_1$ протягом першого року складе:

$$\Delta\Pi_1 = 10 \cdot 1 + (50 + 10) \cdot 45000 = 2\,700\,010 \text{ (грн)}.$$

Збільшення чистого прибутку підприємства $\Delta\Pi_2$ протягом другого року складе:

$$\Delta\Pi_2 = 10 \cdot 1 + (50 + 10) \cdot (45000 + 25000) = 4\,200\,010 \text{ (грн)}.$$

Збільшення чистого прибутку підприємства $\Delta\Pi_3$ протягом третього року складе:

$$\Delta\Pi_3 = 10 \cdot 1 + (50 + 10) \cdot (45000 + 25000 + 10000) = 4\,800\,010 \text{ (грн)}.$$

За результатами прогнозування зроблено висновок, що впровадження наукової розробки приведе до значного зростання чистого прибутку підприємства.

5.4 Розрахунок ефективності вкладених інвестицій та період їх окупності

Оцінка вкладених інвестицій проводиться з метою визначення доцільності фінансування проекту. Показниками доцільності є абсолютна і відносна ефективність вкладених інвестицій, термін їх окупності.

Спочатку розраховується теперішня вартість інвестицій PV , що вкладаються в наукову розробку. Такою вартістю можна вважати прогнозовану величину загальних витрат ZB на виконання та впровадження результатів НДДКР, розраховану раніше за формулою (5.11), тобто приймаємо значення $PV = ZB = 363\,643,36$ (грн).

Наступним кроком розраховується очікуване збільшення прибутку $\Delta\Pi_i$, що його отримає підприємство від впровадження результатів наукової розробки, для кожного із років, починаючи з першого року впровадження. Розрахунок було виконано раніше, у розділі 5.3.

Для спрощення подальших розрахунків будується вісь часу, на яку наносяться всі платежі (інвестиції та прибутки), що мають місце під час виконання науково-дослідної роботи та впровадження її результатів. Платежі показуються у ті терміни, коли вони здійснюються.

На рисунку 5.1 наведено графік, що характеризує рух платежів (інвестицій та додаткових прибутків).



Рисунок 5.1 – Вісь часу з фіксацією платежів, що мають місце під час розробки та впровадження результатів НДДКР

Далі, за формулою 5.13 розраховується абсолютна ефективність вкладених інвестицій $E_{\text{абс}}$.

$$E_{\text{абс}} = (\text{ПП} - \text{PV}), \quad (5.13)$$

де ПП – приведена вартість всіх чистих прибутків, що їх отримає підприємство від реалізації результатів наукової розробки, грн., розраховується за формулою 5.14.

$$\text{ПП} = \sum_1^T \frac{\Delta \Pi_i}{(1+\tau)^t}, \quad (5.14)$$

де $\Delta\Pi_i$ – збільшення чистого прибутку у кожному із років, протягом яких виявляються результати виконаної та впровадженої НДДКР, грн;

t – період часу, протягом якого виявляються результати впровадженої НДДКР, роки;

τ – ставка дисконтування, за яку можна взяти щорічний прогнозований рівень інфляції в країні; для України цей показник знаходиться на рівні 0,1;

t – період часу (в роках) від моменту отримання чистого прибутку до точки „0”.

Розрахунки за формулами наведено нижче.

$$ПП = 2700 / (1+0,1)^2 + 4200 / (1+0,1)^3 + 4800 / (1+0,1)^4 = 8\,665\,384 \text{ (грн.)};$$

$$E_{abc} = 8\,665\,384 - 363\,643,36 = 8\,301\,740,64 \text{ (грн.)}.$$

Так як $E_{abc} > 0$, то результат від проведення наукових досліджень та їх впровадження принесе прибуток, але це також ще не свідчить про те, що інвестор буде зацікавлений у фінансуванні даного проекту.

За формулою 5.15 розрахується відносна (щорічну) ефективність вкладених у наукову розробку інвестицій E_B

$$E_B = \sqrt[T_{ж}]{1 + \frac{E_{abc}}{PV}}, \quad (5.15)$$

де E_{abc} – абсолютна ефективність вкладених інвестицій, грн;

$T_{ж}$ – життєвий цикл наукової розробки, роки.

Підрахунок наведено нижче.

$$E_B = \sqrt[3]{1 + \frac{8301740,64}{363643,36}} - 1 = 1,21 \text{ або } 121 \%$$

Розрахована величина E_B порівнюється з мінімальною (бар'єрною) ставкою дисконтування τ_{\min} , яка визначає ту мінімальну дохідність, нижче за яку інвестиції вкладатися не будуть. У загальному вигляді мінімальна (бар'єрна) ставка дисконтування визначається за формулою 5.16.

$$\tau_{\min} = d + f, \quad (5.16)$$

де d – середньозважена ставка за депозитними операціями в комерційних банках; в 2019 році в Україні $d = (0,14...0,2)$;

f – показник, що характеризує ризикованість вкладень; зазвичай, величина $f = (0,05...0,1)$.

Приймаємо $d = 0,2$, $f = 0,1$, тоді розрахунок матиме наступний вигляд:

$$\tau_{\min} = 0,2 + 0,1 = 0,3.$$

Оскільки $E_B = 121\% > \tau_{\min} = 30\%$, то у інвестора буде причина інвестувати кошти у розробки проекту, так як він отримає прибуток, значно вищий за можливий прибуток від депозиту в банку.

Останнім етапом є розрахунок терміну окупності вкладених у реалізацію наукового проекту коштів. Термін окупності вкладених у реалізацію наукового проекту інвестицій $T_{ок}$ розраховується за формулою 5.17.

$$T_{ок} = 1 / E_B \quad (5.17)$$

$$T_{ок} = 1 / 1,21 = 0,83 \text{ (роки)}.$$

Термін окупності $T_{ок} = 0,8 < 3$ років, що свідчить про швидку окупність проекту. На основі виконаних розрахунків можна зробити висновок, що фінансування наукової розробки є доцільним.

5.5 Висновки

Проведено оцінювання комерційного потенціалу розробки із залученням незалежних експертів. За результатами оцінки проект показав комерційний потенціал рівня «високий».

Проведено порівняльний аналіз технічних показників розроблюваного проекту із проектами-аналогами. За результатами аналізу визнано, що нова розробка перевершує існуючі аналоги.

Виконано прогнозування витрат на виконання наукової роботи та впровадження її результатів. Загальні витрати на виконання та впровадження результатів наукової розробки дорівнюють 363 643,36 грн.

Виконано прогнозування комерційних ефектів від реалізації результатів розробки. За результатами розрахунків, збільшення чистого прибутку підприємства рівне 2700 тисяч гривень, 4200 тисяч гривень, 4800 тисяч гривень за перший, другий і третій роки після впровадження результатів наукової розробки відповідно.

Розраховано ефективність вкладених інвестицій та періоду їх окупності. Абсолютна ефективність вкладених інвестицій рівна 8 665 384 гривень; відносна ефективність вкладених інвестицій рівна 121%; термін окупності – 0,83 року.

На основі проведених розрахунків зроблено висновок про доцільність наукової розробки.

ВИСНОВКИ

Дану роботу присвячено процесу та методам інтерактивної візуалізації тривимірних об'єктів в реальному середовищі. Розроблений додаток призначений для покращення методів реалістичного відображення та швидкодії інтерактивної візуалізації тривимірних об'єктів в реальному часі та в реальному середовищі.

Було проаналізовано стан проблеми сучасної візуалізації тривимірних об'єктів. Розглянуто основні аналоги, визначено їх особливості та недоліки і розроблено порівняння з власним програмним продуктом.

Удосконалено метод підвищення реалістичності відображення тривимірних об'єктів при візуалізації в реальному часі. Розроблено метод зміни параметрів об'єктів при візуалізації в реальному часі та метод підготовки і зберігання тривимірних моделей для покращення швидкодії реалістичної візуалізації на мобільних пристроях.

Розроблено алгоритм маніпуляції віртуальними тривимірними об'єктами в просторі для підвищення інтерактивності візуалізації.

Обрано середовище розробки – ігровий рушій Unity з використанням спеціалізованого засобу для доповненої реальності Vuforia для реалізації поставленої задачі.

Розглянуто методи тестування «чорної скриньки», «білої скриньки» та «сірої скриньки». Виявлено як переваги, так і недоліки перелічених методів. За обраним методом «чорної скриньки» було проведено тестування інтерфейсу та всіх основних модулів програмного додатку, яке підтвердило його повну працездатність.

Виконано економічний аналіз та розрахунки, результати яких підтвердили доцільність розробки.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Комп'ютерна 3D-графіка [Електронний ресурс] // Режим доступу: <https://uk.wikipedia.org/wiki/>
2. Интеграция 3D в фотографию [Електронний ресурс] // Режим доступу: http://www.3dmir.ru/s_tutor/tutor/309.html
3. Білик В. О. Методи інтерактивної візуалізації тривимірних об'єктів у реальному середовищі / Інформаційні технології і автоматизація – 2019 / В.О. Білик, Н.П. Бабюк // Матеріали Міжнародної Інтернет-конференції, м.Одеса, 2019. – [Електронний ресурс]. Режим доступу до матеріалу: https://card-file.onaft.edu.ua/bitstream/123456789/10174/1/Information_technologies_and_automation_2019_%d0%a01.pdf
4. AR — Дополненная Реальность [Електронний ресурс] // Режим доступу: <https://habr.com/ru/post/419437/>
5. Доповнена реальність, або AR-технології. Як це працює? [Електронний ресурс] // Режим доступу: <http://thefuture.news/page1837780.html>
6. Дополненная реальность: что собой представляет и где используется? [Електронний ресурс] // Режим доступу: <https://livesurf.ru/zhurnal/6076-dopolnennaya-realnost-chto-soboj-predstavlyayet-i-gde-ispolzuetsya.html>
7. JigSpace – Presentations for AR [Електронний ресурс] // Режим доступу: <https://jig.space/>
8. Houzz | Home Remodeling and Design [Електронний ресурс] // Режим доступу: <https://www.houzz.ru/?irs=US>
9. Holo – Holograms for Videos in Augmented Reality [Електронний ресурс] // Режим доступу: <https://play.google.com/store/apps/details?id=com.eighti.holo.android&hl=ru>
10. IKEA Place app [Електронний ресурс] // Режим доступу: <https://www.ikea.com/gb/en/customer-service/mobile-apps/>

11. Глобальный рынок Дополненной реальности 2014-2018 [Электронный ресурс] Режим доступа: <http://www.technavio.com/report/global-augmented-realitymarket-2014-2018>
12. Керівництво: Writing Surface Shaders [Электронный ресурс] // Режим доступа: <http://docs.unity3d.com/ru/current/Manual/SLSurfaceShaders.html>
13. Енджел Е. Интерактивная компьютерная графика. Вводный курс на базе OpenGL: пер. с англ //М.: Вільямс. – 2001. – Т. 2
14. 3D models : requirements and limitations - Augment Help Center [Электронный ресурс] // Режим доступа: <http://www.augmentedev.com/help/requirements-and-limitations>
15. Augmented Reality & Virtual Reality Market by Technology Types, Sensors (Accelerometer, Gyroscope, Haptics), Components (Camera, Controller, Gloves, HMD), Applications (Automotive, Education, Medical, Gaming, Military) & by Geography - Global Forecast and Analysis to 2013 - 2018 [Электронный ресурс] Режим доступа: <http://www.marketsandmarkets.com/Market-Reports/augmented-reality-virtual-realitymarket-1185.html>
16. Bowman D. A. et al. New Directions in 3D User Interfaces //IJVR. – 2006. – Т. 5. – №. 2. – С. 3-14.
17. Cook J. et al. Real-time photo-realistic augmented reality for interior design //ACM SIGGRAPH 2003 Sketches & Applications. – ACM, 2003. – С. 1-1.
18. Foley J. D. et al. Fundamentals of interactive computer graphics. – Reading, MA: Addison-Wesley, 1982. – Т. 2.
19. How should I define the sizes of my targets [Электронный ресурс] // Режим доступа: <https://developer.vuforia.com/forum/faq/technical-how-should-i-define-sizes-my-targets>
20. Koller D. et al. Real-time vision-based camera tracking for augmented reality applications //Proceedings of the ACM symposium on Virtual reality software and technology. – ACM, 1997. – С. 87-94.

21. Yemez Y., Schmitt F. 3D reconstruction of real objects with high resolution shape and texture //Image and Vision computing. – 2004. – Т. 22. – №. 13. – С. 1137- 1153.
22. Unreal Engine [Електронний ресурс] // Режим доступу: <https://www.unrealengine.com/en-US/>
23. Unity – Game Engine [Електронний ресурс] // Режим доступу: <http://unity3d.com/ru/>
24. ARToolKit [Електронний ресурс] // Режим доступу: <http://www.hitl.washington.edu/artoolkit/>
25. Metaio SDK [Електронний ресурс] // Режим доступу: <https://habr.com/ru/company/intel/blog/198716/>
26. Степанченко И. В. Методы тестирования программного обеспечения: Учеб. пособие / И. В. Степанченко. – ВолгГТУ. – Волгоград, 2006. – 74 с.
27. Тестування по стратегії чорної скрині [Електронний ресурс]. – Режим доступу: https://ru.wikipedia.org/wiki/Тестування_по_стратегії_чорної_скрині
28. Бейзер Б. Тестирование чёрного ящика. Технологии функционального тестирования программного обеспечения и систем. / Б. Бейзер – СПб.: Питер, 2004. – 320 с.
29. Тестування білої скрині [Електронний ресурс]. – Режим доступу: https://ru.wikipedia.org/wiki/Тестування_білої_скрині
30. Методичні вказівки до виконання студентами-магістрантами економічної частини магістерських кваліфікаційних робіт / Уклад. В. О. Козловський – Вінниця: ВНТУ, 2012. – 22 с

ДОДАТКИ

Додаток А. Технічне завдання

Міністерство освіти і науки України
Вінницький національний технічний університет
Факультет інформаційних технологій та комп'ютерної інженерії
Кафедра програмного забезпечення

ЗАТВЕРДЖУЮ

Завідувач кафедри ПЗ

д.т.н., професор Романюк О. Н.

« ____ » _____ 2019 року

Технічне завдання
на магістерську кваліфікаційну роботу за спеціальністю
121 – Інженерія програмного забезпечення

Керівник магістерської кваліфікаційної роботи:

_____ к.т.н., доц. Н.П. Бабюк

" ____ " _____ 2019 р.

Виконав:

_____ студент гр.2ПІ-18м В.О. Білик

" ____ " _____ 2019 р.

1.1 Найменування та галузь застосування

Розробка методу інтерактивної візуалізації тривимірних об'єктів у реальному середовищі.

Згідно отриманого завдання кінцевий програмний продукт може використовуватись офіційною організацією.

1.2 Підстава для розробки

Підставою для розробки даної магістерської кваліфікаційної роботи є рішення засідання кафедри програмного забезпечення (протокол №__ від «__» _____ 20__ року).

1.3 Мета та призначення розробки

Основною метою роботи є покращення працездатності та швидкодії інтерактивної візуалізації тривимірних об'єктів у реальному часі та в реальному середовищі за рахунок використання засобів доповненої реальності.

Основними задачами дослідження є:

- провести порівняльний аналіз методологічних, алгоритмічних і програмних засобів;
- покращити метод інтерактивної візуалізації у реальному часі;
- удосконалити метод підвищення реалістичності відображення тривимірних об'єктів при візуалізації у реальному часі;
- розробити метод зміни параметрів об'єктів при візуалізації в реальному часі;
- розробити метод підготовки та зберігання тривимірних моделей для покращення швидкодії реалістичної візуалізації на мобільних пристроях;
- розробити алгоритм маніпуляції віртуальними тривимірними об'єктами в просторі для підвищення інтерактивності візуалізації;
- сформулювати рекомендації з проектування інтерфейсу користувача для візуалізації.

1.4. Технічні вимоги

Склад комплексу технічних засобів

Для розв'язку задачі буде використовуватися персональна електронно-обчислювальна машина (ПЕОМ).

Вимоги до складових частин комплексу технічних засобів.

Для нормальної роботи програми, необхідний мобільний пристрій з наступною мінімальною конфігурацією:

- Android 8.0 і більший
- 30мб вільного місця в пам'яті телефону

Вимоги до програмного забезпечення.

Програмний продукт розробляється на мові програмування C# у середовищі Unity.

1.5. Перелік технічної документації, що пред'являється по закінченню робіт

- пояснювальна записка;
- технічне обґрунтування доцільності розробки;
- лістинги програми.

1.6 Стадії і етапи розробки

Завдання на проектування видане _____ 2019 року. Проектування та дослідження повинно бути завершеним до _____ 2019 року.

Аналіз предметної області та існуючих аналогів	
Розробка методу інтерактивної візуалізації тривимірних об'єктів у реальному середовищі	
Розробка програмного засобу для інтерактивної візуалізації тривимірних об'єктів у реальному середовищі	
Тестування програм	
Економічна частина	

1.7 Порядок контролю і приймання

Порядок контролю і приймання роботи регламентується відповідними документами ВНТУ і державними стандартами.

Додаток Б. Лістинг програмного засобу

«MyDragBehaviour.cs»

```
using UnityEngine;
using System.Collections;

public class MyDragBehaviour : MonoBehaviour
{
    private float maxPickingDistance = 2000;

    private Transform pickedObject = null;

    void Start()
    {
    }

    void Update()
    {
        foreach (Touch touch in Input.touches)
        {
            Debug.Log("Touching at: " + touch.position);

            //Gets the ray at position where the screen is touched
            Ray ray = Camera.main.ScreenPointToRay(touch.position);

            if (touch.phase == TouchPhase.Began)
            {
                Debug.Log("Touch phase began at: " + touch.position);

                RaycastHit hit = new RaycastHit();
                if (Physics.Raycast(ray, out hit, maxPickingDistance))
                {
                    pickedObject = hit.transform;
                }
                else
                {
                    pickedObject = null;
                }
            }
            else if (touch.phase == TouchPhase.Moved)
            {
                Debug.Log("Touch phase Moved");
            }
        }
    }
}
```



```

SubShader {
    Tags { "RenderType"="Opaque" }
    LOD 200

    CGPROGRAM
    #pragma surface surf Lambert

    sampler2D _MainTex;
    sampler2D _AOTex;
    struct Input {
        float2 uv_MainTex;
        float2 uv_AOTex;
    };

    void surf (Input IN, inout SurfaceOutput o) {
        half4 c = tex2D (_MainTex, IN.uv_MainTex);
        c = tex2D (_AOTex, IN.uv_AOTex) *c;
        o.Albedo = c.rgb;
        o.Alpha = c.a;
    }
    ENDCG
}
Fallback "Diffuse"
}

```

«TextureChange.cs»

```

using UnityEngine;
using System.Collections;

public class MasterButton: MonoBehaviour
{
    public bool isButtonSelected = false;
    //The Textures you wish to use for swapping ( could use a Texture2D[] if you
have lots)
    public Texture2D onTexture;
    public Texture2D offTexture;
    //The MeshRenderer of the object your trying to swap textures on
    public MeshRenderer currentRenderer;

    // Use this for initialization
    void Start ()

```

```
{
//Find the MeshRenderer Component (could also use FindObjectOfType() )
    currentRenderer = this.GetComponent<MeshRenderer>();
}

// Update is called once per frame
void Update ()
{
    if(currentRenderer != null)
    {
        if(isButtonSelected)
            currentRenderer.material.SetTexture("_MainTex", onTexture);
        else
            currentRenderer.material.SetTexture("_MainTex", offTexture);
    }
}
```

Додаток В. Ілюстративний матеріал до захисту магістерської
кваліфікаційної роботи

ІЛЮСТРАТИВНИЙ МАТЕРІАЛ ДО ЗАХИСТУ МАГІСТЕРСЬКОЇ
КВАЛІФІКАЦІЙНОЇ РОБОТИ

Завідувач кафедри ПЗ, д.т.н., професор _____ О. Н. Романюк

Науковий керівник, к.т.н., доцент _____ Н.П. Бабюк

Рецензент, к.т.н., доцент _____ Я.В. Іванчук

Нормоконтроль, к.т.н., проф.кафедри ПЗ _____ Н.П. Бабюк

Виконавець, студент групи 2ПІ-18м _____ В. О. Білик

Плакат 1 – Титульний лист презентації

ВІННИЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

Білик Владислав Олегович

**Розробка методу інтерактивної візуалізації
тривимірних об'єктів у реальному середовищі**

Науковий керівник
к.т.н., доц. Бабюк Н.П.

Вінниця – 2019

1

Плакат 2 – Мета та завдання досліджень

Мета та завдання досліджень

Мета та завдання дослідження. Основною метою роботи є покращення працездатності та швидкодії інтерактивної візуалізації тривимірних об'єктів у реальному часі та в реальному середовищі за рахунок використання засобів доповненої реальності.

Основними задачами дослідження є:

- провести порівняльний аналіз методологічних, алгоритмічних і програмних засобів;
- покращити метод інтерактивної візуалізації у реальному часі;
- удосконалити метод підвищення реалістичності відображення тривимірних об'єктів при візуалізації у реальному часі;
- розробити метод зміни параметрів об'єктів при візуалізації в реальному часі;
- розробити метод підготовки та зберігання тривимірних моделей для покращення швидкодії реалістичної візуалізації на мобільних пристроях;
- розробити алгоритм маніпуляції віртуальними тривимірними об'єктами в просторі для підвищення інтерактивності візуалізації;
- сформулювати рекомендації з проектування інтерфейсу користувача для візуалізації.

Плакат 3 – Наукова новизна отриманих результатів

Наукова новизна отриманих результатів

1. Подальшого розвитку отримав метод підвищення реалістичності при візуалізації тривимірних об'єктів засобами доповненої реальності, що відрізняється від відомого використанням комбінації статичного та динамічного освітлення, що дозволяє виконувати попередню візуалізацію статичної падаючої тіні об'єктів.
2. Вперше розроблено метод зміни параметрів об'єктів доповненої реальності при візуалізації, особливістю якого є введення параметру матеріалу, що дає можливість змінювати параметри об'єктів в реальному часі та забезпечує спрощення інтерфейсу користувача.

Плакат 4 – Порівняльний аналіз аналогів

Порівняльний аналіз аналогів



Houzz - Home Design & Remodel

IKEA Place



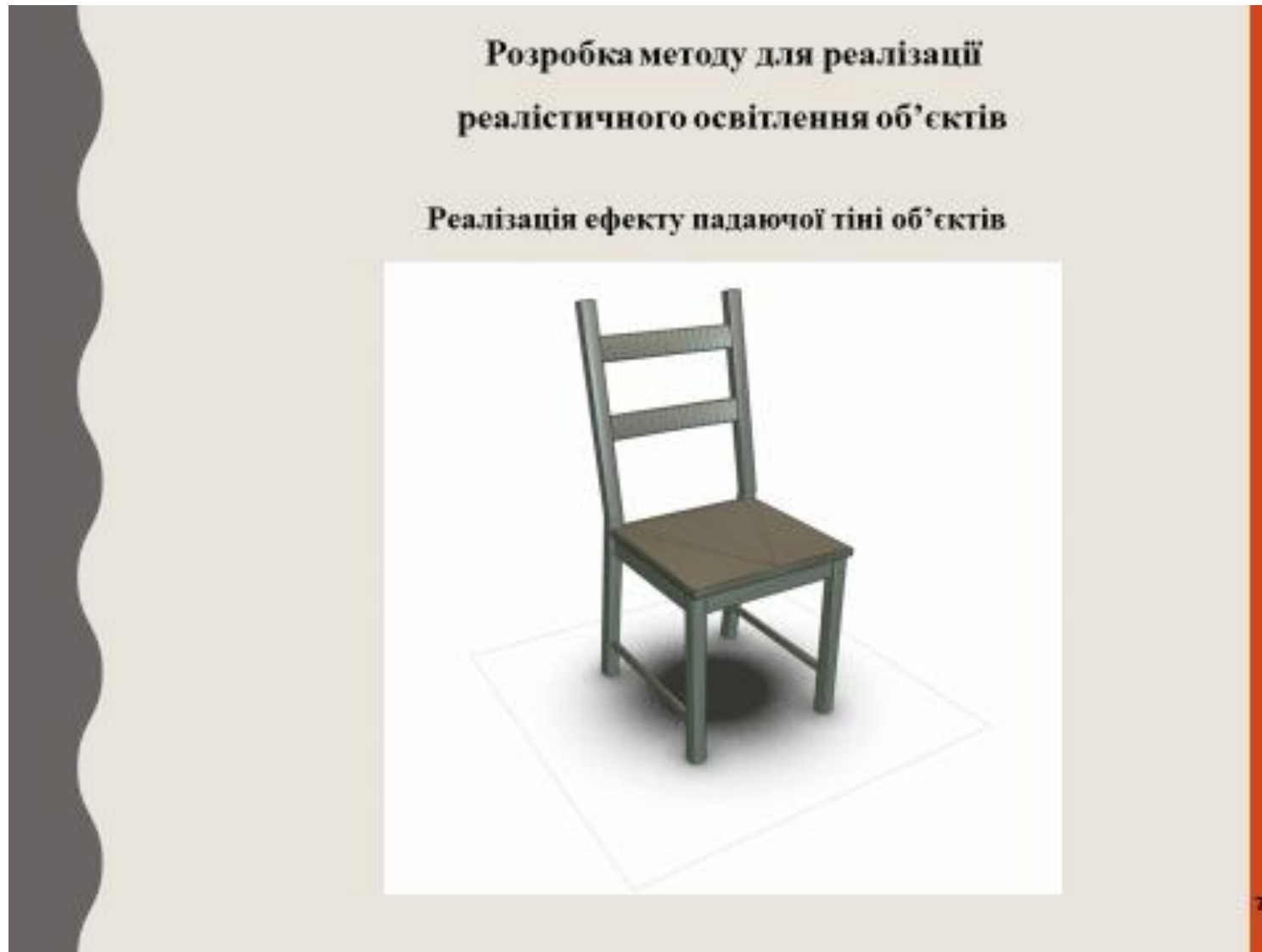
Плакат 5 – Розробка методу для реалізації реалістичного освітлення об'єктів



Плакат 6 – Розробка методу для реалізації реалістичного освітлення об'єктів



Плакат 7 – Розробка методу для реалізації реалістичного освітлення об'єктів



Плакат 8 – Підготовка та зберігання тривимірних моделей об'єктів



Плакат 9 – Алгоритм маніпуляції віртуальними тривимірними об'єктами в просторі з використанням жестових методів введення

Алгоритм маніпуляції віртуальними тривимірними об'єктами в просторі з використанням жестових методів введення

Координати точок об'єкта при повороті, переміщенні і зміні масштабу об'єкта в загальному випадку змінюються у відповідності з наступною формулою:

$$\begin{bmatrix} X_1 \\ Y_1 \\ Z_1 \end{bmatrix} = M \cdot \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \cdot \begin{bmatrix} X_0 \\ Y_0 \\ Z_0 \end{bmatrix} + \begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix},$$

де

X_0, Y_0, Z_0 – початкові координати точки об'єкта,

X_1, Y_1, Z_1 – кінцеві координати точки об'єкта,

M – коефіцієнт зміни масштабу,

матриця $\|a_{mn}\|$ – матриця перетворення повороту, елементи якої виражені через кути повороту довкола трьох просторових осей,

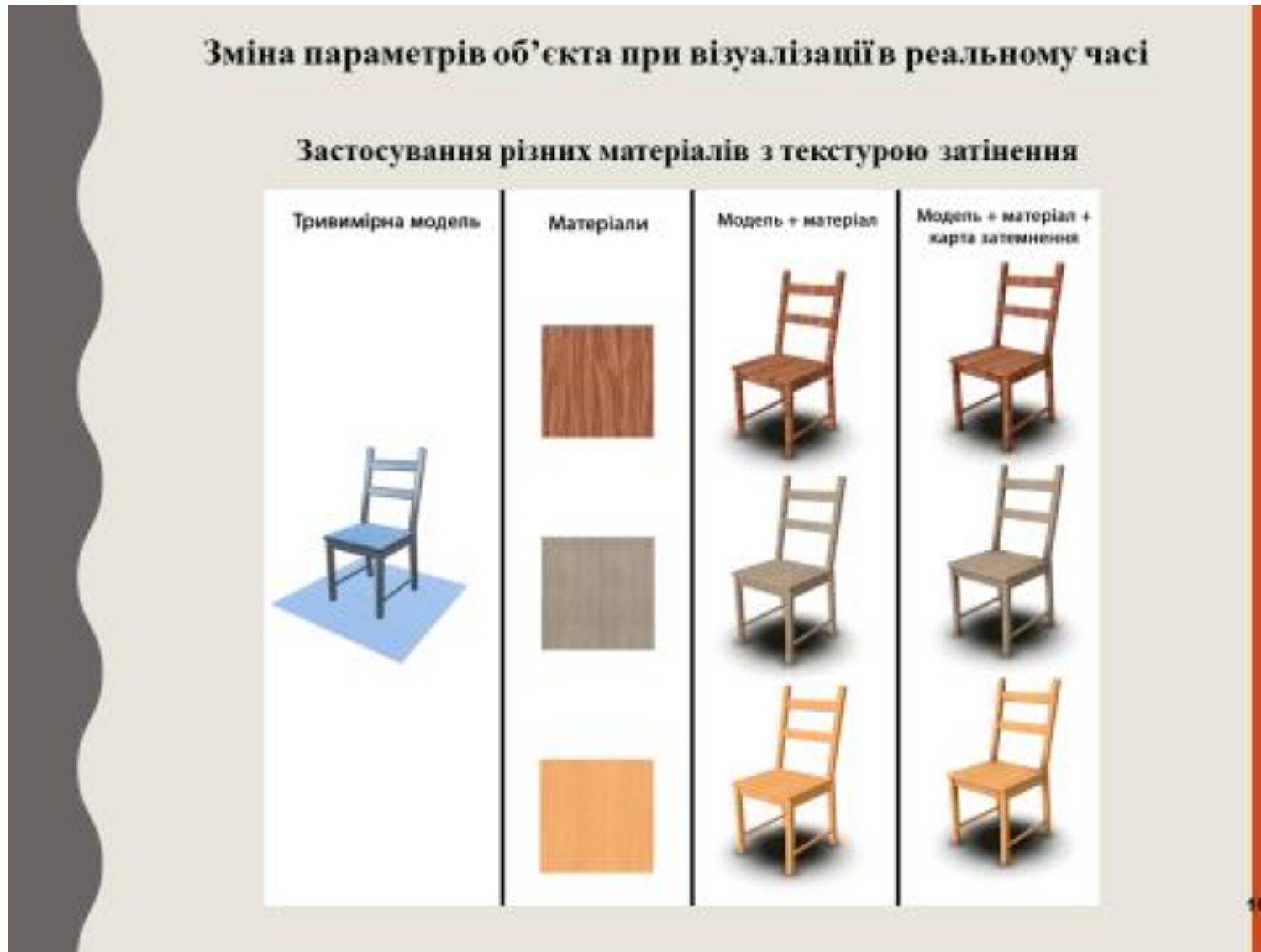
$r = (r_1, r_2, r_3)$ – вектор переміщення об'єкта.

У разі переміщення і обертання об'єкта в площині XZ без зміни масштабу наведена вище формула спрощується: координата Y залишається незмінною, а координати X і Z змінюються відповідно до виразу

$$\begin{bmatrix} Z_1 \\ X_1 \end{bmatrix} = \begin{bmatrix} \cos \varphi & -\sin \varphi \\ \sin \varphi & \cos \varphi \end{bmatrix} \cdot \begin{bmatrix} Z_0 \\ X_0 \end{bmatrix} + \begin{bmatrix} r_3 \\ r_1 \end{bmatrix},$$

де φ – кут повороту об'єкта довкола осі Y

Плакат 10 – Зміна параметрів об'єкта при візуалізації в реальному часі



Плакат 11 – Економічна частина

Економічна частина

Загальні витрати на виконання та впровадження результатів наукової розробки дорівнюють **363 643,36 грн.**

Збільшення чистого прибутку підприємства рівне **2700** тисяч гривень, **4200** тисяч гривень, **4800** тисяч гривень за перший, другий і третій роки після впровадження результатів наукової розробки відповідно.

Абсолютна ефективність вкладених інвестицій рівна **8 665 384** гривень; відносна ефективність вкладених інвестицій рівна **121%**; термін окупності – **0,83** року.

На основі проведених розрахунків зроблено висновок про доцільність наукової розробки.

Плакат 12 – Висновки

Висновки

- Було проаналізовано стан проблеми сучасної візуалізації тривимірних об'єктів.
- Розглянуто основні аналоги, визначено їх особливості та недоліки і розроблено порівняння з власним програмним продуктом.
- Удосконалено метод підвищення реалістичності відображення тривимірних об'єктів при візуалізації в реальному часі.
- Розроблено метод зміни параметрів об'єктів при візуалізації в реальному часі та метод підготовки і зберігання тривимірних моделей для покращення швидкодії реалістичної візуалізації на мобільних пристроях.
- Розроблено алгоритм маніпуляції віртуальними тривимірними об'єктами в просторі для підвищення інтерактивності візуалізації.
- Виконано економічний аналіз та розрахунки, результати яких підтвердили доцільність розробки.

Плакат 13 – Завершальний лист презентації

Дякую за увагу !