

АЛГОРИТМИ ПРОГРАМНОЇ ВЗАЄМОДІЇ З ВІРТУАЛЬНИМИ ОБ'ЄКТАМИ У СИСТЕМАХ ВІРТУАЛЬНОЇ ТА ДОПОВНЕНОЇ РЕАЛЬНОСТІ

Вінницький національний технічний університет

Анотація

У даній роботі досліджено особливості програмної взаємодії користувача з віртуальними об'єктами у системах віртуальної та доповненої реальності. Розглянуто основні алгоритми, що забезпечують коректну обробку дій користувача, зокрема визначення положення та орієнтації, виявлення колізій, використання променевого трасування (raycasting) та обробку жестів і контролерів. Проаналізовано специфіку реалізації таких алгоритмів у реальному часі та їхній вплив на продуктивність і реалістичність VR/AR-додатків. Показано, що правильна програмна організація взаємодії є ключовим фактором створення ефективних і зручних віртуальних середовищ.

Ключові слова: віртуальна реальність, програмна взаємодія, користувацький досвід.

Abstract

This paper investigates the features of software-based user interaction with virtual objects in virtual and augmented reality systems. The main algorithms that ensure correct processing of user actions are considered, including position and orientation tracking, collision detection, the use of raycasting, and the processing of gestures and controllers. The specifics of implementing these algorithms in real time and their impact on the performance and realism of VR/AR applications are analyzed. It is shown that proper software organization of interaction is a key factor in creating effective and user-friendly virtual environments.

Keywords: virtual reality, software interaction, user experience.

Вступ

Системи віртуальної (VR) та доповненої реальності (AR) активно застосовуються у сфері освіти, медицини, інженерії, розваг та тренажерних комплексів. Однією з ключових характеристик таких систем є можливість інтерактивної взаємодії користувача з віртуальними об'єктами у тривимірному просторі. На відміну від традиційних програмних продуктів, VR/AR-додатки вимагають обробки просторових даних у реальному часі, що зумовлює використання спеціалізованих алгоритмів та підходів до програмування.

Актуальність теми полягає в тому, що якість взаємодії безпосередньо впливає на рівень занурення користувача у віртуальне середовище, зручність керування та загальне сприйняття програмного продукту. Некоректна реалізація алгоритмів взаємодії може призводити до затримок, помилкових реакцій системи або втрати відчуття реальності.

Об'єктом дослідження є процес програмної взаємодії користувача з об'єктами у VR/AR-системах.

Предметом дослідження є алгоритми та методи обробки взаємодії у віртуальному та доповненому середовищі.

Основна частина

Програмна взаємодія користувача з віртуальними об'єктами є однією з ключових складових систем віртуальної та доповненої реальності, оскільки саме вона визначає характер і глибину занурення у цифрове середовище. У VR/AR-додатках користувач виступає активним учасником процесу, а не пасивним спостерігачем, що потребує реалізації складних алгоритмів взаємодії, орієнтованих на роботу у тривимірному просторі та режимі реального часу [1].

На відміну від традиційних програмних систем, де взаємодія обмежується двовимірними інтерфейсами, VR/AR-середовища вимагають постійного аналізу просторового положення

користувача. Для цього використовуються алгоритми трекінгу положення та орієнтації, які обробляють дані з камер, гіроскопів, акселерометрів та інших сенсорів [2]. На основі отриманої інформації система визначає координати точки огляду користувача, напрямок його погляду, а також положення рук або контролерів у просторі. Від точності та стабільності цих алгоритмів залежить коректність відображення сцени та узгодженість рухів користувача з віртуальним середовищем.

Одним із базових алгоритмів програмної взаємодії у VR/AR-системах є *raycasting*. Його принцип полягає у випромінюванні віртуального променя з камери користувача або з контролера для визначення об'єкта, з яким відбувається взаємодія [5]. У разі перетину променя з віртуальним об'єктом система реєструє подію та виконує відповідну програмну дію, наприклад вибір, активацію або зміну стану об'єкта. Завдяки своїй універсальності *raycasting* широко застосовується для реалізації меню, кнопок та елементів навігації у тривимірному просторі.

Важливою складовою програмної взаємодії є алгоритми виявлення та обробки колізій, які дозволяють визначати моменти зіткнення між віртуальними об'єктами або між користувачем і елементами середовища [1]. Використання фізичних моделей та колайдерів забезпечує реалістичну поведінку об'єктів, зокрема їх зупинку, відштовхування або падіння. Коректна реалізація колізій запобігає виникненню нелогічних ситуацій, таких як проходження крізь тверді поверхні або некоректне накладання об'єктів один на одного.

Суттєву роль у VR/AR-додатках відіграють алгоритми обробки дій користувача. Програмна система аналізує сигнали з контролерів, рухи рук і положення пальців, інтерпретуючи їх як команди керування [3]. Такий підхід дозволяє створювати інтуїтивні механізми взаємодії, які максимально наближені до реальної фізичної взаємодії з об'єктами. Це особливо важливо для навчальних і тренажерних систем, де точність та зрозумілість взаємодії мають вирішальне значення.

Окрему увагу слід приділити алгоритмам просторової маніпуляції віртуальними об'єктами. До них належать методи переміщення, масштабування та обертання об'єктів у тривимірному просторі. Реалізація таких дій потребує коректної обробки векторів руху, кутів обертання та їх узгодження з фізичною моделлю віртуального середовища [2]. Подібні алгоритми широко застосовуються в інженерних симуляціях, архітектурній візуалізації та освітніх VR/AR-додатках.

Важливим аспектом програмування систем віртуальної та доповненої реальності є забезпечення стабільної роботи алгоритмів у режимі реального часу. Надмірне обчислювальне навантаження або неефективна програмна реалізація можуть призводити до зниження частоти кадрів та погіршення користувацького досвіду [5]. Саме тому розробники приділяють значну увагу оптимізації алгоритмів взаємодії, зокрема зменшенню кількості обчислень та ефективному використанню ресурсів апаратного забезпечення.

Окреме місце в програмуванні VR/AR-систем займають питання синхронізації взаємодії користувача з віртуальним середовищем. Оскільки обробка просторових даних, фізичних взаємодій та візуалізація сцени відбуваються паралельно, важливо забезпечити узгоджену роботу всіх програмних модулів [1]. Порушення синхронізації між вхідними даними та відображенням сцени може призводити до затримок, ривків зображення та зниження рівня комфорту користувача.

Значну увагу при розробці VR/AR-додатків приділяють оптимізації алгоритмів програмної взаємодії. Умови реального часу вимагають стабільної частоти кадрів, тому складні фізичні обчислення та перевірки колізій повинні виконуватися максимально ефективно [5]. Для цього застосовуються спрощені фізичні моделі, зонування простору та обмеження кількості активних об'єктів взаємодії. Подібні підходи дозволяють зменшити навантаження на процесор і графічний адаптер без втрати функціональності.

Важливим аспектом програмної реалізації взаємодії є робота з фізичними рушіями. Фізичні рушії дозволяють автоматизувати обробку колізій, гравітації та руху об'єктів, проте їх використання потребує правильного налаштування параметрів. Невірно підібрані значення маси, сили або швидкості можуть призводити до неприродної поведінки об'єктів та порушення реалістичності віртуального середовища. Тому розробники часто поєднують стандартні можливості фізичних рушіїв із власними алгоритмами керування.

Сучасні VR/AR-системи також активно використовують подієво-орієнтовану модель програмування. У такій моделі взаємодія користувача з об'єктами обробляється у вигляді подій, наприклад дотик, захоплення або відпускання об'єкта [3]. Це дозволяє структурувати програмний код, спростити логіку взаємодії та підвищити масштабованість програмного продукту. Подієвий підхід є особливо ефективним для складних VR-сцен із великою кількістю інтерактивних елементів.

Ще одним важливим напрямом є адаптація алгоритмів взаємодії під різні типи пристроїв. VR/AR-додатки можуть використовуватися на стаціонарних системах, мобільних пристроях або автономних гарнітурах, що мають різні апаратні можливості. Тому алгоритми взаємодії повинні бути достатньо гнучкими, щоб забезпечувати стабільну роботу незалежно від платформи. Це досягається шляхом динамічного налаштування рівня деталізації, складності фізичних обчислень та кількості об'єктів взаємодії.

Окрему увагу слід приділити питанням зворотного зв'язку під час взаємодії з віртуальними об'єктами. Візуальні, звукові та тактильні ефекти дозволяють користувачу краще усвідомлювати результат своїх дій [4]. Програмна реалізація такого зворотного зв'язку потребує синхронізації між алгоритмами взаємодії та системами відтворення ефектів. Коректний зворотний зв'язок підвищує інтуїтивність взаємодії та сприяє глибшому зануренню у віртуальне середовище.

Таким чином, програмна взаємодія у VR/AR-системах є багаторівневим процесом, що включає алгоритми трекінгу, вибору об'єктів, обробки колізій, жестів, оптимізації та синхронізації. Комплексне застосування цих алгоритмів дозволяє створювати ефективні, стабільні та реалістичні віртуальні середовища, орієнтовані на потреби користувача.

Висновок

У роботі розглянуто основні алгоритми програмної взаємодії з віртуальними об'єктами у системах віртуальної та доповненої реальності. Встановлено, що використання алгоритмів трекінгу, raycasting, виявлення колізій і розпізнавання жестів є необхідною умовою створення інтерактивних VR/AR-додатків.

Доведено, що ефективна програмна реалізація взаємодії підвищує реалістичність віртуального середовища та забезпечує комфортну роботу користувача. Подальший розвиток VR/AR-технологій потребує вдосконалення алгоритмів обробки просторових даних та оптимізації їхньої роботи.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Sherman W. R., Craig A. B. Understanding Virtual Reality: Interface, Application, and Design. Boston : Morgan Kaufmann, 2018. 716 p.
2. LaValle S. M. Virtual Reality. Cambridge : Cambridge University Press, 2017. 414 p.
3. Jerald J. The VR Book: Human-Centered Design for Virtual Reality. New York : ACM Books, 2016. 622 p.
4. Pangilinan E., Lukas S., Mohan V. Creating Augmented and Virtual Realities: Theory & Practice for Next-Generation Spatial Computing. New York : CRC Press, 2020. 350 p.
5. Unity Technologies. Unity Manual: Physics, Raycasting and Interaction Systems : електрон. ресурс. Режим доступу: <https://docs.unity.cn/Manual/com.unity.physics.html> (дата звернення: 04.01.2026).

Густяїнська Яна Сергіївна – студентка 5ПІ-22б, факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет. E-mail: husiatynska.yana@gmail.com

Бабюк Наталія Петрівна – к.т.н., доцент кафедри програмного забезпечення, Вінницький національний технічний університет. E-mail: babiuk@vntu.edu.ua

Husiatynska Yana Serhiivna – student of group 5PI-22b, Faculty of Information Technologies and Computer Engineering, Vinnytsia National Technical University. E-mail: husiatynska.yana@gmail.com

Babiuk Natalia Petrivna – PhD, Associate Professor of the Department of Software Engineering, Vinnytsia National Technical University. E-mail: babiuk@vntu.edu.ua