

ПЛАТФОРМНО-АДАПТИВНИЙ МЕТОД КЕРУВАННЯ ІГРОВИМ ПЕРСОНАЖЕМ У СЕРЕДОВИЩІ UNREAL ENGINE 5

Вінницький національний технічний університет

Анотація

В даній публікації розглянуто підходи до створення класичних ретро-відеоігор з танковим управлінням. Розглянуто в яких дослідженнях приділялась увага класифікації ретро-відеоігор та які ігрові механіки використовувались при їх створенні. Запропоновано алгоритм платформно-адаптивного методу керування ігровим персонажем у середовищі Unreal Engine 5, який забезпечує мультимодальне управління на різних платформах: персональний комп'ютер, мобільні пристрої з тачскріном, а також ігрові консолі.

Ключові слова: відеоігри, кросплатформна розробка, ігровий AI, Unreal Engine 5

Abstract

Different approaches for creating classic retro video games with tank controls reviewed in this paper. Discovered which studies have focused on the classification of retro video games and what game mechanics were used in their development. An algorithm for a platform-adaptive method of controlling a game character in the Unreal Engine 5 environment is proposed, providing multimodal control across different platforms: personal computers, touchscreen mobile devices, and gaming consoles.

Keywords: video games, cross-platform development, in-game AI, Unreal Engine 5

Вступ

Жанр ретро-відеоігор з танковим управлінням нечасто використовується у сучасних іграх. Тому у сучасних дослідженнях їм практично не приділяється увага. Є кілька дещо давніших досліджень де надається увага механікам, задіяним у подібних іграх. Стаття [1] розглядає структуру моделі MDA (механіка, динаміка, естетика) для аналізу систем гейміфікації. Механіка представляє основні компоненти, такі як бали, значки та рівні, а динаміка описує, як ці елементи взаємодіють під час гри. Естетика зосереджується на емоційних переживаннях, які вона створює (наприклад, виклик, занурення в розповідь). Цей фреймворк допомагає дизайнерам більш детально проектувати ігри. У іншій праці [2] досліджуються модульні коригування ігрових систем, в яких приділяється увага процесу людино-машинній взаємодії у тогочасних відеоіграх. Стаття [3] аналізує, як дизайн контролера впливає на реалізацію ігрової механіки. Порівнюються такі пристрої введення, як джойстики та клавіатури, досліджуючи, як обмеження апаратного забезпечення формують системи руху та парадигми взаємодії – особливо актуально для того, щоб елементи танкового керування мали ефективний зв'язок з пристроями вводу. У статті [4] наголошується на основній ігровій механіці як на фундаменті якості, стверджуючи, що успішні ігри збалансовують механіку зі здібностями гравців і системами прогресу. У статті наголошується на ітеративному вдосконаленні механічних взаємодій. Ці механізми відображають характерні для епохи компроміси в дизайні, де технічні обмеження (апаратне забезпечення) і бажана естетика жанру хоррор безпосередньо формували схеми керування, які зараз вважаються ретро.

До прикладу розглядаючи найпопулярніші відеоігри кінця 90-х років і початку 2000-х, такі як Silent Hill (1999), Silent Hill 2 (2001), Dino Crisis (1999), та оригінальні серії ігор Resident Evil, які до цих пір вважаються культовими і для багатьох гравців ці ігри символізують ту епоху, коли відеоігри були складнішими та менш поступливими. І навіть через багато років після їх випуску, багато гравців які грали в них у підлітковому віці, часто продовжують грати у них далі через ностальгію.

Це пов'язано з кількома ключовими аспектами:

- атмосфера напруги завдяки управлінню, де кожне неправильне повертання персонажа або спроба швидко втекти від ворогів ставало питанням виживання;
- ретельне планування кожного кроку, від якого могло залежати життя головного героя;
- мінімалістичний, але яскравий дизайн, на якому був створений акцент задля подачі атмосфери історії локацій та персонажів.

В цілому, ці ігри стали культовими завдяки тому, що вони не просто мали незручне управління, а використовували ці механізми, щоб посилити емоційний відгук та переживання гравців коли від цього залежало життя головного героя, і час який минув з останнього збереження. Іншою популярною грою, яка мала нестандартне керування, але при цьому користувалась популярністю була The Witcher (2007). У грі було реалізовано кілька режимів управління рухом персонажу: клавіатура (WASD) та миша (персонаж прослідує за вказаною курсором миші локацією). Але відповідний режим можна було обрати лише на початку гри – в режимі реального часу це неможливо було зробити.

Реалізація методу

У даній публікації пропонується створити новий компонент контролера, який матиме два режими управління: клавіатура (або геймпад) чи дотик тачскріну (або клік мишки). Режими мають перемикатись між собою (рис. 1) щоб дати гравцеві найбільше можливостей для керування персонажем на різних платформах. Це особливо корисно коли у стандартних платформах таких як ПК, мобільні пристрої та консолі з'являються інші, початково не характерні для них методи вводу. Наприклад ноутбуки як правило мають такі самі методи вводу як у ПК, але деякі моделі мають сенсорний екран. Також до планшетів, які є мобільними пристроями може бути приєднана клавіатура. Геймпад в свою чергу може бути приєднаний до будь-яких з вищеперелічених платформ. Тому у відеоіграх існує сенс забезпечення перемикачності між різними методами вводу в режимі реального часу.

Для зручності прототипування метод керування реалізовано у вигляді blueprint-класу контролера (рис. 1). Алгоритм методу передбачає роботу у двох режимах керування персонажем. Таким чином при натисканні клавіш WASD (або рух джойстика переміщення) змушує персонажа рухатись у заданому напрямку.

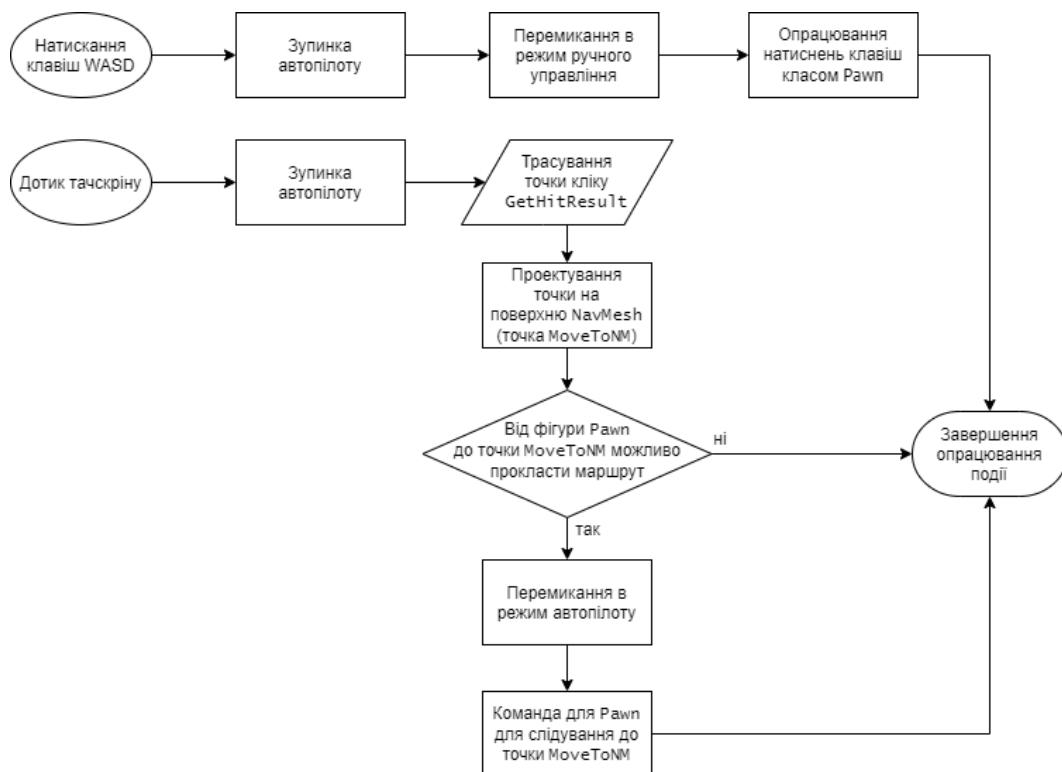


Рисунок 1 - Алгоритм платформно-адаптивного методу керування ігровим персонажем

Однак натиснення на певну точку тачскріну призведе що контролер дасть персонажу команду прослідувати до заданої точки. В класі це названо режимом "автопілоту". Для уникнення конфлікту обох режимів управління при кожній новій команді гравця з будь-якого пристрою вводу діючі накази, які були дані раніше, скасовуються. При дотику тачскріну важливо виконати проектування трасування точки кліку на відповідну точку у тривимірному просторі. Втім гравець може вказати на точку куди ігрова фігура або персонаж не зможуть переміститись (наприклад стіна). Тому необхідно виконати ще одне проектування – на поверхню карти шляхів. Однак між фігурою гравця і знайденою точкою може існувати розрив карти шляхів. Таке може бути якщо між цими двома локаціями існує постійна або тимчасова перешкода (наприклад зруйнована підлога через яку неможливо перескочити). Тому має бути перевірка можливості прокладення маршруту.

В поєднанні з фіксованими камерами, які змінюють ракурс залежно від місця розташування персонажу, запропонована система управління забезпечує адаптивність і зручність. Завдяки автопілоту можна легко переміщатися простором просто натиснувши на потрібне місце локації, що є особливо корисним для гравців на мобільних пристроях. І водночас, ручне керування надає точний контроль, необхідний у напружених ситуаціях, що можуть виникнути через несподівану небезпеку, яку не завжди видно завдяки фіксованій камері. Динамічне перемикання між режимами допомагає уникнути плутанини у керуванні та дозволяє гравцю адаптуватися згідно ситуації: використовувати автопілот для спокійного дослідження, а ручне керування – для бою або несподіваних загроз. Отже, поєднання двох режимів керування з фіксованою камерою забезпечує адаптивність та зручність, а також атмосферу напруги завдяки обмеженому огляду, який підсилює відчуття ризику небезпеки.

Висновки

Згідно з проведеним дослідженням встановлено, що реалізований метод управління з можливістю перемикання між різними режимами вводу (геймпад, клавіатура, мишка та тачскрін) значно покращує процес гри. Автопілот забезпечує комфортне управління при дослідженні ігрового простору, а ручне керування надає можливість точного контролю над персонажем. Таке покращення значно полегшує процес гри з танковим управлінням, де контроль персонажа часто ставить гравця в складні або несподівані умови, особливо коли камера обмежує огляд або змінюється на кожному кроці. У подальшому планується вдосконалити даний метод реалізувавши його у вигляді класу C++ та додавши інтерполяцію для пом'якшення плавності рухів персонажу [5].

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Kim B. Game mechanics, dynamics, and aesthetics // Library Technology Reports. – 2015. – Vol. 51, No. 2. – P. 17–19. URL: <https://journals.ala.org/index.php/ltr/article/view/5630>
2. Van Rozen R., Dormans J. Adapting game mechanics with micro-machinations // Foundations of Digital Games. – Apr. 2014. – Aboard Royal Caribbean Liberty of the Seas, sailing from Fort Lauderdale, Florida, United States.
3. Nævdal L. A comparative analysis of the relationship between gaming controllers and game mechanics: Master's thesis / The University of Bergen. – 2013. URL: <https://bora.uib.no/bora-xmlui/bitstream/handle/1956/7042/109090857.pdf?sequence=1&isAllowed=y>
4. Fabricatore C. Gameplay and game mechanics design: a key to quality in videogames. – 2007. – DOI: 10.13140/RG.2.1.1125.4167.
5. Sashko Peremot. Математика в геймдеві: що таке інтерполяція та як її використовують при розробці ігор [Електронний ресурс]. URL: <https://gamedev.dou.ua/articles/mathematics-gamedev-interpolation/> (дата звернення: 07.03.2025).

Поліщук Альона Ігорівна – студентка групи 4КН-21б, факультет Інтелектуальних інформаційних технологій та автоматизації, Вінницький національний технічний університет

Малініч Ілля Павлович – асистент кафедри Комп'ютерних наук, Вінницький національний технічний університет, e-mail: malinich@vntu.edu.ua

Alona Polishchuk – student of Department of Intelligent Information Technologies and Automation, Vinnytsia National Technical University, Vinnytsia.

Illia Malinich – Assistant Lecturer, Department of Computer Science, Vinnytsia National Technical University, e-mail: malinich@vntu.edu.ua