

АНАЛІЗ СИСТЕМ ПРОЦЕДУРНОЇ АНІМАЦІЇ В 2D-ПЛАТФОРМЕРАХ

Вінницький національний технічний університет

Анотація

У роботі досліджено етапи побудови систем процедурної анімації для 2D-платформерів на базі ігрового рушія Unity. Основну увагу приділено механізмам динамічної адаптації персонажа до складного рельєфу за допомогою алгоритмів інверсної кінематики (IK) та технології Raycasting. Розглянуто концепцію функціональних кінцівок, які поєднують візуальну плавність із активною фізичною взаємодією, що дозволяє персонажу переступати через переешкоди та адаптувати опорні точки в реальному часі.

Ключові слова: *процедурна анімація, Inverse Kinematics (IK), Raycasting, 2D-платформер, пряма кінематика, Unity.*

Abstract

The paper investigates stages to building procedural animation systems for 2D platformers based on the Unity game engine. The main focus is on the mechanisms of dynamic character adaptation to complex terrain using inverse kinematics (IK) algorithms and Raycasting technology. The concept of functional limbs is considered, combining visual fluidity with active physical interaction, allowing the character to step over obstacles and adapt anchor points in real time.

Keywords: *procedural animation, Inverse Kinematics (IK), Raycasting, 2D platformer, forward kinematics, Unity.*

Вступ

У сучасній індустрії розробки відеоігор оптимізація ресурсів та реалізм візуалізації є пріоритетними завданнями. Традиційна покадрова (спрайтова) анімація вимагає значних витрат пам'яті для зберігання великої кількості атласів та зображень усіх можливих станів персонажа. Використання процедурної анімації дозволяє обчислювати рухи безпосередньо під час роботи гри, що суттєво зменшує обсяг графічних даних [1]. Крім того, процедурний підхід забезпечує більш реалістичну реакцію персонажа на зміну геометрії середовища та спрощує масштабування проєкту: створену систему легко адаптувати до нових персонажів, що значно знижує загальну трудомісткість розробки.

Результати дослідження

Системи процедурної анімації в 2D-платформерах складаються з декількох взаємопов'язаних елементів, кожен з яких відповідає за певну частину фізичної взаємодії персонажа зі світом. Аналіз цих компонентів дозволяє зрозуміти механіку формування складних рухів без використання готових кадрів.

Основним елементом аналізу середовища є Raycast-сенсори. Вони працюють за принципом проєктування променів від персонажа до підлоги. В іграх ці промені виконують роль детекторів: вони повертають координати точок зіткнення, нормалі поверхонь та ідентифікатори об'єктів. Завдяки цьому система отримує інформацію про геометрію платформи, нахил схилів та наявність перешкод, що є необхідною умовою для правильної роботи інших модулів [2].

Ключовим обчислювальним елементом є алгоритм інверсної кінематики (IK). У традиційній анімації поворот плеча викликає рух долоні (пряма кінематика). Інверсна кінематика працює навпаки: система

отримує цільову точку для стопи, а пакет 2D ІК автоматично вираховує необхідні кути згинання суглобів. Це забезпечує точну фіксацію кінцівок на нерівному рельєфі, запобігаючи візуальним дефектам проходження ноги крізь текстури [3].

Наступним важливим елементом є контролер кроків. Він відповідає за координацію ніг під час руху. Замість циклічного відтворення кадрів ходьби, контролер порівнює поточне положення опорної ноги з вектором швидкості. Коли відстань перевищує заданий поріг, ініціюється крок – процедурне переміщення цільової точки ІК до нової позиції за допомогою спеціальних контролерів або математичних функцій (наприклад, синусоїди), що створює ефект м'якого кроку [4].

Окрему роль відіграє функціональна інтеграція процедурної анімації з фізичними властивостями персонажа. Кінцівки в такій системі виступають не лише візуальними елементами, а й активними фізичними об'єктами [5]. Інтеграція з фізичним рушієм дозволяє персонажу використовувати ноги для переступання через невеликі перешкоди або підтягування на уступи, якщо цільова точка ІК зміщується на вертикальну поверхню.

Описані етапи процедурної анімації можуть бути реалізовані за допомогою скриптової системи ігрового рушія Unity. Для реалізації можна використати скрипти типу MonoBehaviour, які мають шикокій життєвий цикл, зокрема можливість виконувати операції кожен кадр під час гри, що дає можливість здійснювати перевірку кінцівок персонажа для плавної та вчасної адаптації їх положення. Рушію надає бібліотека для математичних обчислень, зокрема роботи з векторною математикою та обчислення поворотів об'єктів. Також Unity має вбудований фізичний рушій, який підтримує колізії, гравітацію, та дію сил на фізичні тіла, що дозволяє сфокусуватися саме на реалізації елементів процедурної анімації.

Таким чином, перехід від статичної до процедурної моделі дозволяє замінити сотні графічних файлів компактними математичними алгоритмами, які гарантують адаптивність та реалістичність у будь-яких ігрових ситуаціях.

Висновки

Процедурна анімація є ефективним інструментом підвищення інтерактивності та візуальної якості в 2D-іграх, оскільки забезпечує динамічну адаптацію рухів до складного геометрії середовища. Використання інверсної кінематики дозволяє усунути артефакти візуального розриву між персонажем та ігровою поверхнею, роблячи поведінку об'єкта природною для сприйняття. Завдяки автоматизації розрахунку поз у реальному часі досягається значна економія пам'яті за рахунок відмови від статичних графічних атласів. Отже, процедурний підхід є важливою складовою сучасної ігрової розробки та необхідною умовою для створення глибокого ефекту занурення в інтерактивне середовище.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. An Introduction to Procedural Animations [Електронний ресурс] – Режим доступу: <https://www.alanzucconi.com/2017/04/17/procedural-animations/>
2. Unity Technologies. Physics2D.Raycast [Електронний ресурс] – Режим доступу: <https://docs.unity3d.com/ScriptReference/Physics2D.Raycast.html>
3. Inverse Kinematics in 2D [Електронний ресурс] – Режим доступу: <https://www.alanzucconi.com/2018/05/02/ik-2d-1/>
4. Procedural Animation [Електронний ресурс] – Режим доступу: <https://weaverdev.io/projects/proc-animation-tutorial>
5. What Is Physically-Based Animation? [Електронний ресурс] – Режим доступу: <https://medium.com/data-science/what-is-physically-based-animation-cd92a7f8d6a4>

Науковий керівник: **Романюк Оксана Володимирівна** – доцент кафедри програмного забезпечення, Вінницький національний технічний університет, Вінниця, e-mail: romaniukoksanav@gmail.com

Іванчук В'ячеслав Анатолійович – студент групи 6ПІ-22б, факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, Вінниця, e-mail: ivancukvaceslav5@gmail.com

Supervisor: **Romaniuk Oksana Volodymyrivna** – Associate Professor of the Software Department, Vinnytsia National Technical University, Vinnytsia, e-mail: ivancukvaceslav5@gmail.com

Ivanchuk Viacheslav Anatoliiovych – student of group 6PI-22b, Faculty of Information Technology and Computer Engineering, Vinnytsia National Technical University, Vinnytsia, e-mail: romaniukoksnav@gmail.com