

ПРИКЛАДНЕ ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ ДИНАМІЧНОГО ОСВІТЛЕННЯ СЦЕН У РЕАЛЬНОМУ ЧАСІ

Вінницький національний технічний університет

Анотація

Спроектовано прикладне програмне забезпечення для моделювання динамічного освітлення у двовимірному просторі з використанням мови С#. Результатом дослідження є система, що забезпечує реалістичну візуалізацію тіней та світлових плям у реальному часі для ігрових та симуляційних проєктів.

Ключові слова: динамічне освітлення, С#, 2D-графіка, алгоритми візуалізації, реальний час.

Abstract

Application software for modeling dynamic lighting in two-dimensional space using the C# language has been designed. The result of the research is a system that provides realistic visualization of shadows and light spots in real time for gaming and simulation projects.

Keywords: dynamic lighting, C#, 2D graphics, visualization algorithms, real-time.

Вступ

Візуалізація світла та тіні є ключовим аспектом створення занурювального середовища у сучасних графічних застосунках. Традиційні методи статичного записання світла не підходять для інтерактивних сцен, де джерела світла або перешкоди постійно змінюють своє положення [1-3]. Дана робота пропонує програмне рішення для обчислення освітлення «видом зверху», що дозволяє досягти балансу між візуальною якістю та продуктивністю на пристроях з середніми технічними характеристиками.

Основний розділ

У межах дослідження було розроблено архітектуру прикладного програмного забезпечення для симуляції динамічного освітлення у 2D-просторі, реалізовану мовою С#. Основна ідея розробки полягає у створенні системи, яка здатна в реальному часі обчислювати зони видимості та затінення, базуючись на взаємодії джерел світла з геометрією об'єктів сцени.

Центральним елементом системи є обчислювальний алгоритм Ray Casting (трасування променів). Процес формування освітленої сцени розділено на кілька етапів:

1. **Аналіз геометрії:** Програма сканує сцену на наявність об'єктів-перешкод, які представлені у вигляді наборів вершин та ребер. Для оптимізації обчислень використовуються лише ті об'єкти, що потрапляють у радіус дії активного джерела світла.
2. **Побудова полігону видимості:** Від кожної точки джерела світла випускаються вектори (промені) у напрямку кожної вершини перешкоди. Крім того, випускаються додаткові промені з невеликим відхиленням (± 0.0001 радіан), що дозволяє точно визначити точки "зрізу" світла за перешкодами. Схематичне відображення цього процесу та принцип формування тіней наведено на рисунку 1.

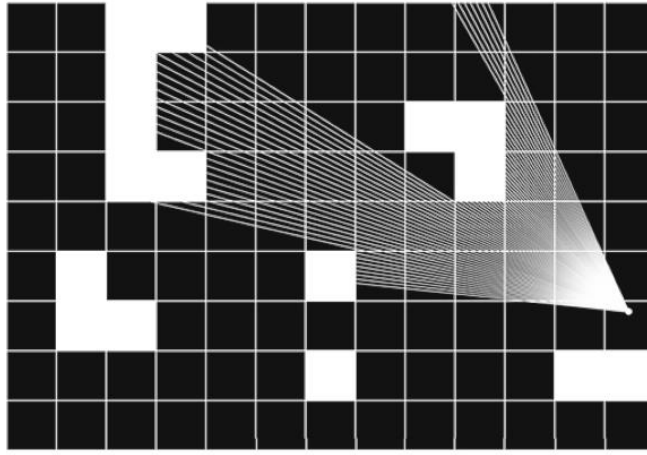


Рисунок 1 – Схема побудови полігону освітлення через трасування променів

3. **Генерація маски світла:** Отримані точки перетину променів з об'єктами або межами сцени сортуються за кутом, утворюючи замкнений полігон. Цей полігон заповнюється градієнтною заливкою, де інтенсивність кольору згасає від центру до периферії, імітуючи фізичне розсіювання світла.
4. **Композиція сцени:** Для реалізації ефекту декількох джерел світла використовується метод адитивного змішування кольорів. Кожне джерело світла рендериться в окрему текстуру (буфер), які потім накладаються на основне зображення сцени. Це дозволяє досягти ефекту перекриття тіней та змішування різних кольорів освітлення, як показано у результатах роботи програми на рисунку 2.

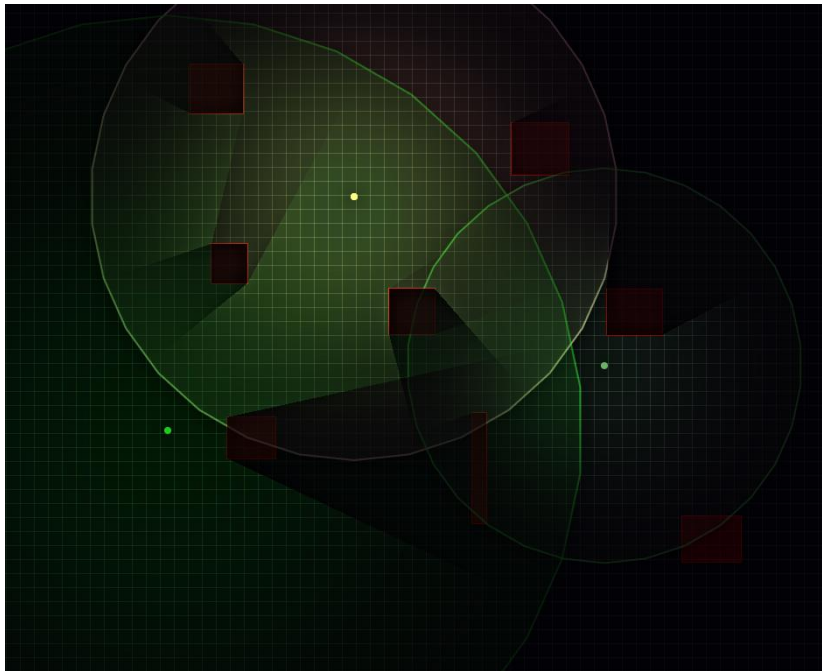


Рисунок 2 – Приклад роботи системи з декількома джерелами світла та перешкодами складної форми

Технічна реалізація на мові C# дозволяє ефективно керувати пам'яттю та об'єктами сцени завдяки використанню високорівневих колекцій та інтеграції з графічними бібліотеками. Завдяки оптимізації алгоритму сортування вершин, система стабільно працює при частоті кадрів понад 60 FPS навіть при значній кількості динамічних об'єктів.

Висновки

Розроблено програмне забезпечення мовою С# для динамічного освітлення сцен у реальному часі. Дана розробка може бути інтегрована у навчальні симулятори, інді-ігри або системи візуалізації планів приміщень. Подальші дослідження будуть спрямовані на впровадження шейдерів для обробки напівтіней та віддзеркалень.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Real-Time Rendering / Tomas Akenine-Möller, Eric Haines, Naty Hoffman — 2018. <https://www.routledge.com/Real-Time-Rendering-Fourth-Edition/Akenine-Moller-Haines-Hoffman-Pesce-Iwanicki-Hillaire/p/book/9781138627000>
2. Learning C# by Developing Games with Unity / Harrison Ferrone — 2020. <https://www.packtpub.com/product/learning-c-by-developing-games-with-unity-2020-fifth-edition/9781800207806>
3. Документація Microsoft .NET [Електронний ресурс]. <https://learn.microsoft.com/en-us/dotnet/>

Коваль Сергій Олександрович — студент групи ІКІ-22б, факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, Вінниця, e-mail: Kovalsergij.zt@gmail.com

Науковий керівник: **Черняк Олександр Іванович** – кандидат технічних наук, доцент кафедри обчислювальної техніки Вінницького національного технічного університету, Вінниця, e-mail: chernyak@vntu.edu.ua

Koval Serhii Oleksandrovych — Faculty of Information Technology and Computer Engineering, Vinnytsia National Technical University, Vinnytsia, e-mail: Kovalsergij.zt@gmail.com

Scientific supervisor: **Chernyak Oleksandr Ivanovych** — PhD, associate professor of the Department of Computer Engineering, Vinnytsia National Technical University, Vinnytsia, e-mail: chernyak@vntu.edu.ua