

ВИКОРИСТАННЯ SCENEKIT ДЛЯ ПРОЦЕДУРНОГО МОДЕЛЮВАННЯ РЕЛЬЄФУ

Харківський національний університет радіоелектроніки

Анотація

SceneKit є потужним фреймворком для створення тривимірної графіки в екосистемі Apple. У роботі розглядається використання SceneKit для рендерингу процедурно згенерованих ландшафтів у реальному часі. Особлива увага приділяється методам генерації рельєфу, таким як фрактальні алгоритми, клітинні автомати, багаторівневе згладжування. Також розглянуто оптимізацію продуктивності за допомогою адаптивної деталізації та шейдерних технологій.

Ключові слова: SceneKit, процедурна генерація, тривимірна графіка, оптимізація продуктивності, алгоритми генерації.

Abstract

SceneKit is a versatile framework for three-dimensional graphics within Apple's ecosystem. This paper explores the application of SceneKit for rendering procedural terrain in real time. Various procedural generation techniques, including fractal-based methods, midpoint displacement, cellular automata, and multi-resolution meshes, are analyzed in the context of their efficiency and realism. Special attention is given to adaptive mesh refinement, shader-based surface modeling, and computational performance tuning. The study examines practical implementations for mobile and desktop devices, evaluating the trade-offs between visual fidelity and processing speed in interactive applications.

Keywords: SceneKit, procedural generation, three-dimensional graphics, performance optimization, generation algorithms.

Вступ

Процедурне моделювання рельєфу є ключовою складовою у створенні динамічних цифрових середовищ, що широко застосовуються у відеоіграх, географічних інформаційних системах, архітектурних візуалізаціях та навчальних симуляціях. Основна перевага такого підходу – можливість автоматичної генерації складних та реалістичних ландшафтів без потреби у значних обсягах пам'яті для збереження попередньо підготовлених моделей.

Результати дослідження

Одним із ефективних методів генерації рельєфу є розбиття висотної карти на багаторівневі сітки, що дозволяє адаптивно керувати деталізацією поверхонь.

Формування висотного профілю може бути описане через дискретизацію висотної функції:

$$H(x, y) = H(x-1, y) + H(x, y-1) - H(x-1, y-1) + R \quad (1)$$

де $H(x, y)$ - висота у вузлі сітки, R - випадкова змінна, що визначає локальні нерівності. Цей підхід використовується у геологічних дослідженнях для прогнозування змін рельєфу внаслідок ерозії та сейсмічної активності.

Метод середньозваженого згладжування дозволяє отримати більш плавні контури ландшафту, що визначається рівнянням:

$$H'(x, y) = \frac{1}{N} \sum_{i=1}^N H(x_i, y_i) \quad (2)$$

де N - кількість сусідніх точок, $H(x_i, y_i)$ - висоти сусідніх вузлів. Даний алгоритм часто застосовується у візуалізаціях містобудування для моделювання поверхонь доріг та рівнинних територій.

Алгоритм клітинного автомата для моделювання змін рельєфу базується на повторюваних правилах модифікації висотних значень залежно від сусідніх клітин:

$$H_{new}(x, y) = H(x, y) + \sum_{i=1}^M \omega_i \cdot \Delta H_i \quad (3)$$

де ω_i - вагові коефіцієнти впливу сусідів, ΔH_i - зміни висот. Цей метод активно використовується у симуляторах природних катаклізмів, де потрібно імітувати процеси зсувів ґрунту чи впливу вітрової ерозії.

Адаптивне розділення сітки з використанням багаторівневої апроксимації забезпечує ефективну деталізацію складних форм поверхні:

$$L(x, y) = \frac{1}{2}(L(x-1, y) + L(x, y-1)) + R \quad (4)$$

де $L(x, y)$ - локальна деталізація, R - випадкове коригування. Такий підхід застосовується у VR-симуляціях для оптимізації візуальної якості сцен без надмірного навантаження на графічний процесор.

На графіку представлена залежність продуктивності рендерингу від рівня складності процедурного рельєфу. Він демонструє, що при зростанні складності генерації кількість кадрів за секунду (FPS) зменшується, що вказує на необхідність застосування оптимізаційних алгоритмів для збереження продуктивності.

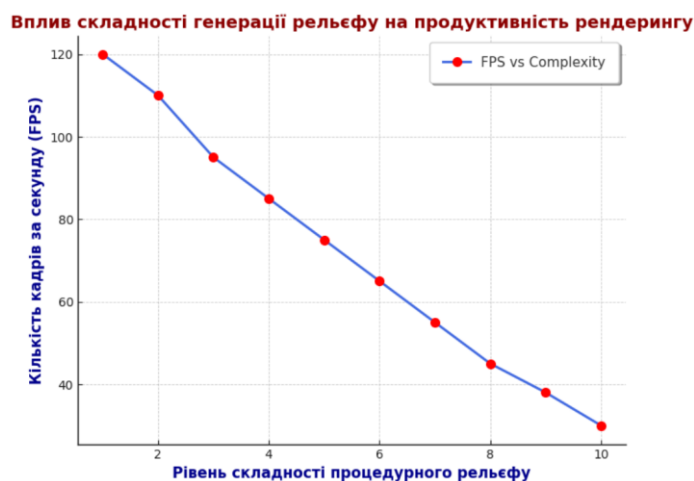


Рис. 1 Вплив складності генерації рельєфу на продуктивність рендерингу

Структурна схема демонструє процес процедурної генерації ландшафтів у SceneKit. Вона містить ключові компоненти: SceneKit як базову платформу, алгоритми процедурної генерації (фрактальні методи, клітинні автомати, багаторівневі сітки), обробку текстур, адаптивну деталізацію та оптимізацію продуктивності. Стрілки вказують на послідовність обробки даних, що дозволяє зрозуміти взаємодію між складовими системи.



Рис. 2 Структурна схема процедурної генерації ландшафтів у SceneKit

Таблиця порівняння основних методів процедурної генерації рельєфу

Метод	Особливості	Використання
Метод середньої точки	Поступове розбиття на дрібніші області	Гірські ландшафти, текстури
Клітинні автомати	Динамічні зміни рельєфу	Моделювання ерозії, симулятори
Багаторівневі сітки	Адаптивне керування деталізацією	Високопродуктивні додатки
Середньозважене згладжування	Генерація плавних контурів рельєфу	Геологічні симуляції, архітектура

Таблиця містить порівняння основних методів процедурної генерації рельєфу. Кожен метод має свої переваги та обмеження, які визначають сфери його застосування.

Висновки

У сучасному світі SceneKit є потужним інструментом для процедурної генерації ландшафтів, що дозволяє ефективно балансувати між продуктивністю та якістю візуалізації. Використання адаптивних алгоритмів та оптимізаційних методів робить можливим створення реалістичних сцен навіть на мобільних пристроях. Проаналізовані методи дозволяють знайти оптимальний баланс між швидкістю обчислень та рівнем деталізації рельєфу.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Apple Inc. SceneKit Documentation. [<https://developer.apple.com/documentation/scenekit>]
2. Ebert, D. S., Musgrave, F. K., Peachey, D., Perlin, K., & Worley, S. (2002). Texturing & Modeling: A Procedural Approach. Morgan Kaufmann. [<https://shop.elsevier.com/books/texturing-and-modeling/ebert/978-1-55860-848-1>]
3. Luebke, D. (2003). Level of Detail for 3D Graphics. Elsevier. [<https://shop.elsevier.com/books/level-of-detail-for-3d-graphics/luebke/978-1-55860-838-2>]
4. Prusinkiewicz, P., & Lindenmayer, A. (1996). The Algorithmic Beauty of Plants. Springer-Verlag. [<https://link.springer.com/book/10.1007/978-1-4613-8476-2>]
5. Smelik, R. M., Tutenel, T., Bidarra, R., & Benes, B. (2014). Procedural generation of virtual worlds. Computer Graphics and Applications. [<https://ieeexplore.ieee.org/document/6875895>]

Стрюк Данило Сергійович – студент групи ІТІНФ-21-2, факультет інформаційно-аналітичних технологій та менеджменту, Харківський національний університет радіоелектроніки, Харків, danylo.striuk@nure.ua

Кобилін Ілля Олегович – PHD, асистент кафедри Інформатики, Харківський національний університет радіоелектроніки, м. Харків, ilya.kobylin@nure.ua

Struyk Danylo S. - Faculty of Information and Analytical Technologies and Management, Kharkiv National University of Radio Electronics, Kharkiv, email: danylo.striuk@nure.ua

Kobylin Ilya O. - PHD, Assistant Professor, Department of Informatics, Kharkiv National University of Radio Electronics, Kharkiv, email: ilya.kobylin@nure.ua