

**ОПТИМІЗАЦІЯ 3D-МОДЕЛЕЙ
ДЛЯ ВИКОРИСТАННЯ У КОМП'ЮТЕРНИХ ІГРАХ
ТА ІНТЕРАКТИВНИХ СЕРЕДОВИЩАХ**

Саєнко Є.А., Конвісар А.А., Ільченко Д.В.

e-mail: yevheniia.saienko@nure.ua

Науковий керівник – стар. викл., Колісник В.І.

Харківський національний університет радіоелектроніки, каф. МІРЕС
м. Харків, Україна

The article discusses the optimization of 3D models for computer games and interactive environments. The main optimization methods including reducing the number of polygons, retopology, and texture optimization are described. Modern approaches that improve performance in game engines and interactive applications are described. The advantages and challenges of optimizing 3D assets for real-time graphics are also discussed. The results show that effective optimization methods allow developers to achieve a balance between high visual quality and stable system performance in interactive digital environments.

У світі комп'ютерної графіки 3D-моделювання відіграє ключову роль у створенні віртуальних середовищ, персонажів і об'єктів. Особливо важливим це стає у сфері комп'ютерних ігор та інтерактивних платформ. Тут якість та оптимізація моделей напряму впливають на продуктивність систем і комфорт користувачів. Завдяки швидкому розвитку ігрових рушіїв та графічних технологій вимоги до деталізації продовжують зростати, проте надмірна складність геометрії може негативно позначитися на швидкодії програмного забезпечення, що робить оптимізацію 3D-моделей ключовим етапом створення цифрового контенту.

Процес оптимізації 3D-моделей полягає у зменшенні кількості полігонів, текстурних ресурсів та інших графічних даних, при цьому зберігаючи високий рівень візуальної якості об'єкта. Його мета – забезпечення стабільної роботи програми, підтримання високої частоти кадрів (FPS) і зниження навантаження на графічний процесор та оперативну пам'ять.

Кількість полігонів у моделі – основний фактор, що впливає на її продуктивність. Полігональна сітка визначає геометричну складність об'єкта та безпосередньо впливає на обсяг обчислень, які повинен виконати графічний процесор під час рендерингу сцени. Чим більше полігонів використовується для побудови моделі, тим точніше передаються дрібні деталі, вигини поверхні та загальна форма об'єкта. Однак надмірна кількість полігонів призводить до збільшення навантаження на графічний процесор, оперативну пам'ять та систему обробки кадрів, що може спричинити зниження частоти кадрів і погіршення плавності відтворення сцени. З цієї причини під час розробки моделей для комп'ютерних ігор та інтерактивних до-

датків широко застосовується підхід створення low-poly моделей. Такі моделі характеризуються мінімально необхідною кількістю полігонів, яка дозволяє зберегти впізнавану форму та основні особливості об'єкта без зайвої геометричної деталізації.

Ретопологія – один із найбільш дієвих методів оптимізації, який передбачає перебудову полігональної сітки для отримання простішої та ефективнішої структури. Під час створення 3D-об'єктів художники працюють із високодеталізованими моделями (high-poly), які можуть містити сотні тисяч або навіть мільйони полігонів. Такі моделі дозволяють максимально точно відтворити форму об'єкта, проте використання подібних моделей у середовищах реального часу є практично неможливим через надмірне навантаження на графічну систему. Саме тому застосовується процес ретопології, у ході якого створюється нова полігональна сітка поверх вже існуючої високополігональної моделі.

Основним завданням є побудова більш оптимальної структури полігонів, яка зберігає загальний силует і пропорції об'єкта, але містить значно меншу кількість елементів. При цьому особлива увага приділяється правильному розташуванню ребер (edge flow) та полігональних петель (edge loops), що забезпечують природну деформацію моделі під час анімації.

Ретопологію виконують як вручну, так і за допомогою спеціалізованих автоматичних інструментів у програмному забезпеченні для 3D-модельовання. Ручна ретопологія дозволяє точніше контролювати структуру полігональної сітки та оптимально розміщувати полігони у важливих ділянках моделі, наприклад у місцях згинів або рухомих частинах персонажа (рис.1). Автоматичні алгоритми, у свою чергу, значно прискорюють процес створення базової оптимізованої сітки, після чого вона може додатково редагуватися вручну.

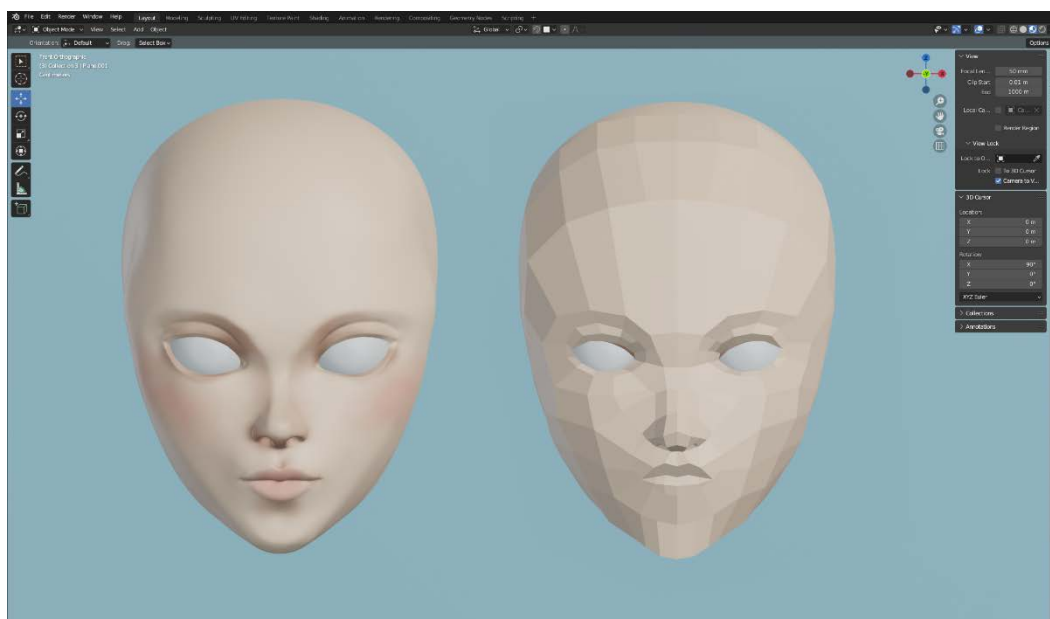


Рисунок 1 – Приклад ручної ретопології

Реалізація рівнів деталізації (Level of Detail, LOD) – це метод, який сприяє ефективності роботи системи. Суть цього підходу полягає у використанні кількох варіантів однієї й тієї ж 3D-моделі, що відрізняються рівнем геометричної деталізації та кількістю полігонів. Для кожного об'єкта створюється декілька версій моделі (рис.2).

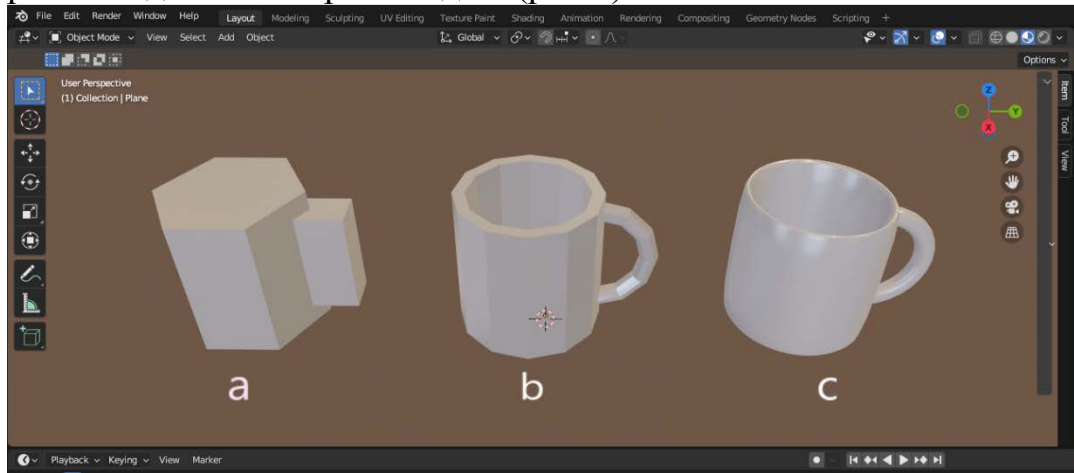


Рисунок 2: а – LOD0, b – LOD2, c – LOD3

Коли об'єкт знаходиться близько до камери, використовується модель з найбільшою кількістю полігонів, і в міру віддалення об'єкта від камери ігровий рушій перемикається на спрощені версії моделі. Оскільки віддалені об'єкти займають менше місця на екрані, користувач практично не помічає зменшення рівня деталізації. З таким підходом можна значно знизити навантаження на апаратні ресурси комп'ютера або мобільного пристрою.

Текстурні карти дозволяють передавати дрібні деталі поверхні без додавання нових полігонів. Наприклад, замість моделювання окремих камінців, їх можна додати як зображення на площині (рис. 3), завдяки чому зберігається вигляд деталізації при значно меншій геометричній складності.

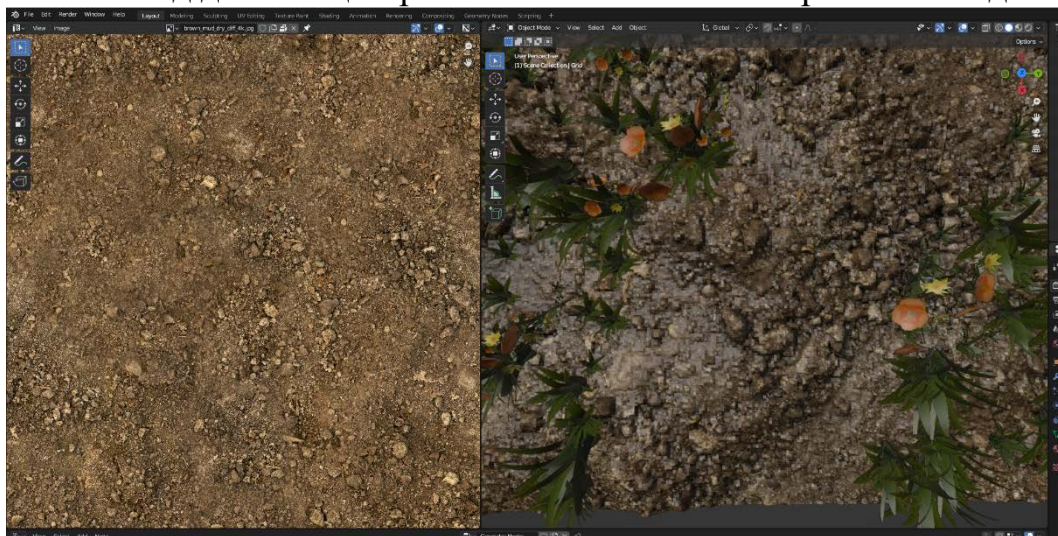


Рисунок 3 – Приклад текстурної карти

Також значну роль відіграє раціональне використання текстур. Текстури великої роздільної здатності займають багато пам'яті та можуть уповільнити завантаження сцени. Для уникнення цього застосовуються методи зменшення роздільної здатності чи компресії текстур, а також створення єдиного файлу, що об'єднує кілька текстур (рис.4). Це дозволяє скоротити кількість викликів графічного рушія і покращити продуктивність.

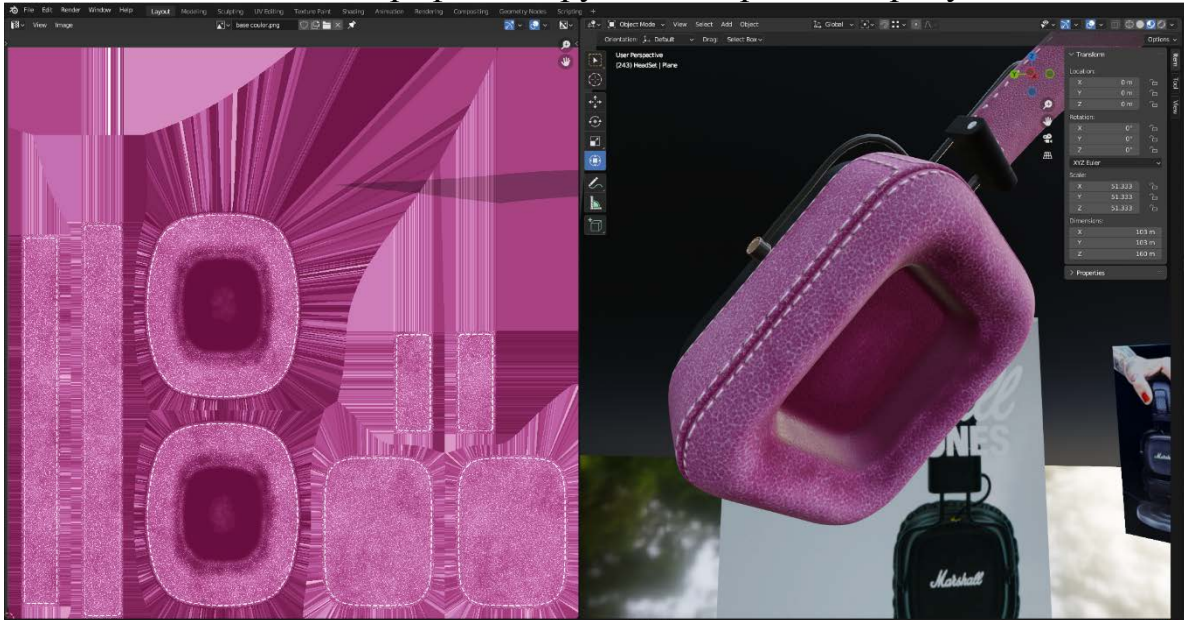


Рисунок 4 – Об'єднана текстурна карта

Таким чином, оптимізація 3D-моделей є невід'ємною частиною процесу створення комп'ютерних ігор та інтерактивних середовищ. Застосування показаних технік, дає змогу досягти оптимального балансу між якістю графіки та стабільною продуктивністю системи. У перспективі подальший розвиток графічних технологій та програмних інструментів відкриє шлях до нових підходів в оптимізації, що дозволить створювати ще більш деталізовані та реалістичні віртуальні світи без втрат в ефективності.

Список використаних джерел:

1. Blender Foundation. Blender Documentation: Official Manual [Electronic resource]. <https://docs.blender.org> (дата звернення 09.03.2026).
2. Foley J.D., van Dam A., Feiner S.K., Hughes J.F. Computer Graphics: Principles and Practice. – Boston: Addison-Wesley, 2014. – 1264 p.
3. Lengyel E. Mathematics for 3D Game Programming and Computer Graphics. – 3rd ed. – Boston: Cengage Learning, 2012. – 656 p.
4. Gregory J. Game Engine Architecture. – 3rd ed. – Boca Raton: CRC Press, 2018. – 1136 p.
5. Murdock K. 3ds Max Modeling for Games: Insider's Guide to Game Character, Vehicle, and Environment Modeling. – Indianapolis: Wiley Publishing, 2010. – 504 p.