

АНАЛІЗ ШЕЙДЕРІВ ДЛЯ ФОРМУВАННЯ ГРАФІЧНИХ ЗОБРАЖЕНЬ І ЇХ ПОКОЛІНЬ

¹Вінницький національний технічний університет, ²Національний університет «Одеська політехніка»

Анотація

У роботі проаналізовано поняття шейдерів, основні типи шейдерів, покоління розвитку шейдерів, а також перспективи розвитку шейдерів.

Ключові слова: шейдери, вершинні шейдери, геометричні шейдери, тесселяційні шейдери, фрагментні шейдери, покоління шейдерів, OpenGL, DirectX.

Шейдери [1-7]— це програми, які виконуються на графічному процесорі (GPU) для обробки різних етапів рендерингу. Вони дозволяють контролювати, як відображаються пікселі, вершини та інші елементи сцени. Існують різні типи шейдерів, кожен з яких має своє призначення і виконується на різних етапах процесу рендерингу. Існує кілька поколінь шейдерів, які відрізняються за можливостями і складністю. Детальний аналіз шейдерів починається з розгляду їх основних типів і еволюції технологій.

Найбільш поширеними типами шейдерів є вершинний шейдер, фрагментний шейдер (також відомий як піксельний шейдер), геометричний шейдер і тесселяційний шейдер [8]. Кожен з них виконується на певних етапах обробки графічних даних. Вершинний шейдер відповідає за обробку вершин геометричних об'єктів. Він виконується до того, як відбувається трансформація координат об'єкта на екран. Фрагментний шейдер працює після обробки геометрії і визначає колір пікселя на екрані, включаючи обчислення світла, тіней, текстур і матеріалів. Геометричний шейдер [3] дозволяє працювати з примітивами, створюючи нові вершини або модифікуючи існуючі, що дає можливість генерувати складніші форми. Тесселяційний шейдер використовується для деталізації геометрії, поділяючи великі примітиви на менші, що дозволяє покращити якість зображення.

Покоління шейдерів починається з перших версій OpenGL та DirectX [9], коли можливості графічних карт були обмежені. Перше покоління шейдерів з'явилося з шейдерами версії 1.0, які дозволяли виконувати прості операції, такі як зміна кольору пікселів. Згодом, у другому поколінні (Shader Model 2.0 для DirectX і OpenGL Shading Language (GLSL)), шейдери стали більш гнучкими та потужними, з'явилися можливості для використання текстур, простих математичних операцій і базових трансформацій.

У третьому поколінні (Shader Model 3.0) шейдери отримали значно більшу потужність, з можливістю використання складних математичних операцій і більш детальних текстур. Це дозволило створювати більш реалістичні графічні ефекти, такі як світлові відбиття, складні тіні, об'ємне освітлення та інші ефекти. Третє покоління шейдерів стало основою для розвитку таких графічних технологій, як перехід до реалістичних візуальних ефектів і фізично коректного рендерингу.

Четверте покоління (Shader Model 4.0) дало ще більше можливостей, включаючи підтримку обчислень на GPU для більш складних обчислювальних задач, таких як обробка часток і реалістичні матеріали, що включають освітлення на основі фізичних характеристик поверхні (PBR). Відтоді графічні процесори отримали значні можливості для реалізації високоякісної графіки та ефектів, що зумовило появу таких технологій, як глобальне освітлення, відображення в реальному часі та висококласні текстури.

З п'ятим поколінням (Shader Model 5.0 [9]) шейдери отримали підтримку ще складніших операцій, таких як використання нових типів даних і складних математичних обчислень. Це покоління дозволяє реалізовувати більш точні і складні фізичні моделі, а також використовувати нові типи шейдерів, такі як тесселяційний і compute-шейдери [9], які дозволяють виконувати загальні обчислення на графічному процесорі, що робить графічні додатки ще більш ефективними.

Покоління шейдерів постійно розвивається, з кожним новим етапом з'являються нові можливості для програмістів, які дозволяють створювати більш реалістичні та складні графічні ефекти. Важливим аспектом є розвиток API, таких як DirectX 12 і Vulkan, які дають можливість програмістам повністю контролювати процеси рендерингу і виконання шейдерів, що відкриває нові горизонти в створенні графічних додатків і ігор. Сьогодні шейдери є незамінним інструментом у розробці високоякісної тривимірної графіки, і їхня роль в індустрії комп'ютерної графіки тільки зростає.

У майбутньому розвиток шейдерів буде орієнтований на покращення реалістичності, ефективності і взаємодії з новими технологіями, такими як штучний інтелект, машинне навчання і розширена реальність. Очікується кілька ключових напрямків розвитку:

Оскільки фізично коректний рендеринг (PBR) став стандартом для реалістичних візуальних ефектів, в майбутньому шейдери будуть ще більш точними у відтворенні матеріалів і світла. Це включатиме ще більш детальні шейдери для складних ефектів, таких як субповерхневе розсіювання, розсіювання світла в різних матеріалах і віртуальні тіні, що дозволяють створювати глибші і правдоподібніші сцени.

Глобальне освітлення, яке включає складні процеси поширення світла через сцену, буде поліпшуватися за рахунок нових технік, таких як ray tracing (трасування променів). Вже зараз трасування променів використовується в рендерингу для створення реалістичних відображень і тіней, і в майбутньому цей процес буде ще більш оптимізований для реального часу, що дозволить використовувати складніші моделі освітлення, які можуть змінюватися в залежності від умов сцени.

Розвиток віртуальної та доповненої реальності вимагатиме нових типів шейдерів, здатних створювати більш реалістичні та інтерактивні середовища. Це включатиме використання шейдерів для інтерактивних елементів у реальному часі, таких як покращене відображення навколишнього світу, більш точне освітлення та ефекти взаємодії з реальними об'єктами.

Оскільки штучний інтелект і машинне навчання стають важливою частиною графічних технологій, можна очікувати на появу шейдерів, які будуть спеціально оптимізовані для таких задач. Це можуть бути шейдери, які реалізують нейронні мережі на графічному процесорі для створення динамічних текстур, синтетичних моделей поведінки або навіть для генерації контенту в реальному часі.

Оскільки відеоігри і графіка намагаються все більше імітувати реальність, будуть розроблені нові шейдери для більш складних і реалістичних ефектів води, туману, диму, дощу та інших природних явищ. Ці шейдери зможуть моделювати складні фізичні процеси, як-от рефракцію світла у воді або динамічну взаємодію часток у повітрі.

Оскільки графічні процесори стають все більш потужними, виникає потреба в оптимізації рендерингу на кількох пристроях або в хмарі. Можна очікувати на нові шейдери, які дозволять розподіляти обчислення між різними обчислювальними одиницями (наприклад, графічними картами, процесорами або хмарними сервісами), що дозволить досягти ще більшої реалістичності і ефективності при рендерингу великих і складних сцен.

Висновок. Загалом, майбутнє шейдерів буде включати більшу інтеграцію з новітніми технологіями, такими як машинне навчання, квантові обчислення і розширена реальність, що дозволить створювати ще більш складні і реалістичні графічні ефекти для різноманітних сфер застосування — від відеоігор до наукових візуалізацій і кіноіндустрії.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Романюк О. Н., Марущак А. В., Шмалюк В. А. Шейдери та їх мови програмування. *The 18 th International scientific and practical conference « MODERN SCIENCE, PRACTICE, SOCIETY »*, Boston, USA, 25-26 May 2020. 2020. С. 402-407.

2. Завальнюк Є. К., Романюк О. Н., Новосельцев О. О. Особливості програмування шейдерів за допомогою мови Gelato. *The 1st International scientific and practical conference «Innovative scientific research: theory, methodology, practice»*, Boston, 03-06 вересня 2024 р. 2024. Рр. 276-281.
3. Завальнюк Є. К., Романюк О. Н., Стахов О. Я. Аналіз використання геометричних шейдерів для програмної візуалізації тривимірних сцен. *Proceedings of the 3rd International Scientific and Practical Internet Conference «Recent Trends in Science»*, Дніпро, 16-17 травня 2024 р. Дніпро, 2024. С. 77-79.
4. Real-Time Rendering (Fourth Edition) / T. Akenine-Möller et al. A. K. Peters, 2018. 1040 p.
5. Rost R. J. OpenGL Shading Language (4th Edition). Addison-Wesley, 2016. 544 p.
6. Engel W. GPU Pro (Volume 1-6). A. K. Peters, 2010–2017. 400 p.
7. Vivo P. G. The Book of Shaders: A Guide to GLSL. 2017. 161 p.
8. Kenwright. Introduction to Computer Graphics and the Vulkan API. CRC Press, 2019. 450 p.
9. Романюк О. Н., Завальнюк Є. К. Аналіз моделей програмних шейдерів у DirectX. *Інформаційні технології в культурі, мистецтві, освіті, науці, економіці та бізнесі: матеріали IX Міжнародної науково-практичної конференції*, Київ, Україна, 2024. 2024. С. 39-42.

Романюк Олександр Никифорович – д. т. н., професор, завідувач кафедри програмного забезпечення, Вінницький національний технічний університет, м. Вінниця, rom8591@gmail.com

Завальнюк Євген Костянтинович – аспірант кафедри програмного забезпечення, Вінницький національний технічний університет, м. Вінниця, qq9272627@gmail.com

Романюк Сергій Олександрович – к. т. н., старший викладач кафедри біомедичної інженерії, Національний університет «Одеська політехніка», м. Одеса, romaniuk.s.o@op.edu.ua

Стахов Олексій Ярославович – к. т. н., старший викладач кафедри програмного забезпечення, Вінницький національний технічний університет, м. Вінниця, aleksey.stakhov@gmail.com

ANALYSIS OF SHADERS FOR IMAGE FORMATION AND THEIR GENERATIONS

Abstract

In the article, it is analyzed the concept of shaders, main types of shaders, generations of shaders, and the prospects of shaders development.

Keywords: *shaders, vertex shaders, geometric shaders, tessellation shaders, fragment shaders, shader generations, OpenGL, DirectX.*

Romanyuk Oleksandr N. – doctor of technical sciences, professor, Head of the Software Department, Vinnytsia National Technical University, Vinnytsia, rom8591@gmail.com

Zavalniuk Yevhen K. – postgraduate student of the Software Department, Vinnytsia National Technical University, Vinnytsia, qq9272627@gmail.com

Romanyuk Sergey O. – candidate of technical sciences, senior lecturer of the Biomedical Engineering Department, Odesa National Polytechnic University, Odesa, romaniuk.s.o@op.edu.ua

Stakhov Oleksii Y. – candidate of technical sciences, senior lecturer of the Software Department, Vinnytsia National Technical University, Vinnytsia, aleksey.stakhov@gmail.com