

МОДИФІКАЦІЯ МЕТОДІВ АНІЗОТРОПНОЇ ФІЛЬТРАЦІЇ

Проаналізовано сучасні методи текстурування. Запропоновано для анізотропної фільтрації метод простого визначення координат текселів, що визначають колір екранного пікселя, з урахуванням кута нахилу полігона. Запропоновано метод визначення усередненого значення кольору пікселя з використанням вагових функцій.

Ключові слова: текстурування, анізотропна фільтрація, фільтрація текстур

O.N. ROMANIUK, O.O. DUDNYK, S.I. VYATKIN
Vinnytsia National Technical University

MODIFICATION OF ANISOTROPIC FILTERING

This article considers the modern methods of texturing. A simple method of determining the coordinates of texels that determine pixel color screen, given the inclination angle ground for anisotropic filtering. The method of calculating the average pixel color values using weighting functions.

Keywords: texturing, anisotropic filtering, texture filtering

Постановка задачі

При побудові високореалістичних зображень використовують текстури[1], які накладають на графічні об'єкти. Текстурування дозволяє успішно вирішувати задачі, які надзвичайно трудомістко розв'язати прямими методами [1], дозволяє суттєво зменшити обчислювальні витрати та зробити можливим інтерактивний режим візуалізації [1]. Застосовані в тривимірній графіці методи накладення текстур, використовуються для візуалізації тривимірних сцен з високим ступенем деталізації. Генерація текстури полягає в проектуванні зображення на тривимірну поверхню. Таким чином забезпечується додаткова деталізація об'єкта без ускладнення його геометрії. Для імітації реалістичних сцен необхідно використовувати велику кількість деталізованих текстур.

Для визначення кольору екранного пікселя використовують усереднення кольору всіх текселей, проекція яких відповідає даному пікселю.

Серед методів фільтрації текстур найбільш поширеними є білінійна та трилінійна фільтрації.

Білінійна фільтрація [3] передбачає розрахунок кольору пікселя на екрані, за допомогою усереднення кольору чотирьох текселів, що задають проекцію цього пікселя з плоского екрану на тривимірну площину.

Трилінійна фільтрація [3] є комбінацією mір-текстурування та білінійної фільтрації. Фактично виконується білінійна фільтрація на двох мір-рівнях, що в результаті дає 2 текселя по одному для кожного мір-рівня. Колір пікселя, який повинен бути виведений на екран, визначається в результаті інтерполяції за кольорами двох мір-текстур [2].

Ці методи текстурування дозволяють відносно коректно розраховувати колір тільки для тих пікселів, відповідні текселя яких знаходяться в текстурній площині, яка паралельна екрану. Таке обмеження викликане тим, що вибірка груп з чотирьох текселей при білінійної фільтрації відбувається по строго заданому закону.

Текселя, як правило, апроксимують коло, що і є проекцією екранного пікселя на площину текстури. Чим більше поверхня текстури відхиляється від паралелі екрану, тим більша буде значення похибки при визначенні кольору пікселя. У результаті чого, текстури, розташовані під гострими кутами, сильно розмиваються [2].

Тому сьогодні при текстуруванні широко використовується анізотропна фільтрація, яка дозволяє більш точно визначати кольори пікселів, які відповідають елементам текстури, що розташовані не паралельно екрану. Це найбільш якісний і поширений метод текстурування. На відміну від ізотропних видів фільтрації (білінійна та трилінійна) використовується проекція пікселя на текстурну поверхню [2].

При анізотропній фільтрації проекція пікселя на поверхню текстури розглядається не як коло, а як витягнутий еліпс (рис. 1), що дозволяє точніше визначати кольори пікселів [3].

Для того, щоб коректно обчислити колір пікселя, необхідно врахувати кольори всіх текселів [1], які охоплює еліпс. Це достатньо складна процедура для генерації зображень у реальному часі, тому використовують деякі спрощення. Основна ідея таких спрощень – це наближення еліпса з великим ексцентриситетом декількома еліпсами з меншими

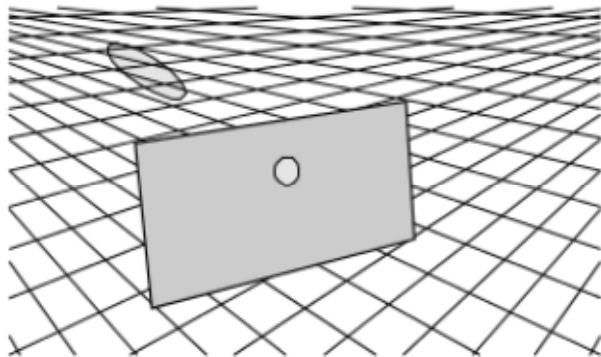


Рис. 1. Проекція пікселя на поверхню текстури

екцентриситетами. Таким чином, можна подолати обмеження, накладені апаратними засобами та забезпечити фільтрацію найкращим фільтром з високим ступенем анізотропії. Найчастіше еліпс розбивають на менші за площею кола рівного

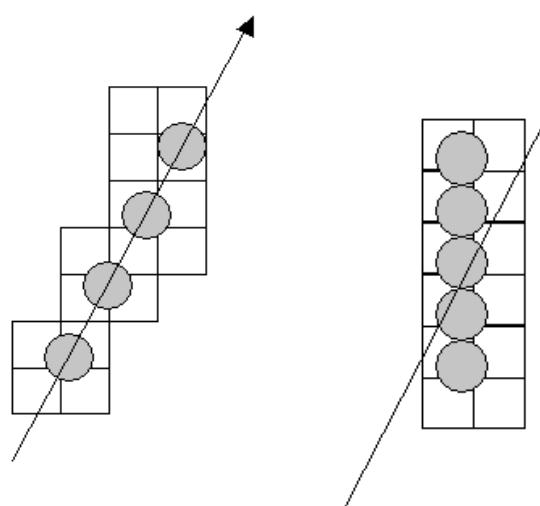


Рис. 3. Визначення точок, що належать до проекції пікселя на текстуру

Рис. 4. Спрощення при визначенні точок, що належать до проекції пікселя на текстуру

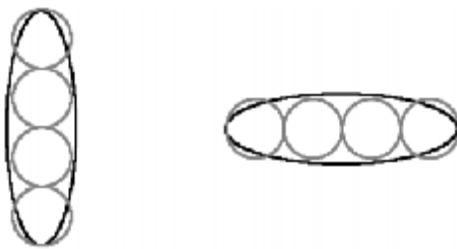


Рис. 2. Розбиття еліпса на кола

діаметру (рис. 2). Потім знаходять середні значення кольору для кожного кола [4].

На практиці коло апроксимують квадратом площею чотири в текселі. Таким чином, фактично виконується білінійна фільтрація у кількох точках вздовж вектору нахилу полігона, а отримані значення усереднюються (рис. 3). Кількість таких точок залежить від кута нахилу.

Визначення координат усіх необхідних текселів є ресурсоємкою задачею, тому ряд виробників графічних процесорів спрощують її, розглядаючи лише два часткові випадки положення

вектору нахилу: паралельно осі OX і перпендикулярно їй.

У такому випадку координати центру проекції в координатах текстури визначать за формулою:

$$\begin{cases} cx = x_d \times \frac{w_d}{w_t}, \\ cy = y_d \times \frac{h_d}{h_t}, \end{cases} \quad (1)$$

де x_d, y_d - відносні координати пікселя в екранній площині; w_d, h_d - ширина та висота, проекції полігона на екранну площину; w_t, h_t - ширина та висота полігона в текстурній площині. Координати одного із чотирьох пікселів ($P1$), що формують квадрат, визначать шляхом округлення координат отриманої точки (C) до більшого цілого (рис. 5). Для визначення координат точок $P2, P3, P4$ аналізують дробову частину i . Для пікселя $P2$ у-координата рівна у-координаті пікселя $P1$, а x -координата на 1 менша x -координаті пікселя $P2$, якщо дробова частина менша 0.5 або на 1 більша якщо дробова частина більша за 0.5. Analogічним чином обчислюється у-координата для $P4$. Для $P3$ x -координата рівна x -координаті пікселя $P2$, а y -координата - y -координаті пікселя $P4$.

Координати пікселів, що утворюють інші квадрати, які входять до проекції, знаходять шляхом зміщення $P1, P2, P3, P4$ на 2 пікселя вверх і вниз, якщо полігон має більший нахил відносно OX ніж відносно OY (кут вектора нахилу у текстурній площині близький до $0^\circ, 90^\circ, 180^\circ$ або 270°) або на 2 пікселя вправо та вліво, якщо нахил відносно OY більший нахилу відносно OX (кут вектора нахилу у текстурній площині близький до 0° або 180°).

Такий підхід забезпечує високу продуктивність, проте він не враховує ситуацію, коли полігон значно відхилено як від осі OX так і від OY , що зумовлює появу артефактів. Тому даний метод можна покращити шляхом обрахунку координат пікселів для кутів, близьких до $45^\circ, 135^\circ, 225^\circ$, або 315° .

Очевидно, що для таких кутів координати текселів, що утворюють додаткові квадрати повинні визначатись шляхом зміщення $P1, P2, P3, P4$ як по осі OX , так і по осі OY . Проте слід урахувати, що центр проекції (C) у більшості випадків зміщено відносно центру квадрата, обумовленого $P1, P2, P3, P4$, а тому для коректності обрахунків необхідно врахувати це відхилення.

Таким чином, необхідно врахувати дробову частину. Якщо вона більша за 0.5, то зсув потрібно

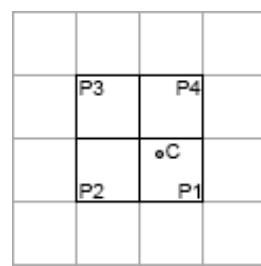


Рис. 5. Координати центру проекції

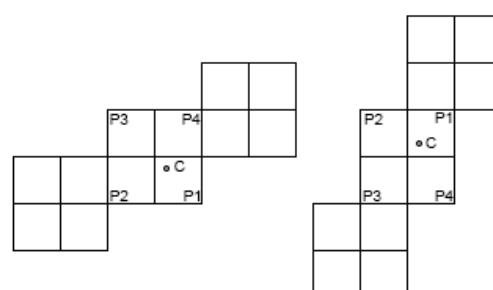


Рис. 6. Текселі що належать до проекції пікселя на текстуру для кута нахилу 45°

виконувати на 2 пікселя по осі OY і на 1 по осі OX , а якщо менша, то навпаки (рис. 6).

З метою підвищення реалістичності відтворення тривимірних сцен при використанні анізотропної фільтрації важливо врахувати те, що інтенсивність кольору всередині пікселя зменшується нелінійно від його центра до периферійних зон. Тому підвищити реалістичність формування вихідного зображення описаним методом можна шляхом використання ваговоих функцій при обрахунку усередненого значення кольору пікселя відповідно до Гаусівської моделі (рис. 7).

Таким чином, усереднене значення кольору в кожному квадраті слід обраховувати за формулою:

$$P = \frac{P_1 B_1 + P_2 B_2 + P_3 B_3 + P_4 B_4}{4}, \quad (2)$$

де P_1, P_2, P_3, P_4 – кольори точок на текстурній площині, B_1, B_2, B_3, B_4 – ваги відповідних точок. Ваги точок визначаються відповідно Гаусівської моделі пікселя. Тому для всіх точок квадрату, що лежить у центрі проекції ваги рівні 0.25. Для інших квадратів точка що, лежить більше до центру проекції матиме вагу 0.3, протилежна їй по діагоналі – 0.2, а інші по 0.25. Для випадків, коли кут вектора нахилу у текстурній площині близький до $0^\circ, 90^\circ, 180^\circ$ або 270° , ваги двох точок, що лежать більше до центру проекції рівні 0.3, а двох інших по 0.2 (рис. 8).

Отримані значення для всіх квадратів слід усереднити за формулою:

$$P = \frac{P_1 B_1 + P_2 B_2 + \dots + P_N B_N}{N}, \quad (3)$$

де P_1, P_2, \dots, P_N – усереднені значення кольорів для кожного квадрату, B_1, B_2, \dots, B_N – ваги відповідних квадратів, N – кількість квадратів. Ваги квадратів необхідно обирати таки чином, щоб сумарна вага була рівною 1, вага квадрату у центрі проекції була найбільшою, а ваги інших квадратів зменшувались по мірі віддалення від центру проекції. На приклад, для трьох квадратів можна обрати ваги 0.3, 0.4, 0.3 (рис. 9).

Висновки

Анізотропна фільтрація дозволяє досягти більш високої реалістичності формування графічних сцен. Запропоновано метод простого визначення координат текстелів, що визначають колір екранного пікселя, з урахуванням кута нахилу полігона.

Література

1. Романюк О. Н. Високопродуктивні методи та засоби зафарбовування тривимірних графічних об'єктів. Монографія. / О. Н. Романюк, А. В. Чорний. - Вінниця : УНІВЕСУМ-Вінниця, 2006. — 190 с.
2. Гусятин В. М. Метод анизотропной фильтрации текстур при синтезе изображений обратным трассированием / В. М. Гусятин, Я. В. Чаговец, Д.Г. Кожушко // Информатика, кибернетика и вычислительная техника (ИКВТ-2009). Выпуск 10 (153). – Донецк: ДонНТУ. – 2009. – С. 64-69.
3. Романюк О. Н. Спрощення процедури накладання текстур на тривимірні графічні об'єкти / О. Н. Романюк // Вимірювальна та обчислювальна техніка в технологічних процесах. — 2006. — № 2. — С. 114—118.
4. Olano M. Vertex-based Anisotropic Texturing Proceedings of Graphics Hardware 2001: the 16th ACM/Eurographics Symposium on Graphics Hardware (Los Angeles, CA, August 12-13, 2001) / Marc Olano, Shrijeet Mukherjee, Angus Dorbie, . — С 95–98.

References

1. Romanyuk O. N. Vy'sokoproduktivnyi metody ta zasoby zaferbovuvannya try'vy'mirnyx grafichnyx ob'yektiv. Monografiya. / O. N. Romanyuk, A. V. Chornyy. - Vinny'cya : UNIVESUM-Vinny'cya, 2006.—190 s.
2. Gusyat'yn V. M. Metod anyzotropnoj fy'l'tracijy tekstur pry' sy'nteze y'zobrazhenyj obratnym trassy'rovany'em / V. M. Gusyat'yn, Ya. V. Chagovecz, D.G. Kozhushko // Ynformaty'ka, ky'bernety'ka y' vychy'slytel'naya texn'ka (Y'KVT-2009). Vypusk 10 (153). – Doneck: DonNTU. – 2009. – C. 64-69.
3. Romanyuk O. N. Sproshhennya procedury nakladannya tekstur na try'vy'mirni grafichni ob'yekty / O. N. Romanyuk // Vy'miryal'na ta obchyslival'na texnika v texnologichnyx procesax. — 2006. — # 2. — S. 114—118.
4. Olano M. Vertex-based Anisotropic Texturing Proceedings of Graphics Hardware 2001: the 16th ACM/Eurographics Symposium on Graphics Hardware (Los Angeles, CA, August 12-13, 2001) / Marc Olano, Shrijeet Mukherjee, Angus Dorbie, . — S 95–98.

Рецензія/Peer review : 27.9.2015 р.

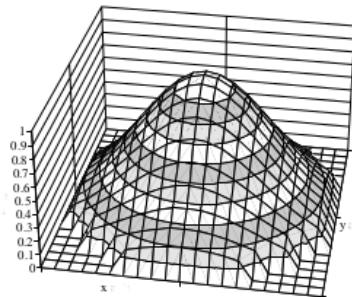


Рис. 7. Гаусівська модель пікселя

	0.25	0.2	0.2	0.2
	0.3	0.25	0.3	0.3
0.25	0.25	0.25	0.25	0.25
0.25	0.25	0.25	0.25	0.25
0.25	0.3	0.3	0.3	0.3
0.2	0.25	0.25	0.2	0.2

Рис. 8. Значення ваг точок

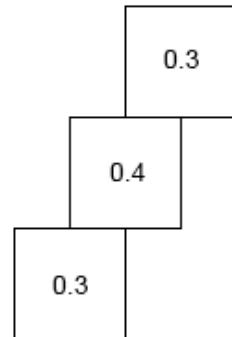


Рис. 9. Ваги квадратів