УДК 004.925.4

О.Н. РОМАНЮК
Вінницький національний технічний університет
О.О. ДУДНИК
Вінницький національний технічний університет

ПІДВИЩЕННЯ ПРОДУКТИВНОСТІ ПЕРСПЕКТИВНО-КОРЕКТНОГО ТЕКСТУРУВАННЯ З ВИКОРИСТАННЯМ АНІЗОТРОПНОЇ ФІЛЬТРАЦІЇ

Запропоновано метод зниження обчислювальної складності визначення часткових похідних в процесі обчислення положення проекції піксела на текстурну площину за методом Хекберта, що дозволяє зменшити кількість операцій додавання та множення шляхом використання ітераційних формул і, як наслідок, істотно підвищити продуктивність.

Ключові слова: текстурування, перспективно-коректне текстурування, анізотропна фільтрації

O.N. ROMANIUK
Vinnytsia National Technical University
O.O. DUDNYK
Vinnytsia National Technical University

INCREASED PRODUCTIVITY PERSPECTIVE-CORRECT TEXTURE MAPPING WIDTH ANISOTROPIC FILTERING

The proposed method reduce the computational complexity of determining the partial derivatives in the process of calculating the position of the projection of pixel on textured plane method Heckbert, which reduces the number of operations of addition and multiplication through the use of iterative formulas and, consequently, to improve performance.

Keywords: texture mapping, perspective-correct texture mapping, anisotropic filtering

Вступ

Підвищення інформативності комп'ютерної графіки досягають за рахунок формування зображень, які точніше відтворюють конструктивні та візуальні особливості об'єкта. При формуванні таких зображень у графічних системах необхідно відображати сцени з великою деталізацією, тому на даному етапі розвитку комп'ютерної графіки особливу увагу приділяють не тільки швидкодії формування графічних зображень, але й їх реалістичності. Тому при побудові високореалістичних зображень використовують текстури, які накладають на графічні об'єкти [1-4].

Аналіз літератури та постановка задачі

Текстурування — це спосіб надання фактурних особливостей 3D деталі — полігону: кольору, блиску, матовості та інших фізичних властивостей (для імітації природного матеріалу, наприклад: паперу, дерева, каменю, металу тощо) [1-4].

Текстурування є важливою процедурою 3D-рендерингу, оскільки дозволяє відтворити також малі об'єкти поверхні, створення яких полігонами виявилося б надмірно ресурсомістким. Наприклад, складки на одязі, дрібні камені, предмети на поверхні стін і ґрунту та багато іншого [1, 3]. Текстура використовується в якості шару для додання певного ефекту або зміни геометрії всьому зображенню або його частини. Генерація текстури полягає в проектуванні зображення на тривимірну поверхню.

Використання текстур для імітації нерівностей на поверхні передбачає встановлення співвідношення між екранними координатами об'єкта та координатами текстури. Розрізняють афінне [1, 3] та перспективно-коректне накладання текстур [1-3].

При афінному текстуруванні [1, 2] текстурні координати лінійно інтерполюються вздовж рядка растеризації, що призводить до появи артефактів та недостовірного відтворення перспективи об'єкта.

При перспективно-коректному текстуруванні, для визначення координат текселя, що відповідає екранному пікселю, використовують нелінійні функції, розрахунок яких передбачає попіксельне виконання трудомістких операцій. Перспективно-коректне текстурування у переважній більшості випадків реалізують за методом Хекберта [1-3]:

$$u = \frac{Ax + By + C}{Gx + Hy + I}, \qquad v = \frac{Dx + Ey + F}{Gx + Hy + I}, \tag{1}$$

де u і v – текстурні координати, x і y – екранні координати об'єкта, $A, B, C, D, E, F, G, H, I$ – коефіцієнти полігона, який текстурується.

Визначення кольору піксела на екрані шляхом врахування кольору лише одного текселя, призводить то появи різноманітних помилок візуалізації – артефактів. Тому для визначення кольору піксела

використовують різні методи фільтрації текстур [1,4]. На даному етапі 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 піксела на текстурну поверхню [1, 4].

При анізотропної фільтрації проекція піксела на поверхню текстури розглядається а як витягнутий еліпс (рис. 1) та передбачає усереднення кольору текселів усередині проекції, що дозволяє точніше визначати кольори пікселів [1, 4].

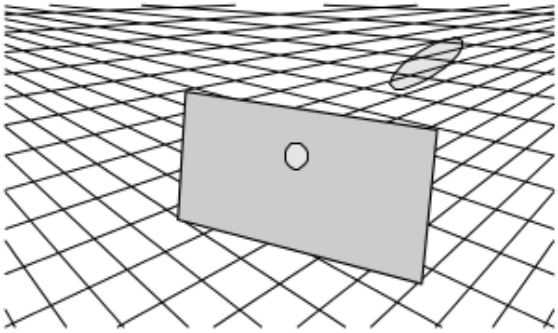

Рис. 1. Проекція піксела на поверхню текстури

Для визначення необхідних для фільтрації текселів складають рівняння еліпса за формулами [4]:

$$
\begin{aligned}
&(U_x, V_x) = \left(\frac{du}{dx}, \frac{dv}{dx}\right),\\
&(U_y, V_y) = \left(\frac{du}{dy}, \frac{dv}{dy}\right),\\
&A = {V_x}^2 + {V_y}^2,\\
&B = -2(U_x V_x + U_y V_y),\\
&C = {U_x}^2 + {U_y}^2,\\
&F = (U_x V_y + U_y V_x)^2,\\
&AU^2 + BUV + CV^2 = F,
\end{aligned}
\qquad (2)
$$

де $U = u - u_0,\ V = v - v_0.$

Часткові похідні знаходять за формулами

$$
\begin{aligned}
&\frac{\partial u}{\partial x} = u(x+1, y) - u(x, y) = \frac{A(x+1) + By + C}{G(x+1) + Hy + I} - \frac{Ax + By + C}{Gx + Hy + I},\\
&\frac{\partial v}{\partial x} = v(x+1, y) - u(x, y) = \frac{D(x+1) + Ey + F}{G(x+1) + Hy + I} - \frac{Dx + Ey + F}{Gx + Hy + I},\\
&\frac{\partial u}{\partial y} = u(x, y+1) - u(x, y) = \frac{Ax + B(y+1) + C}{Gx + H(y+1) + I} - \frac{Ax + By + C}{Gx + Hy + I},\\
&\frac{\partial v}{\partial y} = v(x, y+1) - u(x, y) = \frac{Dx + E(y+1) + F}{Gx + H(y+1) + I} - \frac{Dx + Ey + F}{Gx + Hy + I}.
\end{aligned}
\qquad (3)
$$

Як видно з формул (1), (2), (3) обчислення часткових похідних для визначення коефіцієнтів рівняння еліпса є трудомісткою процедурою. На кожен піксел у рядку растеризації необхідно виконати 10 операцій множення, 10 операцій додавання, 4 операції ділення та 1 операцію віднімання. Тому існує потреба у розробці методів, які б мали меншу обчислювальну складнісить.

Знайдемо часткові похідні для формул (3), підставивши значенні із формул (1):

$$
\frac{A(x+1) + By + C}{G(x+1) + Hy + I} - \frac{Ax + By + C}{Gx + Hy + I} = \frac{AI - CG + AHy - BGy}{(I + Gx + Hy)(G + I + Gx + Hy)} \qquad (4)
$$

$$
\frac{D(x+1) + Ey + F}{G(x+1) + Hy + I} - \frac{Dx + Ey + F}{Gx + Hy + I} = \frac{DI - FG - DHy + EGy}{(I + Gx + Hy)(G + I + Gx + Hy)} \qquad (5)
$$

$$\frac{Ax+B(y+1)+C}{Gx+H(y+1)+I}-\frac{Ax+By+C}{Gx+Hy+I}=\frac{BI-CH-AHx+BGx}{(I+Gx+Hy)(G+I+Gx+Hy)} \quad (6)$$

$$\frac{Dx+E(y+1)+F}{Gx+H(y+1)+I}-\frac{Dx+Ey+F}{Gx+Hy+I}=\frac{EI-FH-DHx+EGx}{(I+Gx+Hy)(G+I+Gx+Hy)} \quad (7)$$

Як видно із формул (4) і (5) чисельники не залежать від значення y, а тому можуть бути обчислені один раз на кожен рядок растеризації. Крім того функції $AI-CG+AHy-BGy$ і $DI-FG-DHy+EGy$ ростуть лінійно, а тому можуть бути обчислені ітераційно. Визначимо приріст чисельників для формул (3) і (4) за методом кінцевих різниць [5]:

$$M=\left(AI-CG+AH\left(y+1\right)-BG\left(y+1\right)\right)-\left(AI-CG+AHy-BGy\right)=AH-BG, \quad (8)$$

$$N=\left(DI-FG-DH\left(y+1\right)+EG\left(y+1\right)\right)-\left(DI-FG-DHy+EGy\right)=DH-EG. \quad (9)$$

Із формул (8) і (9) видно що прирости функцій не залежать від координат точок, а лише від коефіцієнтів перетворення полігону, і тому можуть бути обчислені один раз на полігон. Таким чином, чисельники (num) формул (4) та (5) для першого рядка растеризації обчислюються за формулами:

$$num=AI-CG+AHy-BGy,$$
$$num=DI-FG-DHy+EGy,$$

а для кожного наступного за формулами:

$$num=num_{-1}+M,$$
$$num=num_{-1}+N.$$

Таким чином, кількість арифметичних операцій на один піксел істотно зменшується.

Із формул (6) і (7) видно що чисельники не залежать від значення x і можуть були обчислені один раз для кожного y. Визначимо приріст чисельників для формул (5) і (6) за методом кінцевих різниць:

$$O=\left(BI-CH-AH\left(x+1\right)+BG\left(x+1\right)\right)-\left(BI-CH-AHx+BGx\right)=BG-AH, \quad (10)$$

$$P=\left(EI-FH-DH(x+1)+EG(x+1)\right)-\left(EI-FH-DHx+EGx\right)=EG-DH. \quad (11)$$

Отже чисельники для формул (6) і (7) для найменшого y обчислюються за формулами:

$$num=BI-CH-AHx+BGx,$$
$$num=EI-FH-DHx+EGx,$$

а для кожного наступного за формулами:

$$num=num_{-1}+O,$$
$$num=num_{-1}+P.$$

Відповідно до формул (10) і (11) прирости чисельників для формул (5) та (6) також не залежать від координат точок, а лише від коефіцієнтів перетворення полігону, і тому можуть бути обчислені один раз на полігон.

Оскільки чисельники формул (7) і (8) залежать від значення x, при горизонтальному порядку растеризації інкрементація чисельника функцій повинна виконуватися для кожного піксела в рядку растеризації. Кожен наступний рядок растеризації містить точки x - координати яких рівні x - координатам точок в попередньому рядку растеризації. Тому значення чисельників формул (5) і (6) доцільно розраховувати для всіх y полігону до початку його растеризації з метою уникнення повторних розрахунків.

Обчислення знаменника ($denom$) для формул (3) – (6) також можна спростити. Позначимо вирази $G+I$ як J, $I+Hy$ як K, а $G+I+Hy=J+Hy$ як L. Тоді знаменник формул (3)-(6) має вигляд:

$$denom=(I+Gx+Hy)(G+I+Gx+Hy)=(K+Gx)(L+Gx). \quad (11)$$

Підставивши отримані аналітичні залежності у формули (3), отримуємо, що

$$\frac{\partial u}{\partial x}=\frac{num_{-1}+M}{(K+Gx)(L+Gx)},$$
$$\frac{\partial v}{\partial x}=\frac{num_{-1}+N}{(K+Gx)(L+Gx)},$$
$$\frac{\partial u}{\partial y}=\frac{num_{-1}+O}{(K+Gx)(L+Gx)}, \quad (12)$$
$$\frac{\partial v}{\partial y}=\frac{num_{-1}+P}{(K+Gx)(L+Gx)}.$$

Відповідно до формул (12) для обчислення часткових похідних достатньо 6 операцій додавання, 2 операції множення та 4 операції ділення на кожен піксел. Таким чином, порівняно з базовим методом кількість операцій додавання зменшено на 4, операцій множення зменшено на 8, а операцій віднімання зменшено на 1.

Висновки

Запропоновано метод зниження обчислювальної складності визначення часткових похідних при обчисленні положення проекції піксела на текстурну площину за методом Хекберта, що дозволяє зменшити кількість операцій додавання та множення шляхом використання ітераційних формул і, як наслідок, істотно підвищити продуктивність.

Література

1. Paul S. Heckbert, Survey of texture mapping - Graphics Interface ’1986

2. Романюк О.Н. Неортогональна растеризація при перспективно-коректному текстуруванні/ О.Н. Романюк, О.О. Дудник, О.В.Мельник, // VI міжнародна конференція «Моделювання та комп'ютерна графіка» ДВНЗ " ДонНТУ", м. Красноармійськ, 25-29 травня 2015 р.

3. Романюк О.Н. Метод підвищення продуктивності перспективно-коректного текстурування/ О.Н. Романюк, О.О. Дудник, // Наукові праці ДонНТУ №1(22),2016

4. Ned Greene, Creating raster omnimax images from multiple perspective views using the elliptical weighted average filter / Ned Greene, Paul S. Heckbert // IEEE CG&A 1986.

5. 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

References

1. Paul S. Heckbert, Survey of texture mapping - Graphics Interface '1986

2. Romanyuk O.N. Neortogonal`na rastery`zaciya pry` perspekty`vno-korektnomu teksturuvanni/ O.N. Romanyuk, O.O. Dudny`k, O.V.Mel`ny`k, // VI mizhnarodna konferenciya «Modelyuvannya ta komp'yuterna grafika» DVNZ " DonNTU", m. Krasnoarmijs`k, 25-29 travnya 2015 r.

3. Romanyuk O.N. Metod pidvy`shhennya produkty`vnosti perspekty`vno-korektnogo teksturuvannya/ O.N. Romanyuk, O.O. Dudny`k, // Naukovi praci DonNTU #1(22),2016

4. Ned Greene, Sreating raster omnimax images from multiple perspective views using the elliptical weighted average filter / Ned Greene, Paul S. Heckbert // IEEE CG&A 1986

5. Romanyuk O. N. Vy`sokoprodukty`vni metody` ta zasoby` zafarbovuvannya try`vy`mirny`x grafichny`x ob'yektiv. Monografiya. / O. N. Romanyuk, A. V. Chorny`j. - Vinny`cya : UNIVESUM-Vinny`cya, 2006. — 190 s

Рецензія/Peer review : 9.9.2016 р. Надрукована/Printed : 14.11.2016 р.
Стаття рецензована редакційною колегією