

Комп'ютерна графіка, візуалізація, обробка зображень та відео

УДК 004.274

О.Н. Романюк, д-р техн. наук, проф.,

О.О. Дудник, аспірант,

Вінницький національний технічний університет, м. Вінниця, Україна

Dudnyk1@inmad.vntu.edu.ua

Підвищення продуктивності текстурування з виконанням процедурних операцій в об'єктному просторі

Пропонується метод підвищення реалістичності формування графічних зображень, який усуває артефакти, характерні для методів текстурування в екранній площині. Метод передбачає виконання текстурування в просторі об'єкта з фільтрацією в площині екрану. При цьому забезпечується можливість розпаралелення процесу текстурування.

Ключові слова: текстурування, фільтрація текстур, анізотропна фільтрація.

Вступ

На даному етапі розвитку комп'ютерної графіки потужності графічних систем достатньо для забезпечення високої реалістичності формування графічних сцен. При цьому широко використовують методи текстурування, які надають поверхням об'єктів природної фактури.

Існують два основні підходи до зафарбовування поверхні, на яку накладають текстуру: растеризація в площині екрану та растеризація в текстурній площині [1, 2].

Растеризація в площині текстури є більш простою з обчислювальної точки зору, ніж в площині екрану, так як інверсія проєкції не виконується. Але рівномірною вибіркою в текстурній площині не гарантує рівномірну вибірку в площині екрану. Для неафінних відображень текстур растеризація в текстурній площині призводить до появи артефакту, який полягає в тому, що деякі пікселі не растеризуються (на поверхні з'являються не зафарбовані пікселі, які часто називають отворами). Можливі випадки і накладання пікселів [2].

Растеризація в площині екрану, яку іноді називають оберненою проєкцією, є найбільш поширеним методом. Для кожного піксела в площині екрану, знаходять відповідний текстель, колір якого відтворюють на екрані. Такий підхід не призводить до появи отворів або накладань в площині екрану [2].

Обидва методи базуються на обчисленнях з двовимірними координатами, оскільки екран і текстура є двовимірними площинами. При цьому не враховується реальне положення полігону в просторі. Проєкція полігону на екран має істотні відмінності від реального полігону, оскільки не забезпечується подібність. Як наслідок, обчислення текстурних координат для обчислення кольору виконуються з суттєвою похибкою, що призводить до появи артефактів на вихідному зображенні [1-6]. Для усунення артефактів, при

текстуруванні в площині екрану, виконують фільтрацію текстур. Найпоширенішим на даний час методом фільтрації є анізотропна фільтрація [2-6].

При анізотропній фільтрації проєкція піксела на поверхню текстури розглядається як витягнутий еліпс (рис. 1). Для того, щоб коректно обчислити колір піксела, необхідно врахувати кольори всіх текстелів, які охоплює еліпс. Обчислення форми та положення еліпса, визначення приналежності текстелів еліпсу та їх подальше усереднення – достатньо складна процедура і вимагає значних обчислювальних витрат. Площа еліпса залежить від відстані полігону до площини проєкції та його нахилу відносно координатних осей. Як правило, еліпси частково накладаються один на одного, що призводить до неодноразових обчислень над одними і тими ж текстелями. Тому на практиці, часто використовують спрощення, які призводять до втрати якості [2-6]. Крім того, існуючі методи розпаралелення [5,6] анізотропної фільтрації не забезпечують збереження якості зображення.

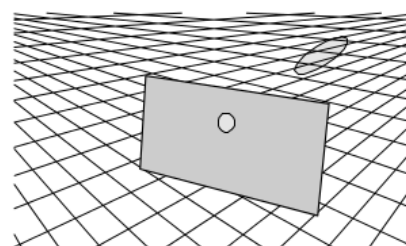


Рисунок 1 – Проєкція екранного піксела на текстурну площину

Метою дослідження є підвищення реалістичності формування графічних зображень шляхом усунення артефактів, характерних для методів растеризації в екранній площині та забезпечення можливості розпаралелення процесу текстурування без втрати точності.

Основною вимогою до методу є врахування положення полігону в просторі та глибини простору для точного визначення кольорів екранних пікселів.

Виклад основного матеріалу дослідження

Для виконання текстурування з урахуванням глибини простору необхідно виконувати текстурування в просторі об'єкта з подальшою проекцією кожного текселя на екранну площину, враховуючи, при цьому, його z -компоненту.

Полігон знаходиться в тривимірному дискретному просторі, а текселі в двовимірному дискретному просторі. Тому встановити відповідність текселів пікселям, що належать площині довільно-орієнтованого полігону, досить складно. Координати точок полігону, які відповідають точкам текстури, мають нецілочисельні координати, а їх визначення методом інтерполяції є некоректним. Тому можлива поява артефактів, характерних при растрезації в площині екрану (рис. 2)

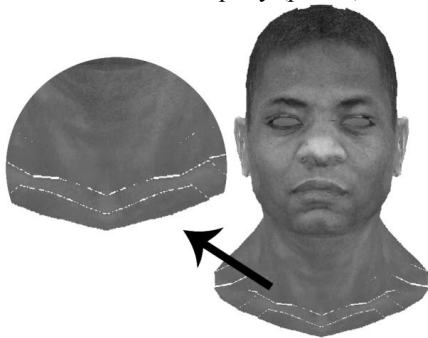


Рисунок 2 – Артефакти, характерні для методів растрезації в площині текстури та в просторі об'єкта

Для коректного врахування усіх точок, що формують площу полігону текстурування слід проводити у такій послідовності:

1. Виконати поворот полігону відносно координатних осей таким чином, щоб він зайняв положення паралельне площині проекції.
2. Визначити кольори пікселів полігону шляхом співставлення їх координат із текстурними.
3. Виконати зворотній поворот кожного пікселя для визначення його координат для початкового положення полігону.
4. Виконати проекцію пікселя на екран та відтворити його колір на екрані.

Для того щоб повернути полігон, який буде паралельний екрану (площині xy), необхідно здійснити два послідовних повороти навколо координатних осей x і y [7].

Щоб визначити кути повороту, знайдемо нормальний вектор площини полігону ABC за трьома точками :

$$\begin{aligned}\bar{N}_x &= (B_y - A_y)(C_z - A_z) - (B_z - A_z)(C_y - A_y) = \\ &= A_y B_z - A_z B_y - A_y C_z + A_z C_y + B_y C_z - B_z C_y\end{aligned}$$

$$\begin{aligned}\bar{N}_y &= (B_z - A_z)(C_x - A_x) - (B_x - A_x)(C_z - A_z) = \\ &= A_z B_x - A_x B_z + A_x C_z - A_z C_x - B_x C_z + B_z C_x\end{aligned}$$

$$\begin{aligned}\bar{N}_z &= (B_x - A_x)(C_y - A_y) - (B_y - A_y)(C_x - A_x) = \\ &= A_x B_y - A_y B_x - A_x C_y + A_y C_x + B_x C_y - B_y C_x.\end{aligned}$$

де $\bar{N}_x, \bar{N}_y, \bar{N}_z$ - координати вектора нормалі \bar{N} , $A_x, A_y, A_z, B_x, B_y, B_z, C_x, C_y, C_z$ - координати вершин полігону ABC . Для визначення кута повороту α навколо осі x спроекуємо вектор \bar{N} на площину yz та обчислимо довжину проекції:

$$d = \sqrt{\bar{N}_y^2 - \bar{N}_z^2},$$

де d довжина проекції нормального вектора \bar{N} на площину yz ; \bar{N}_y, \bar{N}_z - координати вектора \bar{N} . Синус і косинус кута α знайдемо за формулами:

$$\cos \alpha = \frac{\bar{N}_z}{d}, \quad \sin \alpha = \frac{\bar{N}_y}{d}. \quad (1)$$

Після повороту навколо осі x , z -компонента одиничного вектора буде рівна d , x -компонента рівна \bar{N}_x . Довжина нормального вектора рівна 1. Таким чином синус і косинус кута β для повороту навколо осі y рівні [7]:

$$\cos \beta = d, \quad \sin \beta = \bar{N}_x. \quad (2)$$

Поворот можна подати у матричному виді:

$$M = T R_x [R_y] = T [R_{xy}], \quad (3)$$

де M - матриця координат полігону, повернутого паралельно до xy ; T - матриця початкових координат полігону, R_x - матриця повороту відносно осі x ; $[R_y]$ - матриця повороту відносно осі y . $[R_{xy}]$ - матриця послідовного повороту навколо осі x та y відповідно.

Усі перетворення виконуються в однорідних координатах [7]. Для переведення координат в однорідні кожній точці задається четверта компонента w , рівна 1. Для переходу від однорідних координат до звичайних тривимірних після виконання перетворень, знаходять частку від ділення кожної з компонент (x, y, z) на компоненту w .

Для реалізації перетворення повороту навколо осей x і y використовують такі матриці [7]:

$$\begin{aligned}
 [R_x] &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha & \sin \alpha & 0 \\ 0 & -\sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \\
 [R_y] &= \begin{bmatrix} \cos(-\beta) & 0 & -\sin(-\beta) & 0 \\ 0 & 1 & 0 & 0 \\ \sin(-\beta) & 0 & \cos(-\beta) & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} = \\
 &= \begin{bmatrix} \cos \beta & 0 & \sin \beta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \beta & 0 & \cos \beta & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}. \\
 \text{Визначимо перетворення } [R_{xy}] &: \\
 [R_{xy}] &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha & \sin \alpha & 0 \\ 0 & -\sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \\
 &\times \begin{bmatrix} \cos \beta & 0 & \sin \beta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin \beta & 0 & \cos \beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \\
 &= \begin{bmatrix} \cos \beta & 0 & \sin \beta & 0 \\ -\sin \alpha \sin \beta & \cos \alpha & \cos \beta \sin \alpha & 0 \\ -\cos \alpha \sin \beta & -\sin \alpha & \cos \alpha \cos \beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.
 \end{aligned}$$

Підставимо отримане значення у формулу (3):

$$\begin{aligned}
 &\begin{bmatrix} A'_x & A'_y & A'_z & A'_w \\ B'_x & B'_y & B'_z & B'_w \\ C'_x & C'_y & C'_z & C'_w \end{bmatrix} = \\
 &= \begin{bmatrix} A_x & A_y & A_z & A_w \\ B_x & B_y & B_z & B_w \\ C_x & C_y & C_z & C_w \end{bmatrix} \times \\
 &\times \begin{bmatrix} \cos \beta & 0 & \sin \beta & 0 \\ -\sin \alpha \sin \beta & \cos \alpha & \cos \beta \sin \alpha & 0 \\ -\cos \alpha \sin \beta & -\sin \alpha & \cos \alpha \cos \beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} =
 \end{aligned}$$

$$\begin{aligned}
 &\begin{bmatrix} A_x \cos \beta - & A_x \sin \beta + \\ -A_z \cos \cdot & +A_y \cos \beta \cdot \\ \cdot \alpha \sin \beta - & A_y \cos \alpha - & A_w \\ -A_y \sin \alpha \sin \beta & -A_z \sin \alpha & \cdot \sin \alpha + \\ & +A_z \cos \cdot & \\ & \cdot \alpha \cos \beta & \\ B_x \cos \beta - & B_x \sin \beta + \\ -B_z \cos \alpha \cdot & +B_y \cos \cdot \\ \cdot \sin \beta - & B_y \cos \alpha - & B_w \\ -B_y \sin \cdot & -B_z \sin \alpha & \cdot \beta \sin \alpha + \\ \cdot \alpha \sin \beta & +B_z \cos \cdot & \\ & \cdot \alpha \cos \beta & \\ C_x \cos \beta - & C_x \sin \beta + \\ -C_z \cos \cdot & +C_y \cos \beta \cdot \\ \cdot \alpha \sin \beta - & C_y \cos \alpha - & C_w \\ -C_y \sin \cdot & -C_z \sin \alpha & \cdot \sin \alpha + \\ \cdot \alpha \sin \beta & +C_z \cos \cdot & \\ & \cdot \alpha \cos \beta & \end{bmatrix}, \quad (4)
 \end{aligned}$$

де $A'_x, A'_y, A'_z, A'_w, B'_x, B'_y, B'_z, B'_w, C'_x, C'_y, C'_z, C'_w$ - однорідні координати вершин полігону ABC після виконання повороту; $A_x, A_y, A_z, A_w, B_x, B_y, B_z, B_w, C_x, C_y, C_z, C_w$ - однорідні координати вершин полігону ABC до виконання повороту. Таким чином, поворот точок, що лежать в одній довільній площині, у деяку іншу площину, паралельну площині xy , можна за формулами:

$$\begin{aligned}
 x' &= \frac{x \cos \beta - z \cos \alpha \sin \beta - y \sin \alpha \sin \beta}{w}, \\
 y' &= \frac{y \cos \alpha - z \sin \alpha}{w}, \\
 z' &= \frac{x \sin \beta + y \cos \beta \sin \alpha + z \cos \alpha \cos \beta}{w},
 \end{aligned} \quad (5)$$

де x', y', z' - координати точки після повороту в тривимірному просторі; x, y, z, w - однорідні координати точки до повороту.

Як видно із формули (4), w - компонента точки у результаті перетворення повороту не змінюється. Оскільки перетворення повороту виконуються до перспективних перетворень, а початкове значення w - компоненти при переході від тривимірних координат до однорідних встановлюють рівним 1, то формули (5) можна спростити:

$$\begin{aligned}x' &= \frac{x \cos \beta - z \cos \alpha \sin \beta - y \sin \alpha \sin \beta}{1} = \\ &= x \cos \beta - z \cos \alpha \sin \beta - y \sin \alpha \sin \beta, \\ y' &= \frac{y \cos \alpha - z \sin \alpha}{1} = y \cos \alpha - z \sin \alpha, \\ z' &= \frac{x \sin \beta + y \cos \beta \sin \alpha + z \cos \alpha \cos \beta}{1} = \\ &= x \sin \beta + y \cos \beta \sin \alpha + z \cos \alpha \cos \beta,\end{aligned}\quad (6)$$

Із формул (1) і (2) отримаємо такі залежності:

$$\left. \begin{aligned}\cos \alpha &= \frac{\bar{N}_z}{d} \\ \cos \beta &= d\end{aligned} \right\} \Rightarrow \cos \alpha \cos \beta = \frac{\bar{N}_z}{d} d = \bar{N}_z,$$

$$\left. \begin{aligned}\sin \alpha &= \frac{\bar{N}_z}{d} \\ \cos \beta &= d\end{aligned} \right\} \Rightarrow \sin \alpha \cos \beta = \frac{\bar{N}_y}{d} d = \bar{N}_y.\quad (7)$$

Таким чином, складність визначення z -компоненти при повороті можна знизити на одну операцію множення:

$$z' = x \sin \beta + y \cos \beta \sin \alpha + z \bar{N}_z,$$

де \bar{N}_z - z -компонента вектора нормалі полігону.

Оскільки у всіх точок, що належать площині, паралельній площині xu , z -координати рівні то обрахунок z -компоненти достатньо виконувати лише для однієї з вершин полігону.

Після повороту полігону, який стане паралельним площині xu , проводимо його текстурування з використанням техніки *texture mapping* [2, 3, 4, 6]. Оскільки z -координати усіх пікселів рівні, то рівень *texture pyramid* (LOD) визначається один раз для кожного полігону на відміну від текстурування в площині екрану, де LOD-рівень визначається окремо для кожного пікселя. Колір кожного пікселя записується до кадрового буфера з координатами, які відповідають екранній площині. Для визначення екранних координат пікселя необхідно виконати перетворення проєкції на площину xu . Оскільки була виконана процедура повороту перед виконанням перетворення проєкції, кожен піксель необхідно спочатку повернути в початкову площину. Для повороту пікселя в початкову площину необхідно виконати перетворення, зворотнє до $[R_{xy}]$. Враховуючи це, перетворення проєкції пікселя на екран можна виконати в матричному виді:

$$[M'] = M [R_{xy}]^{-1} P = M [PR_{xy}^{-1}],$$

де $[M']$ - матриця координат в екранній площині; M - матриця координат в площині

полігону, повернутого паралельно до xu ; $[R_{xy}]^{-1}$ - матриця обернена до матриці $[R_{xy}]$; P - матриця проєкції на екран; $[PR_{xy}^{-1}]$ - загальна матриця перетворення проєкції.

$[R_{xy}]^{-1}$ визначається шляхом добутку матриць, обернених до R_x і $[R_y]$ (R_x^{-1} і $[R_y]^{-1}$) в зворотному порядку:

$$[R_{xy}]^{-1} = [R_y]^{-1} R_x^{-1}.$$

Зворотні перетворення повороту навколо осей x і y мають вигляд:

$$R_x^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha & 0 \\ 0 & \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

$$[R_y]^{-1} = \begin{bmatrix} \cos \beta & 0 & -\sin \beta & 0 \\ 0 & 1 & 0 & 0 \\ \sin \beta & 0 & \cos \beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Знайдемо перетворення $[R_{xy}]^{-1}$:

$$[R_{xy}]^{-1} = \begin{bmatrix} \cos \beta & 0 & -\sin \beta & 0 \\ 0 & 1 & 0 & 0 \\ \sin \beta & 0 & \cos \beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha & -\sin \alpha & 0 \\ 0 & \sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} \cos \beta & -\sin \alpha \sin \beta & -\cos \alpha \sin \beta & 0 \\ 0 & \cos \alpha & -\sin \alpha & 0 \\ \sin \beta & \cos \beta \sin \alpha & \cos \alpha \cos \beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Знайдемо загальну матрицю перетворення проєкції:

$$[R_{xy}]^{-1} = \begin{bmatrix} \cos \beta & -\sin \alpha \sin \beta & -\cos \alpha \sin \beta & 0 \\ 0 & \cos \alpha & -\sin \alpha & 0 \\ \sin \beta & \cos \beta \sin \alpha & \cos \alpha \cos \beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} A & B & C & D \\ E & F & G & H \\ I & J & K & L \\ M & N & O & P \end{bmatrix} =$$

$$= \begin{bmatrix} A \cos \beta - B \cos \beta - C \cos \beta - D \cos \alpha - \\ -I \cos \alpha \cdot \sin \beta - \\ -E \sin \alpha \cdot \sin \beta \\ E \cos \alpha - F \cos \alpha - G \cos \alpha - H \cos \alpha - \\ -I \sin \alpha \\ A \cos \beta - B \sin \beta + C \sin \beta + D \sin \beta + \\ -E \cos \beta \cdot \sin \alpha - \\ -I \cos \alpha \cdot \cos \beta \\ M \\ N \\ O \\ P \end{bmatrix}, \begin{bmatrix} M + A \cos \beta \cdot x + E \cos \alpha \cdot y + A \sin \beta \cdot z - \\ -I \sin \alpha \cdot y + I \cos \alpha \cos \beta \cdot z + E \cos \beta \sin \alpha \cdot z - \\ -I \cos \alpha \sin \beta \cdot x - E \sin \alpha \sin \beta \cdot x \\ N + B \cos \beta \cdot x + F \cos \alpha \cdot y + \beta \cdot \sin \beta \cdot z - \\ -J \sin \alpha \cdot y + J \cos \alpha \cos \beta \cdot z + F \cos \beta \sin \alpha \cdot z - \\ -J \cos \alpha \sin \beta \cdot x - F \sin \alpha \sin \beta \cdot x \\ O + C \cos \beta \cdot x + G \cos \alpha \cdot y + C \sin \beta \cdot z - \\ -K \sin \alpha \cdot y + K \cos \alpha \cos \beta \cdot z + G \cos \beta \sin \alpha \cdot z - \\ -K \cos \alpha \sin \beta \cdot x - G \sin \alpha \sin \beta \cdot x \\ P + D \cos \beta \cdot x + H \cos \alpha \cdot y + D \sin \beta \cdot z - \\ -L \sin \alpha \cdot y + L \cos \alpha \cos \beta \cdot z + H \cos \beta \sin \alpha \cdot z - \\ -L \cos \alpha \sin \beta \cdot x - H \sin \alpha \sin \beta \cdot x \end{bmatrix},$$

де $A - P$ - параметри матриці проєкції.

Обчислимо координати проєкції:

$$x' \ y' \ z' \ w' = x \ y \ z \ 1 \times$$

$$\times \begin{bmatrix} A \cos \beta - B \cos \beta - C \cos \beta - D \cos \alpha - \\ -I \cos \alpha \cdot \sin \beta - \\ -E \sin \alpha \cdot \sin \beta \\ E \cos \alpha - F \cos \alpha - G \cos \alpha - H \cos \alpha - \\ -I \sin \alpha \\ A \cos \beta - B \sin \beta + C \sin \beta + D \sin \beta + \\ -E \cos \beta \cdot \sin \alpha - \\ -I \cos \alpha \cdot \cos \beta \\ M \\ N \\ O \\ P \end{bmatrix} =$$

де x', y', z', w' - однорідні координати пікселя в екранній площині; x, y, z - координати пікселя в площині, паралельній xy .

Знайдемо тривимірні координати пікселя:

$$x'' = \frac{M + A \cos \beta \cdot x + E \cos \alpha \cdot y + A \sin \beta \cdot z - I \sin \alpha \cdot y + I \cos \alpha \cos \beta \cdot z + E \cos \beta \sin \alpha \cdot z - I \cos \alpha \sin \beta \cdot x - E \sin \alpha \sin \beta \cdot x}{P + D \cos \beta \cdot x + H \cos \alpha \cdot y + D \sin \beta \cdot z - L \sin \alpha \cdot y + L \cos \alpha \cos \beta \cdot z + H \cos \beta \sin \alpha \cdot z - L \cos \alpha \sin \beta \cdot x - H \sin \alpha \sin \beta \cdot x},$$

$$y'' = \frac{N + B \cos \beta \cdot x + F \cos \alpha \cdot y + \beta \cdot \sin \beta \cdot z - J \sin \alpha \cdot y + J \cos \alpha \cos \beta \cdot z + F \cos \beta \sin \alpha \cdot z - J \cos \alpha \sin \beta \cdot x - F \sin \alpha \sin \beta \cdot x}{P + D \cos \beta \cdot x + H \cos \alpha \cdot y + D \sin \beta \cdot z - L \sin \alpha \cdot y + L \cos \alpha \cos \beta \cdot z + H \cos \beta \sin \alpha \cdot z - L \cos \alpha \sin \beta \cdot x - H \sin \alpha \sin \beta \cdot x},$$

$$z'' = \frac{O + C \cos \beta \cdot x + G \cos \alpha \cdot y + C \sin \beta \cdot z - K \sin \alpha \cdot y + K \cos \alpha \cos \beta \cdot z + G \cos \beta \sin \alpha \cdot z - K \cos \alpha \sin \beta \cdot x - G \sin \alpha \sin \beta \cdot x}{P + D \cos \beta \cdot x + H \cos \alpha \cdot y + D \sin \beta \cdot z - L \sin \alpha \cdot y + L \cos \alpha \cos \beta \cdot z + H \cos \beta \sin \alpha \cdot z - L \cos \alpha \sin \beta \cdot x - H \sin \alpha \sin \beta \cdot x}$$

Зменшимо кількість операцій множення, підставивши у формули значення з формул (7):

$$x'' = \frac{M + A \cos \beta \cdot x + E \cos \alpha \cdot y + A \sin \beta \cdot z - I \sin \alpha \cdot y + I \bar{N}_z \cdot z + E \bar{N}_y \cdot z - I \bar{N}_y \cdot x - E \sin \alpha \sin \beta \cdot x}{P + D \cos \beta \cdot x + H \cos \alpha \cdot y + D \sin \beta \cdot z - L \sin \alpha \cdot y + L \bar{N}_z \cdot z + H \bar{N}_y \cdot z - L \bar{N}_y \cdot x - H \sin \alpha \sin \beta \cdot x},$$

$$y'' = \frac{N + B \cos \beta \cdot x + F \cos \alpha \cdot y + \beta \cdot \sin \beta \cdot z - J \sin \alpha \cdot y + J \bar{N}_z \cdot z + F \bar{N}_y \cdot z - J \bar{N}_y \cdot x - F \sin \alpha \sin \beta \cdot x}{P + D \cos \beta \cdot x + H \cos \alpha \cdot y + D \sin \beta \cdot z - L \sin \alpha \cdot y + L \bar{N}_z \cdot z + H \bar{N}_y \cdot z - L \bar{N}_y \cdot x - H \sin \alpha \sin \beta \cdot x}$$

$$z'' = \frac{O + C \cos \beta \cdot x + G \cos \alpha \cdot y + C \sin \beta \cdot z - K \sin \alpha \cdot y + K \bar{N}_z \cdot z + G \bar{N}_y \cdot z - K \cos \alpha \sin \beta \cdot x - G \sin \alpha \sin \beta \cdot x}{P + D \cos \beta \cdot x + H \cos \alpha \cdot y + D \sin \beta \cdot z - L \sin \alpha \cdot y + L \bar{N}_z \cdot z + H \bar{N}_y \cdot z - L \bar{N}_y \cdot x - H \sin \alpha \sin \beta \cdot x}$$

Найпростіша матриця одноточкової перспективної проекції має вигляд [7]:

$$P = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & r \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Значення параметру r приймають рівним $-\frac{1}{z_c}$, де

z_c - відстань від камери до площини проекції [7].

Після підстановки значень у формулу, отримуємо:

$$x'' = \frac{\cos \beta \cdot x + \sin \beta \cdot z}{1 - r \sin \alpha \cdot y + r \bar{N}_z \cdot z - r \bar{N}_y \cdot x},$$

$$y'' = \frac{\cos \alpha \cdot y + \bar{N}_y \cdot z - \sin \alpha \sin \beta \cdot x}{1 - r \sin \alpha \cdot y + r \bar{N}_z \cdot z - r \bar{N}_y \cdot x},$$

$$z'' = \frac{\bar{N}_z \cdot z - \sin \alpha \cdot y - \cos \alpha \sin \beta \cdot x}{1 - r \sin \alpha \cdot y + r \bar{N}_z \cdot z - r \bar{N}_y \cdot x},$$

У результаті обчислень знаходять координати пікселя у вигляді чисел з плаваючою комою. Такі координати не можна однозначно віднести до конкретного пікселя на екрані, оскільки просте округлення до цілого призводить також до появи артефактів, характерних для текстуровання в площині текстури (накладання пікселів і поява дір) (рис. 3). Необхідно врахувати колір текселя для усіх екранних пікселів, які покриває його проекція (рис. 4).

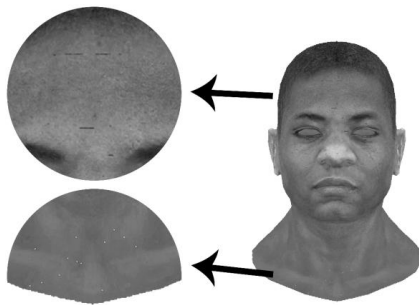


Рисунок 3 – Артефакти при текстурованні описаним методом без фільтрації в площині екрану

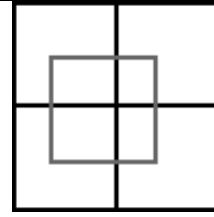


Рисунок 4 – Положення проекції пікселя полігону на екранну площину відносно пікселів екрану

Координати пікселів, що покриває проекція текселя, знаходяться за формулами:

$$x_1 = \lfloor x \rfloor, y_1 = \lfloor y \rfloor,$$

$$x_2 = \lfloor x \rfloor + 1, y_2 = \lfloor y \rfloor + 1,$$

$$x_3 = \lfloor x \rfloor + 1, y_3 = \lfloor y \rfloor,$$

$$x_4 = \lfloor x \rfloor, y_4 = \lfloor y \rfloor + 1.$$

При виконанні проекції текселів із текстурного простору в екранний можливе накладання проекцій кількох текселів на один піксель. Значення кольорів усіх текселів що накладаються на один піксель перед виводом на екран, необхідно усереднити з урахуванням вагових коефіцієнтів. У якості вагового коефіцієнта використаємо площу текселя, що покриває піксель. Площі знаходимо за формулами:

$$S_1 = 1 - x + \lfloor x \rfloor \quad 1 - y + \lfloor y \rfloor,$$

$$S_2 = x - \lfloor x \rfloor \quad y - \lfloor y \rfloor,$$

$$S_3 = x - \lfloor x \rfloor \quad 1 - y + \lfloor y \rfloor,$$

$$S_4 = 1 - x + \lfloor x \rfloor \quad y - \lfloor y \rfloor,$$

Колір пікселя на екрані визначаємо за формулою:

$$I = \frac{\sum_{i=1}^n S_i \cdot I_i}{\sum_{i=1}^n S_i}.$$

Текстуровання в просторі об'єкта забезпечує вищу точність обчислення кольорів екранних пікселів за рахунок врахування глибини простору та збереження геометричних властивостей полігону. Обчислювальні витрати на визначення LOD рівня істотно нижчі порівняно з текстурованням в площині екрану за рахунок того, що LOD рівень визначається один раз на кожен полігон, а не один раз на кожен піксель екрану. Метод не потребує виконання просторово варіативних фільтрацій. Фільтрація в площині екрану істотно продуктивніша порівняно з анізотропною, оскільки не потребує виконання трудомістких операцій по визначенню розмірів та положення еліпса і координат пікселів, що його формують. Формули, необхідні для виконання усіх поворотів і проекцій, мають вищу сумарну складність, порівняно з формулами проекцій для текстуровання в просторі екрану, проте

відсутність потреби виконання анізотропної фільтрації, дає змогу виконання паралельних обчислень без втрати якості. При цьому може застосовуватись окремий потік для кожного пікселя полігону при виконанні операцій визначення текстурних координат і зворотної проекції що дає змогу істотно підвищити продуктивність обчислень. Результат роботи методу відображено на рисунку 5.



Рисунок 5 – Результат роботи методу текстурування в просторі об'єкту з фільтрацією в площині екрану.

Висновки

Класичні методи текстурування не забезпечують високої точності визначення кольорів, оскільки не враховують глибини простору та пропорцій полігонів, а тому потребують виконання складної процедури фільтрації.

Запропоновано метод текстурування в просторі об'єкта, що дозволяє підвищити точність визначення кольорів, забезпечує зменшення обсягу обчислень на фільтрацію та дозволяє підвищити продуктивність шляхом використання паралельних обчислень. При цьому не має місце втрата реалістичності

Список літератури

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. Paul S. Heckbert. Survey of texture mapping // [Електронний ресурс]. — Режим доступу: <https://www.cs.cmu.edu/~ph/textsurv.pdf>.
3. Романюк О.Н. Аналіз методів анізотропної фільтрації текстур / О. Н. Романюк, С. О. Романюк, М. П. Піддубецька // Вимірювальна та обчислювальна техніка в технологічних процесах. — 2013. — № 2. — С. 123-128.
4. Романюк О.Н. Аналіз методів фільтрації текстур / О.Н. Романюк, О.О. Дудник // Міжнародна науково-практична Інтернет-конференція «Молодь в технічних науках: дослідження, проблеми, перспективи» — Вінниця — 2015. - Режим доступу: http://conf.inmad.vntu.edu.ua/fm/files/administrator_materials_1429512871.doc
5. Pavlos Mavridis. High Quality Elliptical Texture Filtering on GPU / Pavlos Mavridis, Georgios Papaioannou // [Електронний ресурс]. — Режим доступу: http://www.pmavridis.com/data/I3D11_EllipticalFiltering.pdf
6. Вяткин С.И. Анизотропная фильтрация текстуры в реальном времени / С.И. Вяткин, А.Н. Романюк, А.А. Дудник // Вимірювальна та Обчислювальна Техніка в Технологічних Процесах № 4' 2015 (53) 219
7. Роджерс Д. Математические основы машинной графики / Роджерс Д., Адамс Дж. // Мир - Москва 2001р.

Надійшла до редакції 22.10.2016

А.Н. РОМАНЮК, А.А. ДУДНИК

Винницкий национальный технический университет

ПОВЫШЕНИЕ ПРОИЗВОДИТЕЛЬНОСТИ ТЕКСТУРИРОВАНИЯ С ВЫПОЛНЕНИЕМ ПРОЦЕДУРНЫХ ОПЕРАЦИЙ В ОБЪЕКТНОМ ПРОСТРАНСТВЕ

Предлагается метод повышения реалистичности формирования графических изображений, который устраняет артефакты, характерные для методов текстурирования в экранной плоскости. Метод предусматривает выполнение текстурирования в пространстве объекта с фильтрацией в плоскости экрана. При этом обеспечивается возможность распараллеливания процесса текстурирования.

Ключевые слова: *текстурирование, фильтрация текстур, анизотропная фильтрация.*

O.N. ROMANYUK, O.O. DUDNYK

Vinnitsia National Technical University

THE METHOD OF INCREASING PRODUCTIVITY OF TEXTURE MAPPING WITH IMPLEMENTATION OF PROCEDURAL OPERATIONS IN OBJECT SPACE

We propose a method to increase the realism of formation of graphic images, which eliminates artifacts typical of the methods of texturing in the display plane. The method provides object texturing in space with filtering in the plane of the screen. This provides the possibility of parallelization of the texturing process.

Keywords: *texturing, texture mapping, texture filtering, anisotropic filtering.*