

АНТИАЛІАЙЗИНГ ЗОБРАЖЕННЯ КРИВИХ ДРУГОГО ПОРЯДКУ, ЗАДАНИХ ЗАГАЛЬНИМ РІВНЯННЯМ

Запропоновано новий підхід до антиаліайзингу кривих другого порядку за умови, що вони задані загальним рівнянням. Метод характеризується простотою апаратної реалізації.

Вступ

На сучасному етапі розвитку комп'ютерної графіки особлива увага приділяється підвищенню реалістичності синтезованих зображень. Одним із факторів, що суттєво впливають на реалістичність зображення, є ефект аліайзингу [1], який спричинений недостатньою роздільною здатністю пристроїв відображення та проявляється у вигляді чітко виражених „зубців” на краях графічних об'єктів.

Існуючі методи антиаліайзингу характеризуються достатньо великими обчислювальними витратами та складністю апаратної реалізації, що обумовлює необхідність розробки методів та засобів згладжування, які б дозволили ефективно застосовувати процедури згладжування в широкому спектрі графічних продуктів.

Аналіз методів і постановка задачі

Найбільш поширені методи антиаліайзингу можна розділити на дві групи [1]. Першу групу складають методи, які базуються на збільшенні дискретизації зображення. Суть даних методів полягає в тому, що зображення обчислюється з високою розподільною здатністю, а при відображенні зменшується шляхом усереднення [1]. Ці методи характеризуються низькою швидкістю, оскільки при збільшенні дискретизації в n разів, кількість пікселів, а отже, і кількість обчис-

лень на один піксель, збільшується в n^2 разів [2].

У методах другої групи піксель розглядається не як умовна точка, а як скінченна ділянка [3]. Найбільшого поширення отримала модель, у якій піксель розглядається як квадрат зі стороною, що дорівнює одиниці [2]. Центр квадрата співпадає з центром пікселя. Інтенсивність кольору пікселя для даної моделі встановлюється пропорційно до площі тієї частини квадрата, яка покривається графічним примітивом. Більшість існуючих методів антиаліайзингу використовують зазначену модель, оскільки вона має порівняно малу обчислювальну складність [4].

У комп'ютерній графіці при формуванні графічних зображень широкого поширення отримали криві другого порядку, які, як правило, задаються у стандартній формі - рівняннями відносно координатних осей. Питання антиаліайзингу кривих другого порядку в стандартній формі розглядалося в працях [5,6,7,8].

У засобах комп'ютерної графіки найбільшого поширення отримали алгоритми Піттуея і Ву. Алгоритм антиаліайзингу Піттуея [9] розглядає криві другого порядку в загальній формі, але використовує спрощенні вирази для обчислення інтенсивностей кольору точок. Даний алгоритм не містить довгих операцій у циклі інтерполювання та має відносно просту апаратну реалізацію, однак забезпечує порівняно низьку якість

згладження, оскільки використовує тільки 4 рівня інтенсивності кольору.

У роботі [10] для антиаліазингу кривих другого порядку у загальній формі був адаптований метод антиаліазингу Ву [8], суть якого полягає в тому, що інтенсивність кольору пікселя траєкторії кола (дуги) розподіляється між двома суміжними пікселями, що знаходяться безпосередньо над і під ідеальною дугою, пропорційно до відстані від центра пікселя до точки, у якій дуга перетинає координатну решітку. Сума інтенсивностей цих двох пікселів завжди дорівнює максимальній.

Метод передбачає виконання операції ділення в циклі інтерполювання, що призводить до великих обчислювальних витрат та ускладнює апаратну реалізацію. Крім того, при визначенні інтенсивності кольору можливі «колізії» [10], коли одна точка координатного простору може розглядатися декілька разів при розрахунках та отримує різні інтенсивності кольору. «Колізії» можуть призводити до артефактів у зображеннях.

Мета статті - розробка нового підходу до антиаліазингу кривих другого порядку, який характеризується простотою обчислювального процесу та відсутністю артефактів при згладжуванні.

Розробка підходу до антиаліазингу

Для антиаліазингу границь об'єктів, обмежених кривими другого порядку, пропонується використати метод додаткових оцінювальних функцій (ОФ).

В якості базового методу формування траєкторії кривої другого порядку використаємо метод, запропонований у [10]. Вибір саме цього методу обумовлений тим, що він є універсальним для формування кривих другого порядку та має більш високу швидкодію порівняно з іншими [8,9].

При формуванні крокової траєкторії використовується така оцінювальна функція:

$$f(x, y) = Ax^2 + Bxy + Cy^2 + Dx + Ey = 0, \quad (1)$$

де $\{A, B, C, D, E\}$ - цілі числа.

Формула (1) задає криві другого порядку, які проходять через початок координат. Таке завдання кривої є найбільш поширеним в комп'ютерній графіці [10].

У загальному випадку при інкрементному формуванні траєкторій від поточної точки можливі вісім напрямків елементарних переміщень (чотири ортогональних і 4 під кутом, кратним 45°). Вибір наступних точок для переміщення для конкретного випадку вибирається залежно від квадранту, в якому розташовано сегмент кривої, і напрямку формування кривої. Так, наприклад, на рис. 1 зображено процедуру формування кривої проти годинникової стрілки у першому квадранті. Якщо на певному кроці обрано точку $P(x, y)$, то наступна точка крокової траєкторії обирається з точок $\{A, B, C\}$ (рис. 1). При цьому оцінювальна функція (1) обчислюється у точках $\{m_1, m_2, m_3, m_4\}$ (рис. 1).

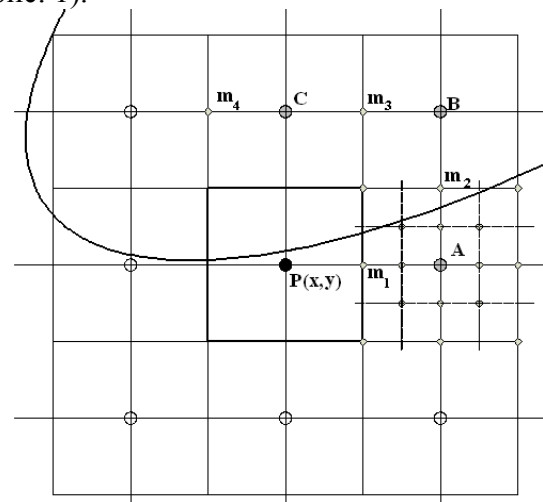


Рис. 1. Формування крокової траєкторії

Для антиаліазингу траєкторії кривої інтенсивність кольору пікселів будемо встановлювати пропорційно до тієї його

площі, яка покривається графічним примітивом (площі покриття). Площу покриття будемо обчислювати згідно з методом додаткових ОФ, який відрізняється від методу суперсемплінга введенням додаткових точок тільки вздовж траєкторії контурів, а не для всього зображення.

Розрахуємо значення ОФ у додаткових точках $\{T_1, T_2, \dots, T_{16}\}$, які розташовано, як показано на рис. 2.

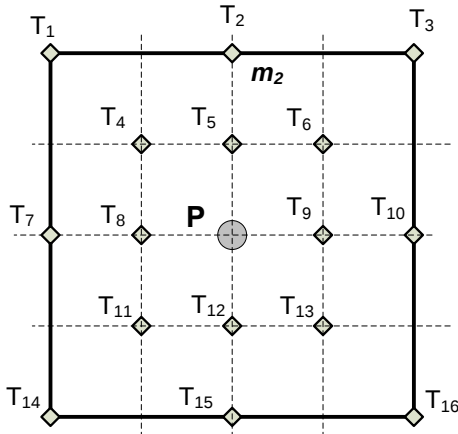


Рис. 2 Розташування точок $T_1 \div T_{16}$ у середині пікселя

Інтенсивність кольору пікселя знайдемо за формулою

$$I = I_M \frac{\sum_{i=1}^{16} \text{sign}(F_i)}{16}, \quad (2)$$

де I_M - інтенсивність кольору, з якою треба відтворити примітив; F_i - значення оцінювальної функції (1) у точці T_i .

Розглянемо вирази для знаходження оцінювальних функцій у точках $\{T_1, T_2, \dots, T_{16}\}$ та оцінимо їх обчислювальну складність.

Для точки T_1 маємо таке співвідношення:

$$\begin{aligned} F_1 &= f\left(x - \frac{1}{2}, y + \frac{1}{2}\right) = A \left(x - \frac{1}{2}\right)^2 + \\ &+ B \left(x - \frac{1}{2}\right) \left(y + \frac{1}{2}\right) + C \left(y + \frac{1}{2}\right)^2 + \\ &+ D \left(x - \frac{1}{2}\right) + E \left(y + \frac{1}{2}\right) = \\ &= (Ax^2 + Bx y + Cy^2 + Dx + Ey) - \\ &- \left(\frac{2Ax}{2} + \frac{By}{2} + \frac{D}{2}\right) + \\ &+ \left(\frac{2Cy}{2} + \frac{Bx}{2} + \frac{E}{2}\right) + \frac{A}{4} + \frac{C}{4} - \frac{B}{4}. \end{aligned}$$

Позначимо:

$$f_x = \frac{\partial f}{\partial x} = 2Ax + By + D;$$

$$f_y = \frac{\partial f}{\partial y} = 2Cy + Bx + E.$$

Отримуємо такий вираз для ОФ у точці T_1

$$\begin{aligned} F_1 &= f\left(x - \frac{1}{2}, y + \frac{1}{2}\right) = f(x, y) - \\ &- \frac{f_x}{2} + \frac{f_y}{2} + \frac{A}{4} + \frac{C}{4} - \frac{B}{4}. \end{aligned}$$

Аналогічним чином отримуємо вирази для оцінювальних функцій у точках $\{T_2, T_3, \dots, T_{16}\}$:

$$F_2 = f\left(x, y + \frac{1}{2}\right) = f(x, y) + \frac{f_y}{2} + \frac{C}{4};$$

$$\begin{aligned} F_3 &= f\left(x + \frac{1}{2}, y + \frac{1}{2}\right) = f(x, y) - \\ &- \frac{f_x}{2} + \frac{f_y}{2} + \frac{A}{4} + \frac{C}{4} + \frac{B}{4}. \end{aligned}$$

$$\begin{aligned} F_4 &= f\left(x - \frac{1}{4}, y + \frac{1}{4}\right) = f(x, y) - \\ &- \frac{f_x}{4} + \frac{f_y}{4} + \frac{A}{16} + \frac{C}{16} - \frac{B}{16}. \end{aligned}$$

$$F_5 = f\left(x, y + \frac{1}{4}\right) = f(x, y) + \frac{f_y}{4} + \frac{C}{16}.$$

$$F_6 = f\left(x + \frac{1}{4}, y + \frac{1}{4}\right) = f(x, y) + \frac{f_x}{4} + \frac{f_y}{4} + \frac{A}{16} + \frac{C}{16} + \frac{B}{16}.$$

$$F_7 = f\left(x - \frac{1}{2}, y\right) = f(x, y) - \frac{f_x}{2} + \frac{A}{4}.$$

$$F_8 = f\left(x - \frac{1}{4}, y\right) = f(x, y) - \frac{f_x}{4} + \frac{A}{16}.$$

$$F_9 = f\left(x + \frac{1}{4}, y\right) = f(x, y) + \frac{f_x}{4} + \frac{A}{16}.$$

$$F_{10} = f\left(x + \frac{1}{2}, y\right) = f(x, y) + \frac{f_x}{2} + \frac{A}{4}.$$

$$F_{11} = f\left(x - \frac{1}{4}, y - \frac{1}{4}\right) = f(x, y) - \frac{f_x}{4} - \frac{f_y}{4} + \frac{A}{16} + \frac{C}{16} + \frac{B}{16}.$$

$$F_{12} = f\left(x, y - \frac{1}{4}\right) = f(x, y) - \frac{f_y}{4} + \frac{C}{16}.$$

$$F_{13} = f\left(x + \frac{1}{4}, y - \frac{1}{4}\right) = f(x, y) + \frac{f_x}{4} - \frac{f_y}{4} + \frac{A}{16} + \frac{C}{16} - \frac{B}{16}.$$

$$F_{14} = f\left(x - \frac{1}{2}, y - \frac{1}{2}\right) = f(x, y) - \frac{f_x}{2} - \frac{f_y}{2} + \frac{A}{4} + \frac{C}{4} + \frac{B}{4}.$$

$$F_{15} = f\left(x, y - \frac{1}{2}\right) = f(x, y) - \frac{f_y}{2} + \frac{C}{4}.$$

$$F_{16} = f\left(x + \frac{1}{2}, y - \frac{1}{2}\right) = f(x, y) + \frac{f_x}{2} - \frac{f_y}{2} + \frac{A}{4} + \frac{C}{4} - \frac{B}{4}.$$

Як видно з отриманих формул, для обчислення значень оцінювальних функцій у додаткових точках $\{T_1, T_3, T_4, T_6, T_{11}, T_{13}, T_{14}, T_{16}\}$ необхідно виконати по 3 операції типу «додавання» на одну точку (з урахуванням того, що значення коефіцієнтів $(A/4 + C/4 - B/4)$, $(A/4 + C/4 + B/4)$, $(A/16 + C/16 - B/16)$, $(A/16 + C/16 + B/16)$ розраховується в циклі підготування).

Слід зазначити, що значення виразів f_x та f_y використовуються у базовому методу формування крокової траєкторії, отже їх не потрібно обчислювати додатково.

Для обчислення значень ОФ у додаткових точках $\{T_2, T_5, T_7, T_8, T_9, T_{10}, T_{12}, T_{15}\}$ необхідно виконати по 2 операції типу «додавання» на одну точку.

Залежно від вимог, які висуваються до алгоритму формування траєкторії, можливо використання для розрахунку інтенсивності кольору різної кількості додаткових точок у декількох комбінаціях. У таблиці 1 наведено основні з них, а також відповідна кількість операцій необхідна для обчислення значень ОФ.

При використанні 16 додаткових точок існує можливість зменшити загальні обчислювальні витрати на обчислення ОФ. Перепишемо вираз для знаходження значення F_1 у вигляді:

$$F_1 = \left[f(x, y) + \frac{f_y}{2} + \frac{C}{4} \right] - \frac{f_x}{2} + \frac{A}{4} - \frac{B}{4}.$$

Таблиця 1. Кількість операцій для обчислення ОФ у додаткових точках

Кількість додаткових точок	Додаткові точки	Кількість операцій типу «додавання»
4	$\{T_4, T_6, T_{11}, T_{13}\}$	$4 \times 3 = 12$
	$\{T_1, T_3, T_{14}, T_{16}\}$	$4 \times 3 = 12$
	$\{T_5, T_8, T_9, T_{12}\}$	$4 \times 2 = 8$
	$\{T_2, T_7, T_{10}, T_{15}\}$	$4 \times 2 = 8$
8	$\{T_1, T_3, T_4, T_6, T_{11}, T_{13}, T_{14}, T_{16}\}$	$8 \times 3 = 24$
	$\{T_2, T_4, T_6, T_7, T_{10}, T_{11}, T_{13}, T_{15}\}$	$4 \times 2 + 4 \times 3 = 20$
	$\{T_2, T_5, T_7, T_8, T_9, T_{10}, T_{12}, T_{15}\}$	$8 \times 2 = 16$
16	$\{T_1, T_2, T_3, T_4, T_5, T_6, T_7, T_8, T_9, T_{10}, T_{11}, T_{12}, T_{13}, T_{14}, T_{15}, T_{16}\}$	$8 \times 2 + 8 \times 3 = 40$

Із урахуванням співвідношення для F_2 , отримуємо:

$$F_1 = F_2 - \frac{f_x}{2} + \left[\frac{A}{4} - \frac{B}{4} \right].$$

При використанні останнього співвідношення для обчислення ОФ у точці T_1 необхідно виконати 2 операції типу «додавання». Аналогічні співвідношення можна отримати для точок $\{T_3, T_4, T_6, T_{11}, T_{13}, T_{14}, T_{16}\}$. При використанні таких співвідношень кількість операцій необхідних для обчислення 16 додаткових ОФ складає 32 операції типу додавання.

Як видно з рис. 2 для випадку, коли обирається піксел A :

$$F_2 = d_2; F_7 = d_1,$$

де d_1, d_2 - значення ОФ відповідно у точці m_1 та m_2 (обчислюється у базовому алгоритмі для формування крокової траєкторії);

Аналогічно для випадку, коли обирається піксел B : $F_{15} = d_2; F_7 = d_3$. Для пікселя C маємо наступне співвідношення: $F_{10} = d_3; F_7 = d_4$.

Таким чином, для кожного випадку значення двох додаткових ОФ відомо, що

дозволяє зменшити обчислювальні витрати на 4 операції додавання.

Отже з урахуванням усіх оптимізацій обчислювальні витрати на розрахунок 16 додаткових оцінювальних функцій складають 28 операцій типу додавання.

Для порівняння витрати алгоритму [10], що розглядався нами раніше, складають 1 операція ділення та 4 операції типу «додавання».

З наведених співвідношень для оцінювальних функцій випливає, що розрахунок значень оцінювальних функцій у додаткових точках можливо виконувати незалежно, що забезпечує високий ступінь паралелізму обчислень і дозволяє значно підвищити швидкодію формування зображень у багатопроекторних системах.

Значення оцінювальних функцій у додаткових точках залежать лише від координат точок. Тому незалежно від напрямку рендерингу крокової траєкторії інтенсивність кольору конкретного пікселя завжди буде однаковою. Згідно ж методу Ву інтенсивність кольору пікселя обчислюється з урахуванням відстані від центра пікселя до крокової траєкторії по одній з координат залежно від потокового квадранту та напрямку рендерингу. Тому можливий випадок, коли один і той же піксел буде двічі вибрано при формуванні крокової траєкторії і він отримає різну інтенсивність кольору [10], що створює колізію при антиаліазингу.

Приклад зображень сформованих із використанням запропонованого алгоритму наведено на рис. 3.

Висновки

У роботі поширено метод додаткових оцінювальних функцій на загальний випадок кривих другого порядку.

Метод не містить «довгих» операцій у циклі інтерполювання і має відносно просту апаратну реалізацію. Незалежне формування оцінювальних функцій дає можливість розпаралелити обчислювальний процес.

Список літератури

1. Роджерс Д. Алгоритмические основы машинной графики. Пер. с англ. – М.: Мир, 1989. – 512 с.
2. William J. Leler. Human Vision, Anti-aliasing, and the cheap 4000 Line Display // ACM. – 1980. – Vol. 14. – P.308–313.
3. Crow, Franklin C. A Comparison of Antialiasing Techniques. // IEEE CG & A. – 1981. – Vol. 1. – p. 40–47.

4. S. Gupta and R. F. Sproull. Filtering edges for gray-scale displays. // Computer Graphics. – 1981. – Vol. 15. – No. 3. – P. 1–5.

5. Kappel M. R. An Ellipse-Drawing Algorithm for Raster Displays // Proceedings of Computer Graphics. – 1985. – P. 110-134.

6.D.A. Field. Algorithms for Drawing Anti-aliased Circles and Ellipses. // Computer Vision Graphics and Image Processing. – 1986. – Vol. 33. – P. 1-15.

7. Agathos A., Theoharis T. Efficient integer algorithms for the generation of conic sections. // Computer & Graphics. – 1998. – Vol. 22. – No. 5. – P. 621-628.

8. Wu X. An Efficient Antialiasing Technique. // Computer Graphics. – 1991. – Vol. 25. – No. 4. – P. 143–152.

9. M.L.V. Pitteway and E. Banissi. Soft Edging Fonts. // Proceedings of Computer Graphics 87. – 1987. – P. 133-154.

10. GOLIPOUR M. General Rendering and Antialiasing Algorithms for Conic Sections. PhD thesis, London South Bank University, May 2005.

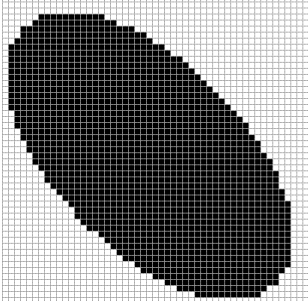
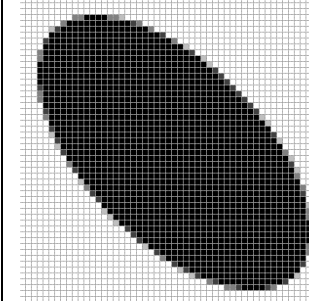
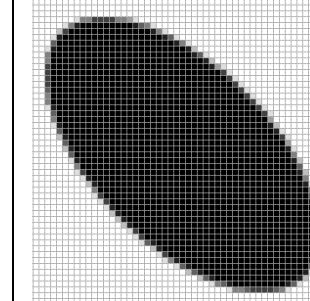
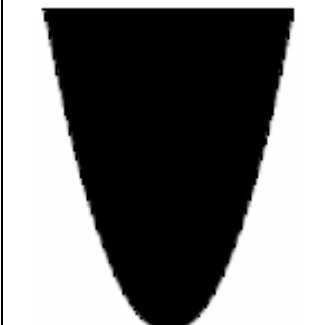
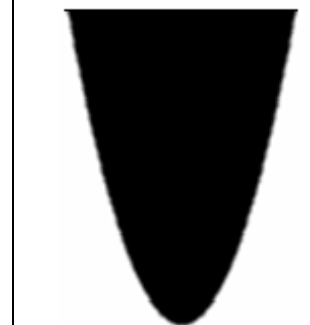
			$A = 45; B = 52;$ $C = 45; D = 300;$ $E = 1555$
			$A = 1; B = 0;$ $C = 0; D = 1;$ $E = -12$
Без антиаліазингу	Із використанням 8-ми додаткових точок	Із використанням 16-ти додаткових точок	

Рис.3. Приклади сформованих зображень

