

УДК 004.92:655.3

*Чінкує Камелія, здобувач*

*Інститут поліграфії та медійних технологій НУ «Львівська  
політехніка»*

## **АЛГОРИТМИ АВТОМАТИЧНОЇ РАСТЕРИЗАЦІЇ ВЕКТОРНОЇ ГРАФІКИ В ЗАДАЧАХ ПОЛІГРАФІЧНОГО ДРУКУ**

Растрезизація векторної графіки важливий етап підготовки друку, що визначає якість відтворення контурів при роздільній здатності 1200-2400 dpi. На відміну від екранного рендерингу (96 dpi), поліграфічний друк не допускає видимого аліасингу - артефактів у вигляді «сходинок» на діагональних лініях і розмитих країв на дрібних елементах. Це висуває принципово інші вимоги до точності алгоритмів перетворення, ніж у задачах комп'ютерної графіки загального призначення. Більшість векторних контурів у форматах PDF, SVG і PostScript описується кривими Без'є третього порядку:  $P(t) = (1-t)^3P_0 + 3(1-t)^2tP_1 + 3(1-t)t^2P_2 + t^3P_3$ , де  $t \in [0, 1]$ ,  $P_0$  та  $P_3$  - кінцеві точки,  $P_1$  та  $P_2$  - опорні. Задача растрезизації створюється як визначення ступеня покриття кожного пікселя сітки контуром, тобто обчислення інтеграла площі перетину піксельного квадрата  $[i, i + 1] \times [j, j + 1]$  з ідеальною фігурою. Розв'язання інтеграла для довільних кривих Без'є є обчислювально нераціональним, тому на практиці застосовуються наближені методи.

Алгоритм Брезенхема виконує растрезизацію відрізків за  $O(n)$  операцій без множення та ділення через цілочисельну арифметику. Проте він є принципово бінарним: піксель або замальований повністю, або ні. При роздільній здатності 1200 dpi це приводить до чітко видимого аліасингу на діагональних елементах товщиною менше 0,3 мм. Для поліграфії такий результат є неприйнятним.

Алгоритм Ву відсторонює цей недолік через введення поняття дробового покриття: для кожного пікселя вздовж контуру обчислюється частка  $\alpha \in [0, 1]$ , що відображає ступінь його перекриття, після чого піксель фарбується з інтенсивністю  $\alpha \cdot C$ , де  $C$  - колір лінії. Це забезпечує плавне згладжування країв практично без зростання обчислювальної складності порівняно з Брезенхемом.

Також було розглянуто scan-line rendering з правилом заповнення, що використовується в інтерпретаторах PostScript та PDF. Для кожного горизонтального рядка пікселів обчислюються точки перетину контуру з прямою  $y = const$ , які сортуються за  $x$ . Заповнення виконується між

парами точок перетину за правилом nonzero winding або even-odd. Це забезпечує правильну обробку самоперетинних контурів, особливо в замкнених гліфів із внутрішніми порожнинами.

Для порівняльного оцінювання сформовано тестовий набір поліграфічних символів різних гарнітур, що включало прямі штрихи, діагональні елементи, серифи та замкнені контури. Еталонне зображення отримано растеризацією при 9600 dpi з подальшим зменшенням до 1200 dpi методом Ланцоша.

Отримані результати: алгоритм Брезенхема - SSIM = 0,81, виразні артефакти на діагональних штрихах; алгоритм Ву - 0,94, помітне покращення на тонких елементах; субдивізія Без'є з суперсемплінгом  $4\times$  - 0,97; суперсемплінг  $16\times$  - 0,99 при чотириразовому збільшенні часу обчислень. Scan-line з алгоритмом Ву показав SSIM = 0,95 на замкнених контурах, що підтверджує його перевагу саме в цій категорії символів.

З'ясовано, що субдивізія кривих Без'є з суперсемплінгом  $4\times$  забезпечує оптимальний баланс точності та швидкості для поліграфічного друку при 1200 dpi. Алгоритм Брезенхема непридатний для якісної поліграфії через бінарну природу. Алгоритм Ву є прийнятним компромісом у задачах, де пріоритетом є швидкість. Перспективою подальших досліджень є GPU-реалізація субдивізії Без'є в RIP-системах реального часу, що може скоротити час растеризації на порядок.

#### **Список використаної джерел:**

1. Li T.-M., Huang W., Aittala M., Durand F. Differentiable vector graphics rasterization for editing and learning // ACM Transactions on Graphics. 2020. Vol. 39, No. 6. P. 1–15.
2. Loop C., Blinn J. Resolution independent curve rendering using programmable graphics hardware // ACM SIGGRAPH 2005 Papers. 2005. P. 1000–1009.
3. Yap C. K. Complete subdivision algorithms. I: Intersection of Bezier curves // Proceedings of the Twenty-Second Annual Symposium on Computational Geometry. 2006. P. 217–226.
4. Tian X., Günther T. A survey of smooth vector graphics: Recent advances in representation, creation, rasterization, and image vectorization // IEEE Transactions on Visualization and Computer Graphics. 2022. Vol. 30, No. 3. P. 1652–1671.