УДК 004.925.86

Улучшенный алгоритм генерации дуги для объемных 3D дисплеев

Харченко Н.О., Авксентьева О.А.
Донецкий национальный технический университет

В статье рассматриваются вопросы выбора оптимального метода генерации дуги, заданной произвольными параметрами в трехмерном пространстве для 3D дисплеев, с учетом цели минимизации количества сложных и комплексных математических преобразований. Рассматривается метод генерации заданного объекта на базе классического алгоритма Брезенхема, использующего поэтапное вычисление координат всех точек, принадлежащих множеству точек объекта. Выполнен анализ относительных и абсолютных погрешностей генерации объекта.

Основным средством общения средств вычислительной техники и человека на сегодняшний день являются устройства отображения информации, посредством которых результаты обработки некоторых данных компьютером предоставляются пользователю. Оставляя в стороне вывод алфавитно-цифровой информации, более ориентированной на поддержку языковой коммуникации, отображение данных в виде естественных или искусственных графических образов стало в настоящее время самым распространенным способом доведения человеку больших объемов информации [1].

Разработка новых способов построения трехмерных устройств отображения [5] информации (3D дисплеев) дает толчок новым исследованиям в области объемной визуализации, направленным на создание алгоритмической базы генерации, обработки и вывода «вокселизированных» объектов и сцен. Следует отметить, что в литературе [6] достаточно подробно рассмотрены вопросы создания воксельных моделей ("вокселизации") реальных объектов, например, с помощью трехмерных сканеров, в то время как методы и алгоритмы генерации типичных графических примитивов не рассматриваются.

Постановка задачи растрового разложения дуги

Предположим, что некоторая часть трехмерного евклидова пространства, которое отображается 3D дисплеем, имеет вид трехмерного параллелепипеда:

$$\Omega \in R3,\ 0 \le x \le X,\ 0 \le y \le Y,\ 0 \le z \le Z$$

С учетом возможности масштабирования, будем считать, что X=Y=Z=H, то есть Ω – трехмерный куб. Положим, что Ω заполнена вокселями – атомарными элементами, которые отображаются 3D дисплеем. Определим воксель как куб с единичной стороной. Центры «соседних» вокселей удалены друг от друга по координатным осям на единичное расстояние. Множество вокселей, заполняющих Ω можно представить как трехмерный массив вокселей $V_{i,j,l}$, где i,j,l – индексы, принимающие значения 1… int(H). Координаты центра вокселя определяются как

$$x_{i,j,l} = 0.5(i+1),\ y_{i,j,l} = 0.5(j+1),\ z_{i,j,l} = 0.5(l+1). \quad (1)$$

Соответственно, воксель, которому принадлежит точка $L \in \Omega$ с координатами XL, YL, ZL имеет индексы, которые определяются как

$$i = int(L_X),\ j = int(L_Y),\ l = int(L_Z). \quad (2)$$

В Ω задана дуга своими начальной точкой $N=[X_N, Y_N, Z_N]$, точкой центра дуги $C=[X_C, Y_C, Z_C]$, некоторой точкой $P=[X_P, Y_P, Z_P]$, однозначно определяющей плоскость, в которой лежит дуга, и углом дуги α.

Необходимо найти такую последовательность вокселей, каждый из которых лежит в некотором смысле «недалеко» от линии дуги и «плотно» заполняет промежуток между точкой N и конечной точкой генерируемой дуги и, будучи отображенной на экране 3D дисплея, воспринимается наблюдателем как дуга в трехмерном пространстве (с учетом масштабирования).

Определим два разных вокселя V_U и V_W как соседние, если у них:

– хотя бы один индекс не равен;
– индексы отличаются не более чем на 1.

Обозначим воксели V_N и V_K как начальный и конечный воксели растрового разложения дуги. Обозначим воксель V_C как центр дуги. Индексы вокселей V_N, V_C и V_P определяются по (2).

Задача растрового разложения дуги формулируется как задача определения множества вокселей $V^{<q>}$, q=1,2...M, к которому принадлежат воксели $V_N=V^{<1>}$ и $V_K=V^{<M>}$, каждый из которых (кроме начального и конечного) имеет два и только два соседних вокселя, центр каждого из которых лежит на минимальном расстоянии до дуги и количество вокселей во множестве минимально.

В [2] предложен метод построения такого множества, суть которого кратко заключается в следующем. Предполагается, что генерируемая дуга лежит в первом октанте. Пусть найден q-й воксель $V^{<q>}$ последовательности. Требуется определить следующий q+1-й воксель $V^{<q+1>}$. Рассматриваются семь соседних с $V^{<q>}$ вокселей - претендентов $V_{(k)}^{<q+1>}$, k=1,2,…,7 в октанте, соответствующем направлению вектора, исходящего из центра $V^{<q>}$ и параллельного касательной в точке дуги, ближайшей к центру $V^{<q>}$ (рис.1).

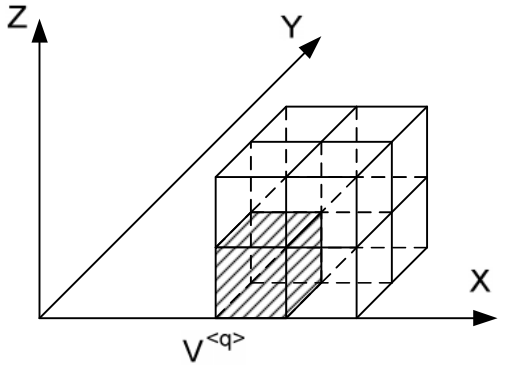

Рисунок 1 – Воксели-претенденты для первого октанта

Окончательный выбор следующего вокселя $V^{<q+1>}$ зависит от значений двух показателей погрешности вычислений разрабатываемого алгоритма:

$E1_{(k)}^{<q+1>}$ – расстояния от центра k-ого вокселя-претендента до плоскости дуги, где: k=1,2…7;

$E2_{(k)}^{<q+1>}$ – разницей между радиусом дуги и расстоянием от центра k-ого вокселя-претендента до центра окружности, где k=1,2…7;

Окончательно, $V^{<q+1>}$ в [2] предлагается выбирать как воксель-претендент с минимальной суммой $E1_{(k)}^{<q+1>}$ и $E2_{(k)}^{<q+1>}$.

Разница между длиной радиуса дуги и расстоянием от вокселя-претендента до центра дуги рассчитывается как

$$E1_{(k)}^{<q+1>} = R - |\overline{CV_{(k)}^{<q+1>}}|, \quad (3)$$

где: R – радиус дуги, $|\overline{CV_{(k)}^{<q+1>}}|$ - модуль вектора, соединяющего точку центра дуги и k-й воксель-претендент, k=1,2…7. Радиус дуги рассчитывается как

$$R = \sqrt{(X_N - X_C)^2 + (Y_N - Y_C)^2 + (Z_N - Z_C)^2} \quad (4)$$

Расстояние до плоскости от каждого вокселя-претендента рассчитывается как

$$E2_{(k)}^{<q+1>} = (xn \cdot V_{X(k)}^{<q+1>} + yn \cdot V_{Y(k)}^{<q+1>} + zn \cdot V_{Z(k)}^{<q+1>} + D) \cdot \frac{1}{|\overline{n}|}, \quad (5)$$

где: xn, yn, zn – координаты вектора нормали к заданной плоскости, D – коэффициент уравнения плоскости, $V_{X(k)}^{<q+1>}, V_{Y(k)}^{<q+1>}, V_{Z(k)}^{<q+1>}$ - координаты k-го вокселя-претендента, k=1,2…7, $|\overline{n}|$ - модуль вектора нормали к заданной плоскости.

Модифицированный алгоритм 3D растрового разложения дуги

С целью ускорения исходного подхода из [2] предлагается следующие модификации алгоритма.

Во-первых, можно существенно упростить вычисление координат конечной точки дуги с помощью умножения вектора-столбца, координаты которого заданы координатами вектора

$$\overline{CN} = \{XN - XC, YN - YC, ZN - ZC\} = \{XCN, YCN, ZCN\} \quad (6)$$

на матрицу поворота,

$$M(\bar{n},\alpha) = \begin{pmatrix} \cos(\alpha) + (1-\cos(\alpha))\cdot xn2 & (1-\cos(\alpha))\cdot xnyn - \sin(\alpha)\cdot zn & (1-\cos(\alpha))\cdot xnzn + \sin(\alpha)\cdot yn \\ (1-\cos(\alpha))\cdot xnyn + \sin(\alpha)\cdot zn & \cos(\alpha) + (1-\cos(\alpha))\cdot yn2 & (1-\cos(\alpha))\cdot ynzn - \sin(\alpha)\cdot xn \\ (1-\cos(\alpha))\cdot xnzn - \sin(\alpha)\cdot yn & (1-\cos(\alpha))\cdot znyn + \sin(\alpha)\cdot xn & \cos(\alpha) + (1-\cos(\alpha))\cdot zn2 \end{pmatrix}, \quad (7)$$

где: $\bar{n}$ = {xn,yn,zn} - вектор нормали к плоскости, т.е. единичный вектор оси вращения, а α – угол дуги.

Во-вторых, предлагается унифицировать определение множества векселей-претендентов на каждом шаге алгоритма за счет вычисления матрицы приращений

M	Sx	0	0	Sx	0	Sx	Sx
	0	Sy	0	Sy	Sy	0	Sy
	0	0	Sz	0	Sz	Sz	Sz

(8)

с помощью координат вектора касательной к дуге, проведеного из текущего вокселя. Координаты вектора касательной $\bar{k}$ = {kx,ky,kz} находятся с помощью векторного

произведения радиус-вектора $\overline{CN}$ и вектора нормали к плоскости генерации дуги $\bar{n}$. Приращения для каждого вокселя-претендента определяются как

$$Sx = sign(kx),\ Sy = sign(ky),\ Sz = sign(kz), \tag{9}$$

где: kx, ky, kz – координаты вектора касательной $\bar{k}$ к заданной дуге.

Координаты центров вокселей - претендентов $V_{(k)}^{<q+1>}$, k=1,2...7 при этом могут быть найдены как

$$\begin{bmatrix} V_{X\,(k)}^{<q+1>} \\ V_{Y\,(k)}^{<q+1>} \\ V_{Z\,(k)}^{<q+1>} \end{bmatrix} = \begin{bmatrix} V_X^{<q>} \\ V_Y^{<q>} \\ V_Z^{<q>} \end{bmatrix} + M^{(k)} + 0.5. \tag{10}$$

В-третьих, можно отказаться от вычисления квадратних корней при расчете расстояний $E1_{(k)}^{<q+1>}$ и $E2_{(k)}^{<q+1>}$, т.к. абсолютное значение расстояний не важны для выбора следующего вокселя последовательности из вокселей-претендентов.

Алгоритм растрового разложения дуги в трехмерном пространстве, с учетом вышеизложенного, может бать записан на псевдоязыке из [3] следующим образом.

Алгоритм воксельного представления дуги

Входные данные: N начальная точка дуги, C центральная точка дуги, P точка на плоскости, α угол дуги.
Выходные данные: массив вокселей V

```
Begin
    Input C, P, N, α;
    Определение V_C, V_P, V_N по (2)
    Вычисление координат векторов CP, CN по (6)
    Вычисление координат вектора нормали |n|  к плоскости
    Вычисление радиуса дуги по (4)
    Вычисление координат точки K по (6) и (7)
    Repeat
        Вычисление координат вектора касательной k
        Заполнение матрицы приращений M по (8) и (9)
        Do k = 1,2 … 7
            V_(k)^<q+1> := V_N + M^(k) +0.5 по (1) и (10)
            Вычисление погрешности E1_(k)^<q+1> по (3)
            Вычисление погрешности E2_(k)^<q+1> по (5)
        End Do
        MIN = MIN(E1_(k)^<q+1> + E2_(k)^<q+1>)
        Do k = 1,2 … 7
            If (E1_(k)^<q+1> + E2_(k)^<q+1> = MIN)
                V_points := V_N + M^(k)
                points:=points+1
            End if
        End Do
    Until V_N≠V_K
Output V
End
```

Экспериментальные исследования модифицированного алгоритма 3D растрового разложения дуги

Анализ результатов работы 1000 экспериментальных запусков генерации дуги, заданной случайными начальными данными, показал, что:

- предложенный алгоритм генерирует дугу с ошибкой, не превышающей половину диагонали вокселя – 0.86602, что свидетельствует о корректности работы алгоритма.
- модификация алгоритма уменьшает время генерации в 5.7 раза. Такой показатель достигается за счет уменьшения количества процессорного времени, затраченного на вычисление сложных математических операций.

Таблица 1 – Сравнение характеристик базового и модифицированного алгоритмов

	Базовый алгоритм	Модифицированный алгоритм
Скорость генерации, вокселей/с	16695	96153
Количество экспериментов	1000	1000
Максимальная ошибка	0.84812	0.84506
Время генерации 1000 дуг, с	94.499	24.781
Количество рассчитанных вокселей	1577000	1577000

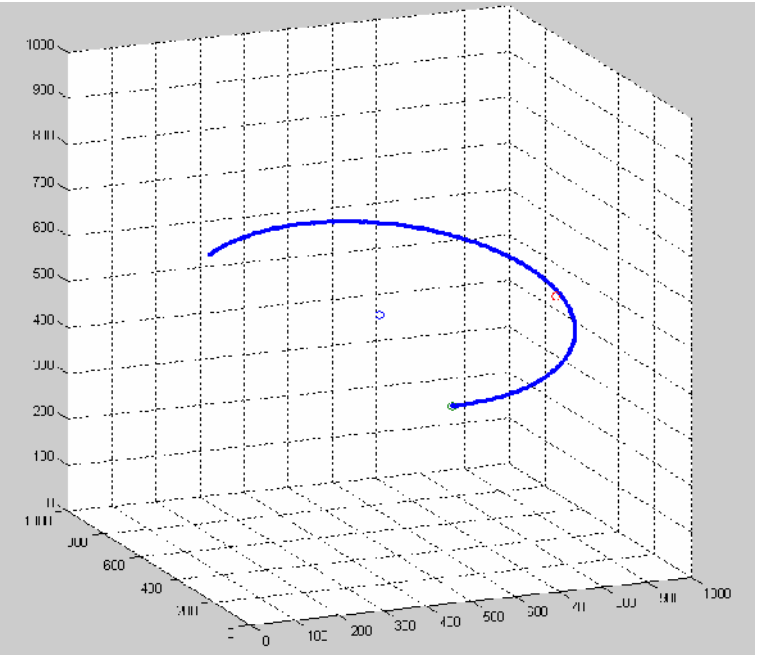

Рисунок 2 – Пример генерации дуги (α = 217.90)

Литература

[1] Башков Е.А., Авксентьева О.А., Аль-Орайкат Анас М. К построению генератора графических примитивов для трехмерных дисплеев. В сб. Наукові праці Донецького національного технічного університету, серія "Проблеми моделювання та автоматизації проектування динамічних систем". Вип. 7 (150). - Донецьк, ДонНТУ. - 2008 .- ст. 203-214

[2] Мілютін М.О., Аль-Орайкат Анас Махмуд, Авксентьева О.А. Генерація окружності для 3-D дисплеїв. В сб. «Информатика и компьютерные технологии-2009» - Донецьк: ДонНТУ. - 2009. – 147с.

[3] Препарата Ф., Шеймос М. Вычислительная геометрия: Введение: Пер. с англ. – М.: Мир, 1989. 478 с.

[4] Д.В. Беклемишев. Курс аналитической геометрии и линейной алгебры. М.: Наука, 1980.

[5] Favalora G.E., Volumetric 3D Displays and Application Infrastructure // “Computer”, 2005, August, pp 37- 44.

[6] Роджерс Д. Алгоритмические основы машинной графики. — Пер. с англ. — Москва.: «Мир», 1989. — 512 с.

[7] Авксентьева О.А., Аль-Орайкат Анас, Хлопов Д.И. Улучшенный алгоритм генерации прямой для 3D – дисплеев. Материалы 14 – 10 международного молодежного форума «Радиоэлектроника и молодежь в XXI веке», 18 -20 марта 2010, Харьков 2010, сб. материалов 4.1. – Харьков: ХНУРЭ. 2010. – 527с.