Министерство образования и науки РФ

ФГБОУ ВПО «Тамбовский государственный технический университет»

Кафедра «САПР»

Лабораторная работа №5

по дисциплине «Компьютерная графика»

Закраска методом Гуро

Выполнил: студент гр. Д. Ю.

Проверил: С. А.

Тамбов,

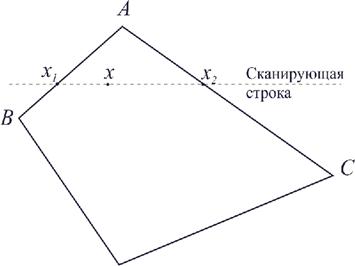
***Цели и задачи****.*

Реализовать закраску объекта методом Гуро.

***Решение задачи****.*

Метод заключается в том, что используются не нормали к плоским граням, а нормали к аппроксимируемой поверхности, построенные в вершинах многогранника. После этого вычисляются интенсивности в вершинах, а затем во всех внутренних точках многоугольника выполняется билинейная интерполяция интенсивности.

Метод сочетается с алгоритмом построчного сканирования. После того как грань отображена на плоскость изображения, для каждой сканирующей строки определяются ее точки пересечения с ребрами. В этих точках интенсивность вычисляется с помощью линейной интерполяции интенсивностей в вершинах ребра. Затем для всех внутренних точек многоугольника, лежащих на сканирующей строке, также вычисляется интенсивность методом линейной интерполяции двух полученных значений. На Рисунке 1 показан плоский многоугольник с вычисленными значениями интенсивностей в вершинах [1].

Пусть - интенсивности в вершинах A, B, C.

- горизонтальные координаты этих точек.

Тогда в точках пересечения сканирующей строки с ребрами многоугольника интенсивности можно вычислить по формулам интерполяции:

Рисунок Интерполяция интенсивности

где

где

После этого интенсивность в точке http://ok-t.ru/studopedia/baza5/460370616637.files/image068.gifполучаем путем интерполяции значений на концах отрезка:

где

К недостаткам метода Гуро следует отнести то, что он хорошо работает только с диффузной моделью отражения. Форма бликов на поверхности и их расположение не могут быть адекватно воспроизведены при интерполяции на многоугольниках. Кроме того, есть проблема построения нормалей к поверхности. В алгоритме Гуро нормаль в вершине многогранника вычисляется путем усреднения нормалей к граням, примыкающим к этой вершине. Такое построение сильно зависит от характера разбиения.

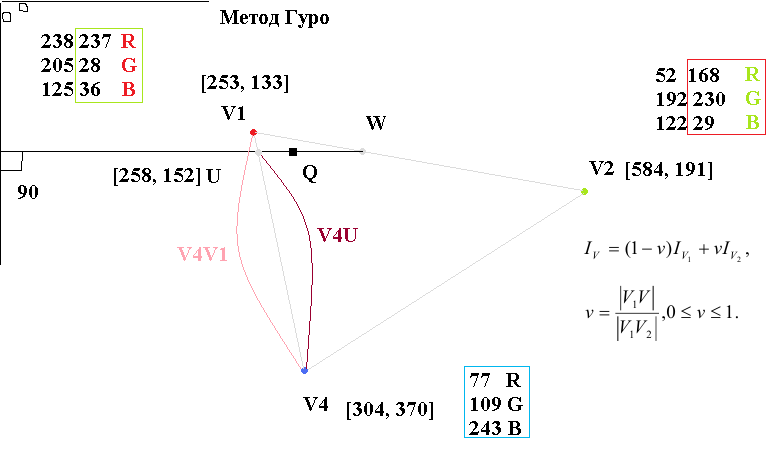
Пример вычисления интенсивности по схожему алгоритму:

Рисунок Треугольник для заливки

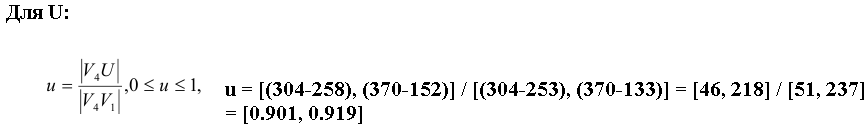
Даны три точки (Рисунок 2), у них известны координаты и rgb значения цвета. Для нахождения интенсивности точки U необходимо рассчитать во сколько длина V4U больше\меньше длины V4V1, а затем рассчитать rgb коэффициент точки U (Рисунок 3).

Рисунок Получить коэффициент u

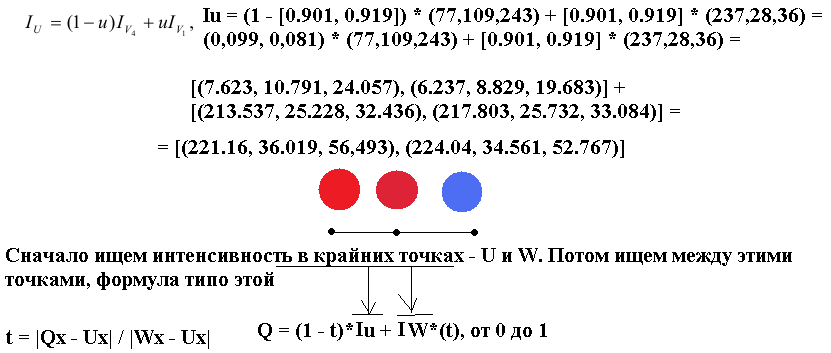
На рисунке 4 в конечном счете получаем интенсивности RGB с учетом того, что подсчет велся по XY координатам, следовательно, в конечном итоге два значения. Так как нам необходимо одно число, усредняем полученное значение и строим точку с полученным цветом.

Рисунок Вычислить интенсивность точки U

***Исходный код:***

//0.0 - темный//1.0 - светлый

AreaGuro**(**1.0**,** 1.0**,** 0.0**,** 500**,** 335**,** 400**,** 25**,** 30**,** 335**);**

/// <summary>

/// закрашивание Гуро.

/// Интенсивность считается только в вершинах, в остальных апроксимируется

///double Ia,Ib,Ic - интенсивности отраженного света в вершинах A,B,C

///int (xa...yc) - координаты вершин треугольника A,B,C

/// </summary>

**private** void AreaGuro**(**double Ia**,**double Ib**,**double Ic**,**double xa**,**double ya**,**double xb**,**double yb**,**double xc**,**double yc**)**

**{**

double I1**,** I2**;** //интенсивности краёв

double y**,** x**;** //искомая точка

double x1**,** x2**,** t1**,** t2**;** //вспомогательные точки (строка)

**if** **(**ya **==** yb**)**

**{**

t1 **=** ya**;**

ya **=** yc**;**

yc **=** t1**;**

t1 **=** xa**;**

xa **=** xc**;**

xc **=** t1**;**

t1 **=** Ia**;**

Ia **=** Ic**;**

Ic **=** t1**;**

**}**

**if** **(**ya **==** yc**)**

**{**

t1 **=** ya**;**

ya **=** yb**;**

yb **=** t1**;**

t1 **=** xa**;**

xa **=** xb**;**

xb **=** t1**;**

t1 **=** Ia**;**

Ia **=** Ib**;**

Ib **=** t1**;**

**}**

**if** **(**yc **>** ya**)**

**{**//вершина в верху

//закрашиваем линии от верхней к нижней

**for** **(**y **=** ya**;** y **<=** yc**;** y**++)**

**{**

//точки пересечения линии(y) с границами грани

x1 **=** **((**y **-** ya**)/(**yb **-** ya**))\*(**xb **-** xa**)** **+** xa**;**

x2 **=** **((**y **-** ya**)/(**yc **-** ya**))\*(**xc **-** xa**)** **+** xa**;**

//интенсивности в точках пересечения линии(y) с границами грани

t1 **=** **(**x1 **-** xb**)/(**xa **-** xb**);**

t2 **=** **(**x2 **-** xc**)/(**xa **-** xc**);**

I1 **=** t1**\***Ia **+** **(**1 **-** t1**)\***Ib**;**

I2 **=** t2**\***Ia **+** **(**1 **-** t2**)\***Ic**;**

**if** **(**x2 **>** x1**)**

**{**

//рисовать линию

**for** **(**x **=** x1**;** x **<** x2**;** x**++)**

**{**

Intensive**(**x**,** y**,** x1**,** x2**,** I1**,** I2**);**

**}**

**}**

**else**

**{**

//рисовать линию

**for** **(**x **=** x1**;** x **>** x2**;** x**--)**

**{**

Intensive**(**x**,** y**,** x1**,** x2**,** I1**,** I2**);**

**}**

**}**

**}**

**}**

**else**

**{**//вершина в низу

//закрашиваем линии от нижней k верхней

**for** **(**y **=** ya**;** y **>=** yc**;** y**--)**

**{**

//точки пересечения линии(y) с границами грани

x1 **=** **((**y **-** ya**)/(**yb **-** ya**))\*(**xb **-** xa**)** **+** xa**;**

x2 **=** **((**y **-** ya**)/(**yc **-** ya**))\*(**xc **-** xa**)** **+** xa**;**

//интенсивности в точках пересечения линии(y) с границами грани

t1 **=** **(**x1 **-** xb**)/(**xa **-** xb**);**

t2 **=** **(**x2 **-** xc**)/(**xa **-** xc**);**

I1 **=** t1**\***Ia **+** **(**1 **-** t1**)\***Ib**;**

I2 **=** t2**\***Ia **+** **(**1 **-** t2**)\***Ic**;**

//рисовать линию

**if** **(**x2 **>** x1**)**

**{**

//рисовать линию (строка с лево на право)

**for** **(**x **=** x1**;** x **<** x2**;** x**++)**

**{**

Intensive**(**x**,** y**,** x1**,** x2**,** I1**,** I2**);**

**}**

**}**

**else**

**{**

//рисовать линию (строка с право на лево)

**for** **(**x **=** x1**;** x **>** x2**;** x**--)**

**{**

Intensive**(**x**,** y**,** x1**,** x2**,** I1**,** I2**);**

**}**

**}**

**}**

**}**

**}**

/// <summary>

/// Вычислить интенсивность точки [x,y]

/// x, y - рассматриваемая точка;

/// x1, x2 - граничные точки;

/// I1, I2 - интенсивность в граничных точках

/// </summary>

**private** void Intensive**(**double x**,** double y**,** double x1**,** double x2**,** double I1**,** double I2**)**

**{**

//интенсивность в точке x

double t **=** **(**x2 **-** x**)** **/** **(**x2 **-** x1**);**

double I **=** t **\*** I1 **+** **(**1 **-** t**)** **\*** I2**;**

//поставить точку

SetPixel**((**int**)**x**,** **(**int**)**y**,** I**,** I**,** I**);**

**}**

/// <summary>

/// Отрисовка

/// </summary>

**private** void SetPixel**(**int x**,** int y**,** double r**,** double g**,** double b**)**

**{**

Graphics graphics **=** Graphics**.**FromImage**(**pictureBox1**.**Image**);**

//Заливка выбор цвета

double sr **=** 105**\***r**;**

double sg **=** 155 **\*** g**;**

double sb **=** 25 **\*** b**;**

graphics**.**DrawRectangle**(new** Pen**(**Color**.**FromArgb**((**int**)**sr**,** **(**int**)**sg**,** **(**int**)**sb**)),** x**,** y**,** 1**,** 1**);**

**}**

***Конечный результат:***

На рисунке 5 представлен треугольник с закрашиванием с лево на право из темно зеленого в светло зеленый цвет.



Рисунок Заливка фигуры алгоритмом

Список источников:

1. Закраска методом Гуро, URL: http://studopedia.net/5\_20178\_zakraska-metodom-guro.html