

происходит, и они задают цвет фона. Второй цвет в области пересечения дает смесевой.

Так как было установлено, что масштабирование изображения на экране монитора приводит к формированию различной цветности, проведен эксперимент по выводу на разработанных векторных штриховых изображениях на различных принтерах. При выводе на лазерном принтере достигается высокая четкость штрихов, однако визуального смещения тона не происходит. Струйный принтер позволяет воспроизвести более однородные тона. При сравнении результата вывода на струйном принтере на различных видах бумаги (офсетной и мелованной) формируется некоторое смещение тональности вследствие различной белизны бумаги. Установлено, что если бумага характеризуется большей белизной, то происходит формирование холодного тона.

#### **Выводы:**

1. При внедрении штриховых данных необходимо учитывать зависимость от разрешения вывода и заданной толщины штрихов.

2. При выводе на лазерном принтере необходимо задавать толщину линий не менее 100 мкм, в то время как струйный принтер хорошо передает цветность при толщине линий в 10 мкм.

#### **ЛИТЕРАТУРА**

1. Русский орфографический словарь: около 180 000 слов [Электронная версия] / О. Е. Иванова, В. В. Лопатин (отв. ред.), И. В. Нечаева, Л. К. Чельцова. – 2-е изд., испр. и доп. – М.: Российская академия наук. Институт русского языка имени В. В. Виноградова, 2004.

2. Марр Д. Зрение. Информационный подход к изучению представления и обработки зрительных образов. – М., Радио и связь, 1987.

3. Бондаренко В.М. [и др.]. Пространственное зрение. – СПб.: Наука, 1999.

УДК 004

А.В. Харланович, маг.; О.А. Новосельская, доц., канд. техн. наук  
(БГТУ, г. Минск)

### **ИСПОЛЬЗОВАНИЕ ФРАКТАЛОВ В ТОПОЛОГИИ 3D-ОБЪЕКТОВ**

Существуют несколько видов трехмерного моделирования. Объемная визуализация с использованием полигонов является самой первой разновидностью 3D-моделирования. Это низкоуровневое моделирование, которое позволяет визуализировать объект с помощью полигональной сетки.

Простыми словами, Полигональная сетка – представляет собой

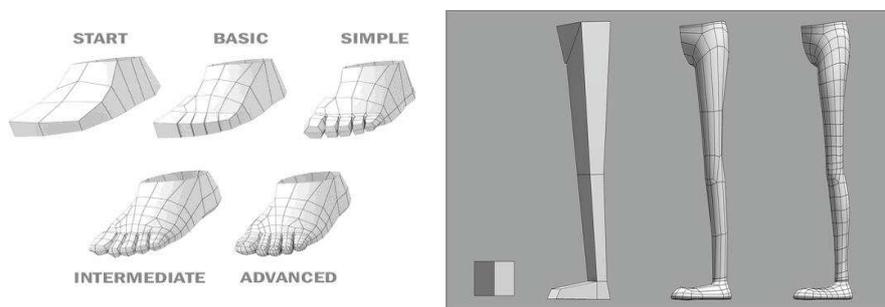
набор вершин, ребер и граней, которые определяют форму многогранного объекта в трехмерной компьютерной графике. По своей сути, когда мы видим 3D-модель, мы и видим полигональную сетку, так как составляющие полигональной сетки и образуют формы в созерцаемой нами модели.

Обычно, когда мы рассматриваем законченный и отрендеренный 3D-объект, то зачастую нам не показывают его в сетчатом виде. Но при создании 3D-объекта, художники очень часто используют режим отображения сетки (wire-frame), для того, чтоб правильно строить топологию и формы модели. Работая в этом режиме, художники могут манипулировать составляющими полигональной сетки и тем самым создавать 3D-модель нужным им образом с учетом правильной формы и топологии.

Являясь неотъемлемой частью 3D-модели, работа над сеткой происходит в компьютерных программах, предназначенных для работы с 3D-графиком. Из наиболее популярных программ можно выделить: Autodesk Maya, Autodesk 3Ds-max, Blender, Modo и другие.

Они позволяют работать с сеткой в более традиционном виде, точно так, как над формированием статуй работают классические скульпторы. Вместо работы с каждым полигоном по отдельности или с группой полигонов, 3d скульпторы работают одновременно с тысячами или десятками тысяч полигонов при помощи специальных кистей и инструментов, которые встроены в программы для 3D-скульптинга.

На рисунке 1 изображены уровни детализации двух 3D-моделей. Чем больше производится разбиение на полигоны, тем выше уровень ее детализации. Топология полигональной сетки – это плавная и потоковая организованность полигонов в 3D-модели или как именно полигоны формируют 3D-модель.



**Рисунок 1 – Вариация уровней разбиения полигональной сетки**

При использовании модификаторов, что позволяют создавать сложные объекты, получаемый объект будет зависеть от топологии модели. При использовании модификатора, эффект при применении

модификатора будет зависеть от уровня разбиения. Чем на больше частот сетки мы разобьем, тем плавнее будет эффект.

В 3D-моделировании часто используют сплайны в качестве основы для построения более сложных трехмерных тел. Все сплайны состоят из точек и сегментов. Меняя тип точек, можно создавать сглаженные линии. Тут же отображая сплайны при рендере можно указать количество разбиений на сегменты. Чем больше выставляем число разбиений, тем плавнее мы добиваемся результата при рендере.

Напомним, про определения фрактала. У этого понятия нет строгого определения, но в основе фракталов лежит очень простая идея: бесконечное по красоте и разнообразию множество фигур можно получить из относительно простых конструкций при помощи всего двух операций – копирования и масштабирования.

Можно сказать, что фракталы – это способ легкого и удобного представления сложных неевклидовых объектов, образы которых похожи на природные. Одним из таких образов можно реализовать через скрипт для построения дерева Пифагора.

В листинге 1 представлен скрипт построения фрактального дерева в 3Ds-max. Алгоритм состоит из построение вертикального отрезка, рекурсивного построения из верхнего конца этого отрезка еще двух под определенными углами и вызывания функцию построения двух последующих отрезков для каждой ветви дерева.

```
/* Основной код скрипта */
s="ff"
s1=gens s "ff-[-f+f+f]+[+f-f-f]&[f]^" 2
CreateFrac s1 0 90 10
/* Функция замены символов */
fn gens str s n=(
  str1=copy str
  count=0;
  for j=1 to n do (
    for i=1 to str.count do (
      count+=1
      if str[i]=="f" then (
        str1=replace str1 count 1 s
        count+=s.count-1 ) )
    str = copy str1
    count=0; )
  return str1
)
/* Генерация фрактала */
fn CreateFrac str fi theta a= ( ...
for i=1 to str.count do (
  if str[i]=="f" then (
    ss=SplineShape pos:[X,Y,Z]
```

```

s11 = AddNewSpline ss
addKnot ss 1 #corner #curve [X,Y,Z]
X+=a*(cos theta)*(cos fi)
Y+=a*(cos theta)*(sin fi)
Z+=a*(sin theta)
addKnot ss 1 #corner #curve [X,Y,Z]
updateshape ss
count+=1
) ...

```

### Листинг 1 – Скрипт построения фрактального дерева

Поскольку в основе фрактального дерева лежит стандартная кривая Безье, то и топология результирующего объекта подчиняется ее параметрам. Эти параметры можно регулировать непосредственно из скрипта. Преобразовывая прямые линии в Безье, для плавной детализации модели, требуется указать количество разбиений на сегменты. Пример представлен на рис. 2.

Как и в предыдущих случаях, чем больше мы укажем число разбиений, тем плавнее будет модель. Также можно реализовать задание типов узлов и совместно с количеством сегментов управлять нужным уровнем детализации

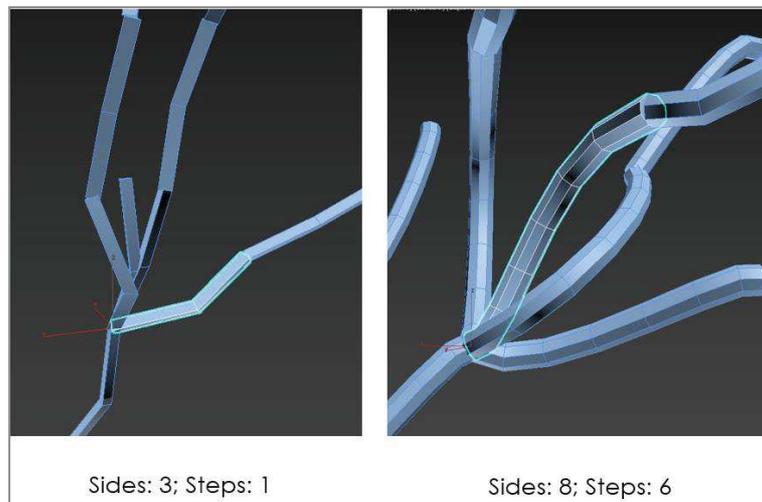


Рисунок 2 – Применение топологии к кривой Безье

Можно реализовать построение дерева и другим способом. Для этого скрипт будет содержать функцию с передаваемыми параметрами высоты и процент разветвления дерева, выбора родителя для каждого последующего элемента дерева. При этом есть возможность непосредственно в скрипте указать формы, которые будут включать деревья, задавать их свойства и применять модификаторы. Т.е. с помощью скрипта можно управлять всеми необходимыми параметрами результирующих объектов, независимо от их форм и типа (геометрические формы или плоские кривые), что особенно важно для построе-

ния 3d-моделей с заданным уровнем детализации.

#### ЛИТЕРАТУРА

1. Основы компьютерной графики: 3D-моделирование и 3D-печать. Лисяк В.В., 2021.
2. Разработка трехмерных (3D) моделей в Autodesk 3ds max 7 + CD. Стивен Тилл, Джеймс О'Коннелл, 2006.
3. Геометрическая топология и теория множеств. Чернавский А.В., 2004.
4. Просто Фрактал. 3-е изд., Сергей Деменок, 2016.

УДК 004.4

А.Н. Щербакова, ассист.;  
Д.М. Романенко, канд. техн. наук, зав. кафедрой ИиВД  
(БГТУ, г. Минск)

#### **РАЗРАБОТКА АЛГОРИТМА КОДИРОВАНИЯ ШТРИХОВЫХ ВЕКТОРНЫХ ИЗОБРАЖЕНИЙ**

Принцип штрихового кодирования – кодирование алфавитно-цифровых символов в виде чередования черных и светлых полос различной ширины (штрихов и пробелов).

Для воспроизведения защитных изображений было рекомендовано применение толщины штрихов не менее 40 мкм. Выделяющая способность на стандартной офсетной бумаге для лазерных и струйных принтеров составляет порядка 100 мкм. Итоговый размер штриха будет определяться разрешением изображения и частотой линий. Чем выше частота линий и ниже разрешение экрана, тем меньше итоговый размер штрихов [1].

Далее будет произведен расчет параметров защитного штрихового изображения для лазерных и струйных принтеров и специального оборудования.

Пусть размер холста – 1000x1000 px, размер защитного штрихового изображения – 600x600 px, разрешение изображения – 300 dpi, для лазерных и струйных принтеров толщина базового штриха условно равна 100 мкм, расстояние между штрихами – 100 мкм [1].

Необходимо перевести размер защитного элемента в мм, учитывая разрешение изображения:

$$A(\text{мм}) = \frac{a}{D} \cdot 25,4, \quad (1)$$

где  $a$  – ширина защитного элемента, px;  $D$  – разрешение изображения, dpi. Таким образом, размер защитного элемента равен 50x50 мм.

Тогда количество линий в 1 см для лазерных и струйных принтеров: