

УДК 621.397:004.932

СИНТЕЗ ТРЕХМЕРНЫХ ОБЪЕКТОВ С ПОМОЩЬЮ ПОЛИГОНАЛЬНЫХ СЕТОК

ОШАРОВСКАЯ Е.В., СОЛОДКА В. И.

Одесская национальная академия связи им. А.С. Попова

SYNTHESIS OF THREE-DIMENSIONAL OBJECTS BY MEANS OF GROUND GRIDS

OSHAROVSKAYA E. V., SOLODKA V. I.

Odessa national academy of telecommunications n.a. A.S. Popov

Аннотация. Показаны способы представления полигональных сеток с помощью хранения вершин, ребер и граней. Даны вершинное и графовое представления для трёхмерного объекта.

Abstract The methods of presentation of polygonal nets by storage of tops, ribs and verges are presented. Given vertex and graph representation for three-dimensional object.

Во многих приложениях телевизионных систем возникает потребность в представлении трехмерных объектов: при проектировании объемных изображений, при восстановлении трехмерных объектов по изображениям их поперечных сечений, построенных с помощью компьютерной графики.

Основной проблемой полигональных сеток является слишком сложная для объекта визуализация.

Нам уже известно, как изображаются трехмерные объекты, когда их можно представить в виде последовательности отрезков прямых, заданных в мировых координатах. Совокупность отрезков не является описанием объекта, поскольку отрезки сами по себе не определяют поверхностей. В то же время информация о поверхностях необходима для проведения вычислений, связанных с удалением невидимых частей изображения, для определения объемов и т. д. Таким образом, мы приходим к выводу, что для описания трехмерных объектов необходимы поверхности – примитивы более высокого уровня, чем отрезки.

Рассмотрим широко используемое представление поверхностей в пространстве – полигональные сетки. Полигональная сетка представляет собой совокупность ребер, вершин и многоугольников. Вершины соединяются ребрами, а многоугольники рассматриваются как последовательности ребер или вершин [1].

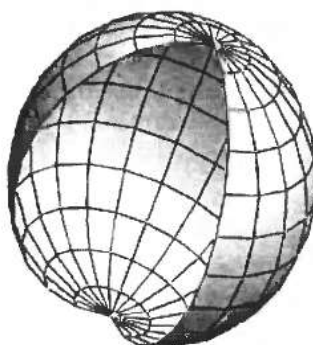


Рисунок 1 – Вершины трехмерного объекта

Вершины — это точки, в которых сходится и соединяется друг с другом любое число ребер.



Рисунок 2 – Вершины трехмерного объекта

Ребра — это линии границы грани. Ребра могут быть видимыми, если соседние грани не лежат в одной плоскости, в противном случае они невидимы.



Рисунок 3 – Ребра трехмерного объекта

Грани (полигоны) — это участки плоскости треугольной или четырехугольной формы, представляющие собой элементарные ячейки сетки. В одной плоскости объекта может находиться множество граней, которые внешне будут совершенно неразличимы.

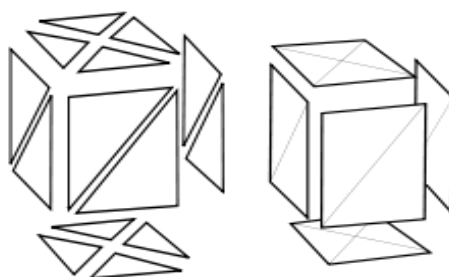


Рисунок 4 – Грани и ребра трехмерного объекта

И так, наружную форму всех объемных объектов легко и естественно описать с помощью полигональных сеток. Полигональные сетки применяются также для представления объектов, ограниченных криволинейными поверхностями. Однако недостатком этого метода является его приближительность. Видимые ошибки в таком представлении можно сделать сколь угодно малыми, используя все большее число многоугольников для улучшения кусочно-линейной аппроксимации объекта, но это приведет к дополнительным затратам памяти и вычислительного времени для алгоритмов, работающих с таким представлением. При этом необходимо минимизировать ошибки при построении объекта с помощью полигональной сетки.

Однако, математически, полигональная сетка может быть представлена в виде неструктурированной сетки, или неориентированного графа, с добавлением свойств геометрии, формы и топологии.

Полигональные сетки строятся путём решения алгебраических уравнений. Примером простейшей сетки, заданной на отрезке, может служить множество

$$\{X_k\} = \{X_1, X_2 \dots X_k\},$$

где $X_k = X_1 + dx \cdot k - 1$.

Величина dx в этом случае называется шагом расчетной сетки. Основными достоинствами алгебраических методов являются хороший контроль распределения внутренних узлов сетки и высокая эффективность их численной реализации, что особенно важно при построении адаптивных сеток.

Поверхности, чаще называемые группами сглаживания, полезны, но не обязательны для группирования гладких областей. Представьте себе цилиндр с крышками, такой как жестяная банка. Для гладкого затенения сторон, все нормали должны указывать горизонтально от центра, тогда как нормали крышек должны указывать в $+ / - (0,0,1)$ направлениях. Этот способ определения где прекращать сглаживание для того, чтобы группировать гладкие части сетки, также, как полигоны группируют трехсторонние грани. Как альтернатива предоставлению поверхностей сглаживания, сетка может содержать другую информацию для расчета тех же данных, такая как разделяющий угол. Также, полигональные сетки с очень высоким разрешением менее подвержены проблемам, для решения которых требуются группы сглаживания, так как их полигоны настолько малы, что нужда в них пропадает. Кроме того, другая альтернатива существует в возможности просто отсоединения самих поверхностей от оставшейся части сетки. Финальная визуализация, т.е. визуализировать не пытаются сглаживать ребра между несмежными полигонами.

Целью данной работы является анализ методов синтеза трехмерных объектов с помощью полигональных сеток.

Формат полигональной сетки может определять и другие полезные данные[2]. Полигональные сетки могут быть представлены множеством способов, используя разные способы хранения вершин, ребер и граней. В них входят:

- список граней: описание граней происходит с помощью указателей в список вершин.
- граф представление: в нём каждая точка ребра указывает на две вершины, две грани и четыре ребра, которые её касаются. Крылатое представление позволяет обойти поверхность за постоянное время, но у него большие требования по памяти хранения.
- полуреберные сетки: способ похож на граф представление, за исключением того, что используется информация обхода лишь половины грани.
- четырехреберные сетки, которые хранят ребра, полуребра и вершины без какого-либо указания полигонов. Полигоны прямо не выражены в представлении, и могут быть найдены обходом структуры. Требования по памяти аналогичны полуреберным сеткам.
- таблица углов, которые хранят вершины в предопределенной таблице, такой что обход таблицы неявно задает полигоны. В сущности, это “веер треугольников”, используемый в финальной визуализации. Представление более компактное и более производительное для нахождения полигонов, но операции по их изменению медленны. Более того, таблицы углов не представляют сетки полностью. Для представления большинства сеток нужно несколько таблиц углов, т.е. “вееров треугольников”.
- вершинное представление: представлены лишь вершины, указывающие на другие вершины. Информация о гранях и ребрах выражена неявно в этом представлении. Однако, простота представления позволяет проводить над сеткой множество эффективных операций.

Выбор структуры данных определяется применением, необходимой производительностью, размером данных, операциями, которые будут выполняться. К примеру, легче иметь дело с треугольниками, чем с многоугольниками общего вида, особенно в вычислительной геометрии. При реализации любого объекта используется метод эквидистантной сетки, которая позволяет создать шаблон с высокой точностью. Важной особенностью сеток метода является разбиения объекта на менее мелкие детали с большой точностью. Чем больше плотность исследуемого объекта, тем меньше шаг дискретизации сетки (треугольников) [3]. Для определенных операций необходимо иметь быстрый доступ к топологической информации, такой как ребра или соседние грани; для этого требуются более сложные структуры, такие как граф представление.

Вершинное представление (ВП) описывает объект как множество вершин, соединенных с другими вершинами. Это простейшее представление, но оно не широко используемое, так как информация о гранях и ребрах не выражена явно. Поэтому нужно обойти все данные чтобы сгенерировать список граней для финальной визуализации. Кроме того, нелегко выполняются операции на ребрах и гранях.

Таблица 1 – Список вершин объекта

Список вершин	Координаты вершин	Обход вершин
V0	0,0,0	V1 V5 V4 V3 V9
V1	1,0,0	V2 V6 V5 V0 V9
V2	1,1,0	V3 V7 V6 V1 V9
V3	0,1,0	V2 V6 V7 V4 V9
V4	0,0,1	V5 V0 V3 V7 V8
V5	1,0,1	V6 V1 V0 V4 V8
V6	1,1,1	V7 V2 V1 V5 V8
V7	0,1,1	V4 V3 V2 V6 V8
V8	0,7 0,7 0	V5 V6 V7 V8
V9	0,7 0,7 1	V0 V1 V2 V3

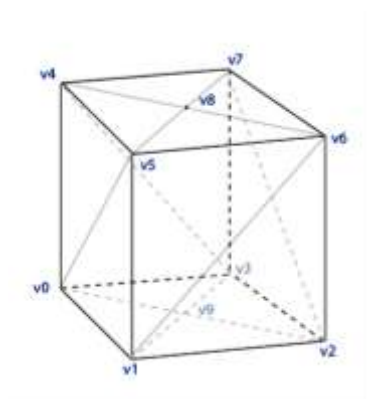


Рисунок 5 – Параллелепипед с использованием вершинного представления сетки

Однако, сетки ВП извлекают выгоду из малого использования памяти и эффективной трансформации. Рисунок 5 показывает пример параллелепипеда изображенный с использованием ВП сетки. Каждая вершина индексирует её соседние вершины. Заметьте, что последние две вершины, 8 и 9 сверху и снизу параллелепипеда, имеют четыре связанных вершины, а не пять. Главная система должна справляться с произвольным числом вершин связанных с любой данной вершиной.

Таблица 2 – Список используемых граней объекта

Список граней	Обход вершин
f0	V0 V4 V5
f1	V0 V5 V1
f2	V1 V5 V6
f3	V1 V6 V2
f4	V2 V6 V7
f5	V2 V7 V3
f6	V3 V7 V4
f7	V3 V4 V0
f8	V8 V5 V4
f9	V8 V6 V5
f10	V8 V7 V6
f11	V8 V4 V7
f12	V9 V5 V4
f13	V9 V6 V5
f14	V9 V7 V6
f15	V9 V4 V7

Таблица 3 – Параметры трехмерного объекта

Список вершин	Координаты вершин	Список граней
V0	0,0,0	f0 f1 f12 f15 f7
V1	1,0,0	f2 f3 f13 f12 f1
V2	1,1,0	f4 f5 f14 f13 f3
V3	0,1,0	f6 f7 f15 f14 f5
V4	0,0,1	f6 f7 f0 f8 f11
V5	1,0,1	f0 f1 f2 f9 f8
V6	1,1,1	f2 f3 f4 f10 f9
V7	0,1,1	f4 f5 f6 f11 f10
V8	0,7; 0,7; 0	f8 f9 f10 f11
V9	0,7; 0,7; 1	f12 f13 f14 f15

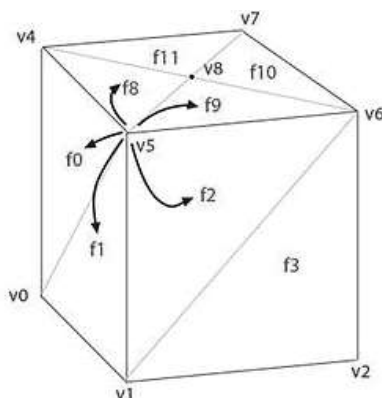


Рисунок 6 – Параллелепипед с использованием списка граней

Сетка с использованием списка граней представляет объект как множество граней и множество вершин. Это самое широко используемое представление, будучи входными данными типично принимаемыми современным графическим оборудованием.

Список граней лучше для моделирования, чем вершинное представление тем, что он позволяет явный поиск вершин грани, и граней окружающих вершину. Рис. 6 показывает пример параллелепипеда в виде сетки с использованием списка граней. Вершина V5 показывает грани, которые её окружают. У каждой грани обязательно 3 вершины. Однако это не означает что у каждой вершины одно и то же количество окружающих граней.

Для финальной визуализации грань обычно посылается в графический процессор как множество индексов вершин, и вершины посылаются как позиция-цвет-структуры нормалей. Поэтому изменения формы, но не геометрии, могут быть динамически обновлены просто переслав данные вершины без обновления связанности граней.

С сеткой использующей список граней очень легко найти вершины грани. Также, список вершин содержит список всех граней связанных с каждой вершиной. В отличие от вершинного представления и грани, и вершины явно представлены, так что нахождение соседних граней и вершин постоянно по времени. Однако ребра не заданы явно, так что поиск все ещё нужен, чтобы найти все грани, окружающие заданную грань. Другие динамические операции, такие как разрыв или объединение грани, также сложны со списком граней.

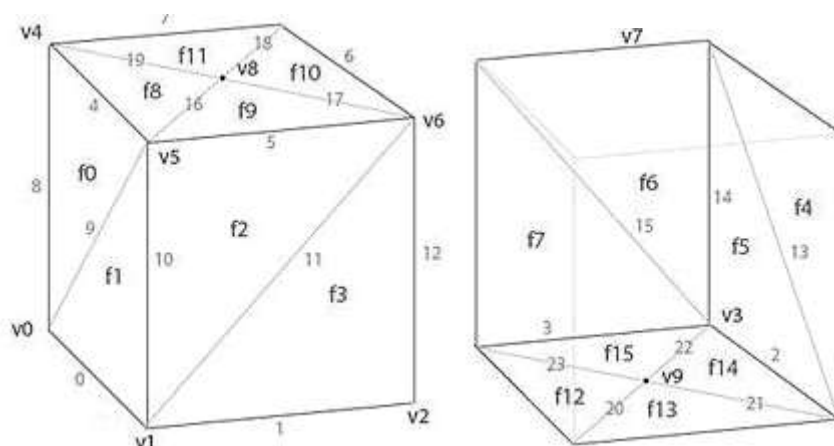


Рисунок 7 – Представление трёхмерного объекта

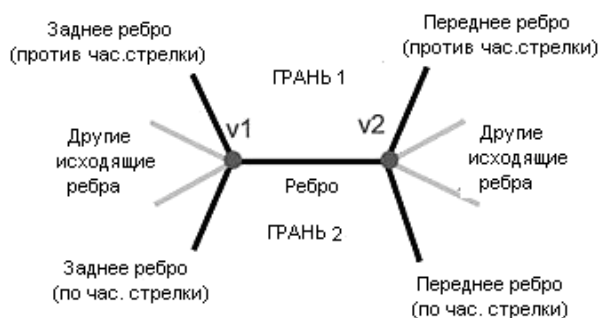


Рисунок 8 – Граф представление трёхмерного объекта

Таблица 4 – Список используемых граней

Список граней	Координаты граней
f0	4 8 9
f1	0 10 9
f2	5 10 11
f3	1 12 11
f4	6 12 13
f5	2 14 13
f6	7 14 15
f7	3 8 15
f8	4 16 19
f9	5 17 16
f10	6 18 17
f11	7 19 18
f12	0 23 20
f13	1 20 21
f14	2 21 22
f15	3 22 23

Таблица 5 – Параметры трёхмерного объекта

Ребра	Вершины	Грани	Координаты
E0	V0 V1	f1 f12	9 23 10 20
E1	V1 V2	f3 f13	11 20 12 21
E2	V2 V3	f5 f14	13 21 14 22
E3	V3 V0	f7 f15	15 22 8 23
E4	V4 V5	f0 f8	19 8 16 9
E5	V5 V6	f2 f9	16 10 17 11
E6	V6 V7	f4 f10	17 12 18 13
E7	V7 V4	f6 f11	18 14 19 15
E8	V0 V4	f7 f0	3 9 7 4
E9	V0 V5	f0 f1	8 0 4 10
E10	V1 V5	f1 f2	0 11 9 5
E11	V1 V6	f2 f3	10 1 5 12
E12	V2 V6	f3 f4	1 13 11 6
E13	V2 V7	f4 f5	12 2 6 14
E14	V3 V7	f5 f6	2 15 13 7
E15	V3 V4	f6 f7	14 3 7 15
E16	V5 V8	f8 f9	4 5 19 17
E17	V6 V8	f9 f10	5 6 16 18
E18	V7 V8	f10 f11	6 7 17 19
E19	V4 V8	f11 f8	7 4 18 16
E20	V1 V9	f12 f13	0 1 23 21
E21	V2 V9	f13 f14	1 2 20 22
E22	V3 V9	f14 f15	2 3 21 23
E23	V0 V9	f15 f12	3 0 22 20

Таблица 6 – Список вершин трехмерного объекта

Список вершин	Координаты вершин	Обход вершин
V0	0,0,0	8 9 0 23 3
V1	1,0,0	10 11 1 20 0
V2	1,1,0	12 13 2 21 1
V3	0,1,0	14 15 3 22 2
V4	0,0,1	8 15 7 19 4
V5	1,0,1	10 9 4 16 5
V6	1,1,1	12 11 5 17 6
V7	0,1,1	14 13 6 18 7
V8	0,7; 0,7; 0	16 17 18 19
V9	0,7; 0,7; 1	20 21 22 23

Граф представление было предложено Брюсом Баумгартом в 1975; где явно представляются вершины, грани и ребра сетки. Это представление широко используется в программах для моделирования динамического изменения геометрии сетки, потому что могут быть быстро выполнены операции разрыва и объединения. Их основной недостаток - высокие требования к емкости памяти и повышенная сложность из-за содержания множества индексов.

Граф представление решает проблему обхода от ребра к ребру и обеспечивает упорядоченное множество граней вокруг ребра. Для любого заданного ребра число исходящих ребер может быть произвольным. Для упрощения граф представления используются лишь четыре ближайших ребра по часовой стрелке и против часовой стрелки на каждом конце ребра. Другие ребра можно обойти по-степенно. Информация о каждом ребре напоминает ветвь, поэтому представление называется графом.

Рис. 8 показывает пример параллелепипеда в граф – представлении. Полные данные по ребру состоят из двух вершин (конечные точки), двух граней (по каждую сторону) и четырёх рёбер (граф ребра).

Финальное представление графа требует генерирования списка индексов граней. Обычно это делается только когда изменяется геометрия. Граф представление идеально подходит для динамической геометрии, такой как представление поверхностей, так как изменения сетки могут происходить локально. Обход вокруг грани сетки, может пригодиться для обнаружения столкновений, может быть эффективно выполнено тремя способами описания полигональных сеток:

Явное задание многоугольников

Каждый многоугольник можно представить в виде списка координат его вершин:

$$P = X1, Y1, Z1, X2, Y2, Z2, \dots, Xn, Yn, Zn$$

Вершины запоминаются в том порядке, в котором они встречаются при обходе вокруг многоугольника. При этом все последовательные вершины многоугольника (а также первая и последняя) соединяются ребрами. Для каждого отдельного многоугольника данный способ записи является эффективным, однако для полигональной сетки дает потери памяти вследствие дублирования информации о координатах общих вершин.

Полигональная сетка изображается путем вычерчивания ребер каждого многоугольника, однако это приводит к тому, что общие ребра рисуются дважды – по одному разу для каждого из многоугольников.

Задание многоугольников с помощью указателей в список вершин

При использовании этого представления каждый узел полигональной сетки запоминается лишь один раз в списке вершин

$$V = X1, Y1, Z1, \dots, Xn, Yn, Zn$$

Многоугольник определяется списком индексов в списке вершин. Многоугольник, составленный из вершин 3, 5, 7 и 10 этого списка, представляется как $P = (3, 5, 7, 10)$.

Такое представление имеет ряд преимуществ по сравнению с явным заданием многоугольников. Поскольку каждая вершина многоугольника запоминается только один раз, удастся сэкономить значительный объем памяти. Кроме того, координаты вершины можно легко изменять. Однако все еще не просто отыскивать многоугольники с общими ребрами. Ребра при изображении всей полигональной фигуры по-прежнему рисуются дважды. Эти две проблемы можно решить, если описывать ребра в явном виде.

Явное представление ребер

В этом представлении имеется список вершин V , однако будем рассматривать теперь многоугольник как совокупность указателей на элементы списка ребер, в котором ребра встречаются лишь один раз. Каждое ребро в списке ребер указывает на две вершины в списке вершин, определяющие это ребро, а также на один или два многоугольника, которым это ребро принадлежит. Таким образом, мы описываем многоугольник как $P = E1, \dots, E2$, а ребро – как $E = (V1, V2, P1)$. Если ребро принадлежит только одному многоугольнику, то либо $P1$ либо $P2$ – пусто.

При явном задании ребер полигональная сетка изображается путем вычерчивания не всех многоугольников, а всех ребер. В результате удастся избежать многократного рисования общих ребер. Отдельные многоугольники при этом также изображаются довольно просто [4].

И так можно сделать выводы, что в данной статье показаны методы синтеза трехмерного объекта с помощью полигональных сеток, где сетка это совокупность вершин, ребер и граней которые определяют форму многогранного объекта в трехмерной компьютерной графике и телевизионных системах. Гранями обычно являются треугольники или другие многоугольники, т.е. полигоны, так как это упрощает построение проекций в соответствии с выбранной физической моделью. Разные представления полигональных сеток используются для разных целей: моделирования, компьютерных игр, медицины и телевидения.

ЛИТЕРАТУРА

- 1 Никулин Е.А. Компьютерная геометрия и алгоритмы машинной графики / Никулин Е.А./ – Санкт-Петербург: БХВ, Санкт-Петербург, 2003. – 560 с.
- 2 Шикин А. В., Компьютерная графика. Полигональные модели / А. В. Шикин, А. В. Боресков // М.: ДИАЛОГ- МИФИ, 2001. – 464 с.
- 3 Ошаровская Е. В. Методы построение сеток в трехмерных областях. [Текст] / Е.В. Ошаровская, Н.А. Патлаенко, В.И. Солодка // Восточное европейский журнал передовых технологий – 2011. – № 5/4.– С. 8–11.
- 4 Методы построения поверхностей. [Электронный ресурс] / Винницкий национальный технический университет. - Режим доступа <http://posibnyky.vntu.edu.ua> – 2007г.
- 5 Терстон У.П. Математика трёхмерных многообразий [Электронный ресурс] //В мире науки – 1984.– № 9. – <http://www.astronet.ru/db/msg/1195055>. – 09.2010.
- 6 Compact Representations of Simplicial Meshes In Two and Three Dimensions D.K. Blandford, G. Blesloch, D. Cardoze, C. Kadow // Proc. of 12th Intern. Meshing Roundtable, Sandia National Laboratories. – sept. 2003. – P.135–146.
- 7 Прэрт У., Цифровая обработка изображения / Прэрт У.; под. ред. Д.С. Лебедева. – М.: Мир – 1998. – 480 с.