

UNIVERSITATEA TEHNICĂ A MOLDOVEI
Facultatea Calculatoare, Informatică și Microelectronică
Departamentul Informatică și Ingineria Sistemelor

GRAFICA PE CALCULATOR

ТЕМА 10. ТЕКСТУРЫ

I.u., dr. NASTAS Andrei

- 10.1. Понятия о текстурах
- 10.2. Генерация текстур фрактальным методом
- 10.3. Фильтрация текстур

10.1. Понятия о текстурах

Текстура — это изображение, визуальное отображающее совокупность свойств поверхности какого-либо объекта — реального или вымышленного.

Понятия «текстура» и «фактура» применительно к свойствам какой-либо поверхности используются синонимично. Иногда словом «фактура» называют совокупность тактильных свойств, а текстурой — совокупность свойств визуальных. Однако за цифровым графическим изображением таких свойств закрепилось слово «текстура».

Текстуры можно изучать с точки зрения:

- анализа изображений (используется в областях: сегментация изображений, классификация изображений или кодирование изображений),
- или синтеза образов.

В приложениях анализа изображения текстура является особенностью изображения, она определяет с помощью количественных показателей внешнего вида области изображения.

С точки зрения синтеза изображений, текстура сама по себе является изображением, которое может быть нанесено на определенную поверхность (плоскую или нет).

Текстурные характеристики изображения также могут быть определены путем вычисления преобразования Фурье анализируемого изображения или путем вычисления распределения уровней серого (или цветов) точек изображения.

Среди типичных областей применения текстур (в области анализа изображений) это — классификация, сегментация, кодирование.

10.1. Понятия о текстурах

1. *Классификация текстур* включает в себя такие приложения, как:

– автоматическое распознавание тканей на снимках, полученных с помощью компьютерной томографии (СТ - computer tomography) или магнитно-резонансной томографии (MR – magnetic rezonancy).

– контроль качества различной продукции (бумага, марки, печатные платы).

2. *Сегментация* – это процесс преобразования изображений, который выявляет области изображения, в пределах которых значения некоторых характерных параметров примерно одинаковы.

Текстура поверхности может быть критерием сегментации изображений, особенно если других критериев (таких как цвет или интенсивность) недостаточно для различения областей изображения.

3. *Кодировку* используют, чтобы изображения занимали как можно меньший объём памяти, сохраняя при этом всю информацию, содержащуюся в изображении, степень сжатия изображения может быть улучшена путем анализа изображения.

10.1. Понятия о текстурах

В области синтеза изображений природные объекты могут быть смоделированы с использованием различных методов генерации текстур, таких как:

- биологические объекты (тканевые поверхности, растения);
- участки, характерные для форм рельефа (горы, море);
- поверхности природных тел (например, кристаллов);
- вселенная;
- рукотворные предметы (ковры, поверхности стен зданий).

Далее будут рассмотрены текстуры применённые в этой области.

10.2. Генерация текстур фрактальным методом

10.2.1. Понятие фрактала

Понятие фрактала определяется в пределах метрического пространства.

Метрическое пространство это пара (X, d) , где X это множество (элементы которого называются точками метрического пространства), а $d: X^2 \geq R$ — функция (расстояние), которая выполняет следующие свойства:

1. (позитивность) любая $x, y \in X \Rightarrow d(x, y) = 0$,
 $d(x, y) = 0$ тогда и только тогда, когда x совпадает с y ,
2. (симметрия) любая $x, y \in X \Rightarrow d(x, y) = d(y, x)$,
3. (неравенство треугольников) любое $x, y, z \in X \Rightarrow d(x, z) = d(x, y) + d(y, z)$,

Определение: **Фрактал** — подмножество евклидова пространства, для которого размерность Хаусдорфа-Безиковича строго больше топологической размерности.

Топологическая размерность — целое число:

1. в случае кривой,
2. в случае плоскости,
3. в случае объема.

10.2.2. Самоподобные кривые

Многие такие кривые (фракталы топологического размера 1) носят имена известных математиков:

- Пеано,
- Серпинский,
- Гильберт.

Далее мы расскажем, как строятся такие кривые.

10.2.3. Кривая Коха

– Кривая Коха 0-го порядка — отрезок прямой (длиной l).

Кривая Коха является типичным геометрическим фракталом. Процесс её построения выглядит следующим образом: берём единичный отрезок, разделяем на три равные части и заменяем средний интервал равносторонним треугольником без этого сегмента. В результате образуется ломаная, состоящая из четырёх звеньев длины $1/3$. На следующем шаге повторяем операцию для каждого из четырёх получившихся звеньев и т. д... Предельная кривая и есть кривая Коха.

Свойства

- Кривая Коха нигде не дифференцируема и не спрямляема.
- Кривая Коха имеет бесконечную длину.
- Кривая Коха не имеет самопересечений.
- Кривая Коха имеет промежуточную (то есть не целую) хаусдорфову размерность, которая равна $\ln 4 / \ln 3 \approx 1,26$, поскольку она состоит из четырёх равных частей, каждая из которых подобна всей кривой с коэффициентом подобия $1/3$.
- Плоскость допускает замощение снежинками Коха двух размеров (площадь меньшей снежинки в 3 раза меньше площади большей). При этом не существует замощения снежинками одного размера.

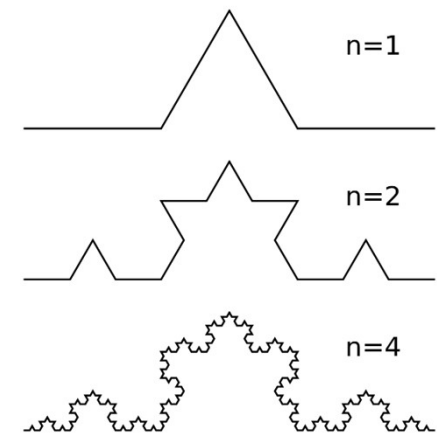


Рис. 10.1. Примеры кривых Коха

10.2.4. Кривая Пеано-Госпера

В случае кривой Пеано-Госпера, если предположить, что инициатор (кривая порядка 0) имеет длину 1, генератор (кривая порядка 1) будет включать $n = 3$ отрезков, каждый из которых имеет длину $(1/7)^{1/2}$ (коэффициент подобия $1/s$).

Функция получает в качестве параметров координаты точек A и B.

Если обозначать через $alfa$ угол CAB (зная, что углы ACD и CBD равны 120 градусам), то получим:

$$\cos(alfa) = \frac{2,5}{\sqrt{7}},$$

$$\sin(alfa) = 0,5\sqrt{\frac{3}{7}}.$$

Будут рассчитаны координаты точек C и соответственно D (вращения вокруг A и B на угол $alfa$ соответственно, некоторых точек (M, N) принадлежащих отрезку AB). На рисунке 10.2 кривая 4-го порядка имеет в качестве инициатора правильный шестиугольник.

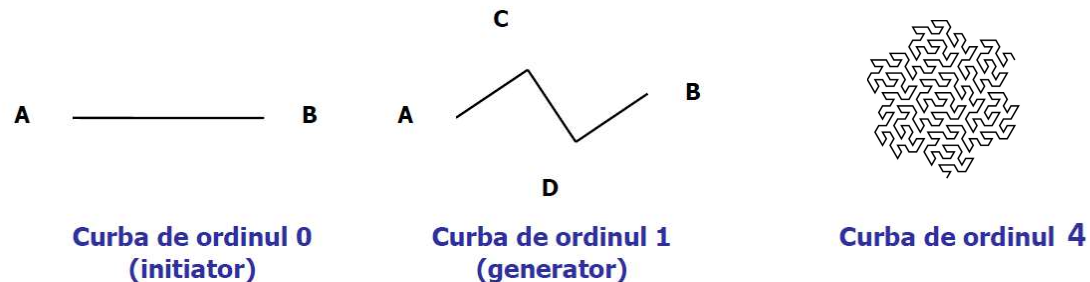


Рис. 10.2. Примеры кривых Пеано-Госпера

10.2.5. Кривая дракона (Хартера — Хейтуэя)

Для его построения берется отрезок. Поворачивается на 90 градусов вокруг одной из вершин и добавляется полученный отрезок к исходному. Получается уголок из двух отрезков. Описанная процедура повторяется. Поворачивается уголок на 90 градусов вокруг вершины и добавляется полученная ломаная к исходной.

Такая кривая обладает тем свойством, что может заполнять (укладывать) плоскую поверхность.

Случаем кривой, которая, подобно кривой дракона, покрывающей часть плоской поверхности, является кривая (типа Пеано-Госпера), генератор которой состоит из 7 копий инициатора, каждая из которых масштабируется с отношением $(1/7)^{1/2}$.

Функция, которая его генерирует, разработана с использованием техники, аналогичной той, которая используется для записи функции Госпера.



Рис. 10.3. Построение кривой дракона

10.3. Фильтрация текстур

В компьютерной графике фильтрация текстур или сглаживание текстур — это метод, используемый для определения цвета текстуры для текстурированного пикселя с использованием цветов близлежащих текселей (текстурных пикселей). Существует две основные категории фильтрации текстур, увеличения и минимизации. В зависимости от ситуации фильтрация текстур является либо типом фильтра реконструкции, в котором скудные данные интерполируются для заполнения пробелов (увеличение), либо типом сглаживания (anti-aliasing AA), где образцы текстур существуют с частотой, превышающей частоту, необходимую для частоты выборки, необходимой для заполнения текстуры (минимизация). Фильтрация описывает, как текстура применяется к фигурам, размерам, углам и масштабированию. В зависимости от выбранного алгоритма фильтрации результат покажет различные степени размытия, детализации, пространственного сглаживания, сглаживания по времени и блокировки. В зависимости от обстоятельств фильтрация может выполняться в программном обеспечении (например, в программном пакете воспроизведения) или в аппаратном обеспечении для ускоренного воспроизведения в реальном времени или графическом процессоре, или в сочетании обоих. Для большинства распространенных интерактивных графических приложений современная фильтрация текстур выполняется выделенным аппаратным обеспечением, которое оптимизирует доступ к памяти посредством кэширования и предварительной выборки, а также реализует выбор алгоритмов, доступных пользователю и разработчику.

Существует несколько методов фильтрации текстур, которые различают вычислительную сложность, пропускную способность памяти и качество изображения.

10.3. Фильтрация текстур

Во время наложения текстуры Для любой произвольной трехмерной поверхности выполняется поиск текстуры, чтобы определить, где на текстуре попадает центр каждого пикселя. Для многоугольных поверхностей с отображением текстуры, состоящих из треугольников, типичных для большинства поверхностей в трехмерных играх и фильмах, каждый пиксель (или образец подчиненного пикселя) этой поверхности будет связан с некоторым треугольником (ами) и набором барицентрические координаты, которые используются для определения положения в текстуре. Такое положение может не идеально ложиться на «пиксельную сетку», что требует некоторой функции для учета этих случаев. Другими словами, поскольку текстурированная поверхность может находиться на произвольном расстоянии и произвольной ориентации относительно наблюдателя, один пиксель обычно не соответствует непосредственно одному текселю. Чтобы определить лучший цвет для пикселя, необходимо применить какую-либо форму фильтрации. Недостаточная или неправильная фильтрация будет отображаться на изображении в виде артефактов (ошибок в изображении), таких как «блочность», неровности (jaggies – зазубрины) или мерцание.

10.3. Фильтрация текстур

Между пикселем и текселем / текселом, которые он представляет на экране, могут быть разные типы соответствия. Они зависят от положения текстурированной поверхности относительно зрителя, и в каждом случае необходимы разные формы фильтрации. Учитывая квадратную текстуру, отображаемую на квадратную поверхность в мире, на некотором расстоянии просмотра размер одного пикселя экрана в точности равен одному текселю. Ближе к этому текстели больше, чем пиксели экрана, и их необходимо соответствующим образом масштабировать - процесс, известный как увеличение текстуры. Дальше каждый текстель меньше пикселя, поэтому один пиксель покрывает несколько текстелей. В этом случае соответствующий цвет должен быть выбран на основе покрытых текстелей посредством минимизации текстуры. Графические API-интерфейсы, такие как OpenGL, позволяют программисту устанавливать различные варианты фильтров уменьшения и увеличения.

10.3.1. Мипмэппинг (Mipmapping)

Мипмэппинг - это стандартный метод, используемый для экономии некоторой работы по фильтрации, необходимой во время минимизации текстуры. Это также очень полезно для когерентности кэша - без него шаблон доступа к памяти во время выборки из удаленных текстур будет демонстрировать чрезвычайно плохую локальность, что отрицательно скажется на производительности, даже если фильтрация не выполняется.

Во время увеличения текстуры количество текселей, которые необходимо найти для любого пикселя, всегда равно четырем или меньше; однако во время минификации, поскольку текстурированный многоугольник перемещается дальше, потенциально вся текстура может упасть в один пиксель. Это потребовало бы чтения всех его текселей и объединения их значений для правильного определения цвета пикселя, что является недопустимо дорогой операцией. Mipmapping позволяет избежать этого, предварительно фильтруя текстуру и сохраняя ее в меньших размерах, вплоть до одного пикселя. По мере того как текстурированная поверхность удаляется дальше, применяемая текстура переключается на предварительно отфильтрованный меньший размер. MIP-карты разных размеров называются «уровнями», при этом уровень 0 является наибольшим размером (используется ближе всего к средству просмотра), а увеличивающиеся уровни используются с увеличением расстояния.

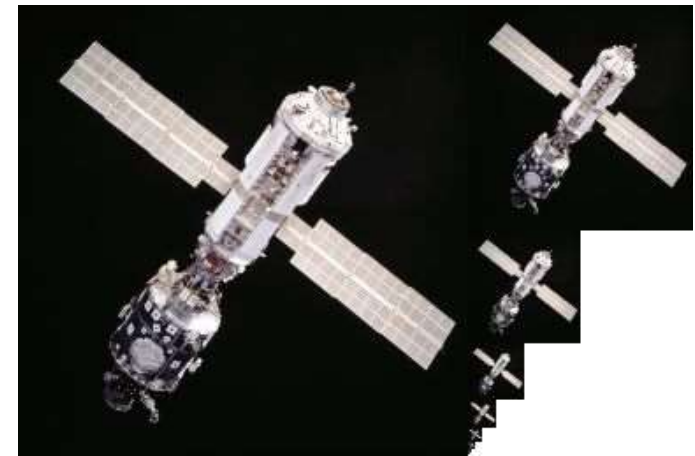


Рисунок 10.4. Пример Мипмэппинга

10.3.2. Методы фильтрации текстур

Для текстурной фильтрации используют: интерполяцию ближайшего соседа, билинейную фильтрацию, трилинейную фильтрацию, анизотропную фильтрацию, MIP-карты.

Интерполяция ближайшего соседа

Интерполяция ближайшего соседа - это самый простой и грубый метод фильтрации - он просто использует цвет текселя, ближайшего к центру пикселя, для цвета пикселя. Хотя это просто, это приводит к большому количеству артефактов - «блочности» текстуры во время увеличения, а также к сглаживанию и мерцанию во время минификации. Этот метод работает быстро во время увеличения, но во время минификации шаг по памяти становится произвольно большим, и он часто может быть менее эффективным, чем MIP-отображение из-за отсутствия пространственно согласованного доступа к текстуре и повторного использования строки кэша.

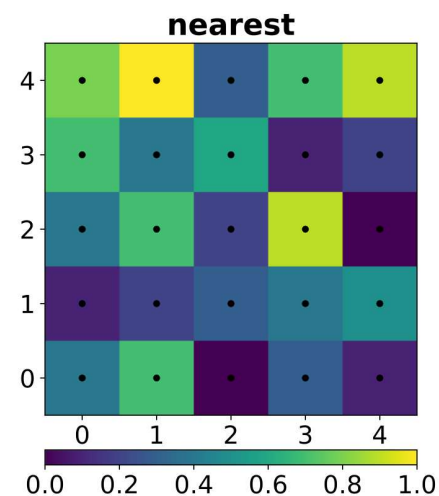


Рис. 10.5. Интерполяция ближайшего соседа

10.3.2. Методы фильтрации текстур

Линейная фильтрация MIP-карт

Поддерживает выборку ближайшего соседа из отдельных MIP-карт при линейной интерполяции двух ближайших MIP-карт, относящихся к выборке.

Билинейная фильтрация

Билинейная фильтрация является следующим шагом. В этом методе отбираются четыре текселя, ближайшие к центру пикселя (на самом близком уровне к *тіртар*), и их цвета объединяются с помощью средневзвешенного в соответствии с расстоянием.

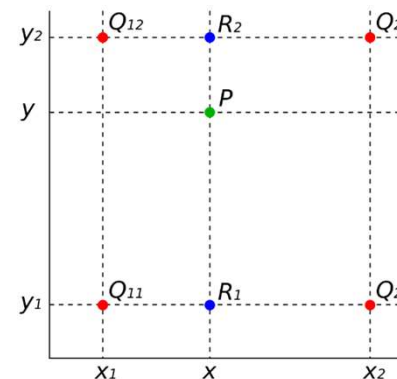


Рис. 10.6. Билинейная фильтрация

10.3.2. Методы фильтрации текстур

Трилинейная фильтрация - это средство от распространенного артефакта, наблюдаемого в билинейно отфильтрованных изображениях с мип-отображением: резкое и очень заметное изменение качества на границах, где средство визуализации переключается с одного уровня мип-карты на следующий. Трилинейная фильтрация решает эту проблему, выполняя поиск текстуры и билинейную фильтрацию на двух ближайших уровнях MIP-карты (один более высокого качества и один более низкого качества), а затем линейно интерполируя результаты. Это приводит к плавному ухудшению качества текстуры по мере увеличения расстояния от зрителя, а не к серии резких падений. Конечно, ближе к уровню 0 доступен только один уровень MIP-карты, и алгоритм возвращается к билинейной фильтрации.

10.3.2. Методы фильтрации текстур

Анизотропная фильтрация

Анизотропная фильтрация - это фильтрация высочайшего качества, доступная в современных потребительских 3D-видеокартах. В более простых, «изотропных» методах используются только квадратные MIP-карты, которые затем интерполируются с использованием би- или трилинейной фильтрации. (Изотропный означает одинаковый во всех направлениях и, следовательно, используется для описания системы, в которой все карты представляют собой квадраты, а не прямоугольники или другие четырехугольники.)



Рис. 10.7. Сравнение трилинейной и анизотропной фильтрации

ВОПРОСЫ