

Российская Академия Наук
Институт прикладной математики им. М.В. Келдыша

На правах рукописи

Переберин Антон Валерьевич

Многомасштабные методы синтеза и анализа изображений

Специальность 05.13.11 — математическое и программное обеспечение
вычислительных машин, комплексов и компьютерных сетей

АВТОРЕФЕРАТ

диссертации на соискание ученой степени
кандидата физико-математических наук

Москва — 2002

Диссертация выполнена в Институте прикладной математики
им. М.В. Келдыша РАН.

Научный руководитель — кандидат физико-математических наук
Юрий Матвеевич Баяковский.

Официальные оппоненты:

- доктор физико-математических наук, профессор
Александр Константинович Платонов,
- кандидат физико-математических наук
Андрей Серджевич Крылов.

Ведущая организация — Механико-математический факультет
МГУ им. М.В. Ломоносова (Москва).

Защита состоится «___» _____ 2002 г. в ___ часов на заседа-
нии диссертационного совета Д 002.024.01 ИПМ им. М.В. Келдыша РАН в
конференц-зале Института (125047, Москва, Миусская пл., 4).

С диссертацией можно ознакомиться в библиотеке ИПМ им. М.В. Келдыша
РАН.

Автореферат разослан «___» _____ 2002 г.

Ученый секретарь диссертационного совета
доктор физико-математических наук

Т.А. Полилова

Общая характеристика работы

Актуальность работы

Повышение эффективности обработки информации является актуальной задачей компьютерной графики. Требования к реалистичности генерируемых изображений постоянно растут, что, в конечном итоге, приводит к росту вычислительных затрат. В то же время, для многих приложений (например, игровых) необходима очень высокая скорость обработки графической информации. Рост производительности оборудования решает эту проблему, как показывает практика, лишь отчасти.

Один из путей повышения эффективности обработки информации — применение методов, основанных на *многомасштабном* представлении графических объектов. Многомасштабное представление — это многослойная структура, первый слой которой содержит информацию, достаточную для грубого (с низким разрешением) приближения объекта; при добавлении информации из каждого последующего слоя степень детализации постепенно увеличивается, пока объект не будет восстановлен полностью (то есть с максимальным разрешением).

С помощью методов, основанных на многомасштабном представлении, может быть решен широкий круг задач синтеза, анализа и обработки графических объектов. Кроме того, эти методы обеспечивают сокращение объемов данных за счет удаления избыточной и несущественной информации, снижая, тем самым, вычислительные затраты на последующую обработку. Алгоритмы обработки многомасштабных представлений, основанные на *вейвлет-анализе* (или *анализе всплесков*), достаточно просты и эффективны в реализации.

Цель работы

Разработка многомасштабных методов анализа и синтеза графических объектов разной структуры. Изучение возможностей адаптации этих методов и реализующих их алгоритмов к особенностям конкретных задач и требованиям приложений. Осуществление с помощью указанных методов процессов многоэтапной обработки графической информации.

Научная новизна работы

В работе рассмотрены новые способы решения нескольких задач компьютерной графики, основанные на многомасштабном представлении информации. Предложен метод адаптивной триангуляции на основе дерева вейвлет-преобразований и его модификация для решения задачи расчета и представления освещенности. Предложено применение В-сплайнового вейвлет-преобразования для обработки и отображения линий уровня. Предложена модель описания стохастических текстур, близкая по структуре к разложению сигнала по вейвлет-базису.

Новым является комплексный подход к применению многомасштабных методов для задач, требующих многоэтапной обработки информации. Предлагается использовать одно и то же представление для решения возможно большего числа подзадач. Такой подход упрощает общую процедуру обработки и повышает эффективность ее реализации. Кроме того, становится возможным расширять функциональные возможности метода, внося в него минимум дополнений.

Дополнительно можно отметить, что в процессе разработки модели текстур была сделана заявка на новое, «функциональное» расширение вейвлет-преобразования. (Однако изучение свойств, возможностей, способов реализации и класса приложений такого расширения является предметом самостоятельного исследования).

Практическая значимость

Разработаны и доведены до реализации методы решения нескольких актуальных задач компьютерной графики. Реализованные алгоритмы удовлетворяют требованиям и ограничениям, которые были сформулированы при постановке задач. В частности, алгоритм генерации и нанесения текстур разрабатывался с учетом возможной аппаратной реализации в графических ускорителях. Метод построения изолиний внедрен в программный продукт, разработанный для геологических расчетов.

Апробация работы и публикации

Результаты работы докладывались и обсуждались на:

- 10-й международной конференции по компьютерной графике и машинному зрению GraphiCon'2000 “Fast Multi-Scaled Texture Generation and Rendering” («Быстрая многоуровневая генерация и отображение текстур»), Россия, Москва, 2000;
- 9-й международной конференции по компьютерной графике и машинному зрению GraphiCon'99 “Hierarchical Approach for Texture Compression” («Иерархический подход к сжатию текстур»), Россия, Москва, 1999;
- 7-й международной конференции по компьютерной графике и машинному зрению GraphiCon'97 “From Photon Map to Irradiance Function via Wavelet Transform” («От фотонных карт к функции освещенности через вейвлет-преобразование»), Россия, Москва, 1997;
- семинаре по компьютерной графике и обработке изображений Ю.М. Баяковского (ф-т ВМиК МГУ).

Основные результаты работы изложены в 5-и научных публикациях.

Структура и объем работы

Диссертация состоит из введения, четырех глав, заключения, списка литературы и приложения. Содержание работы изложено на 138 страницах (включая 12 страниц приложения). Список литературы включает 78 наименований. В работе содержится 19 рисунков и 3 таблицы.

Содержание работы

Во введении обосновывается тема диссертации, ее актуальность и практическая значимость, приводится обзор литературы по теме, кратко излагается содержание работы.

Первая глава содержит краткое описание аппарата вейвлет-преобразований одно- и двумерных дискретных сигналов, который применяется при решении задач, рассматриваемых в последующих главах.

Вейвлет-преобразование (ВП) одномерного дискретного сигнала $\mathbf{s} = \{s_j\}$, $j \in \mathbf{Z}$, выполняется по следующим рекурсивным формулам:

$$\begin{aligned} \mathbf{v}_i = \downarrow_2 [\mathbf{v}_{i+1} * \tilde{\mathbf{h}}], \quad \mathbf{w}_i = \downarrow_2 [\mathbf{v}_{i+1} * \tilde{\mathbf{g}}], \quad i = i_1 - 1, i_1 - 2, \dots, i_0; \\ \mathbf{v}_{i_1} = \mathbf{s}. \end{aligned} \quad (1)$$

Знак “*” обозначает операцию свертки, оператор $\downarrow_2 [\mathbf{x}]$ удаляет из сигнала \mathbf{x} каждый второй элемент. Фильтры $\tilde{\mathbf{h}}$ и $\tilde{\mathbf{g}}$ называются соответственно *НЧ* и *ВЧ фильтрами анализа*.

Индекс i называется *уровнем разрешения* или *разрешением*. Исходному сигналу ставится в соответствие некоторое произвольное значение разрешения i_1 .

На каждом шаге ВП сигнал распадается на две составляющие: приближение с более низким разрешением \mathbf{v}_i (НЧ составляющую) и детализирующую информацию \mathbf{w}_i (ВЧ составляющую). Элементы ВЧ составляющих ВП называют *вейвлет-коэффициентами*.

Разрешение i_0 , до которого выполняется преобразование, теоретически может быть любым, меньшим i_1 . Для случая конечных сигналов самым низким разрешением является то, при котором НЧ составляющая состоит из единственного элемента (сокращение количества элементов происходит под действием оператора $\downarrow_2 [\bullet]$).

Обратное ВП, или *восстановление* сигнала \mathbf{s} выполняется по формуле:

$$\mathbf{v}_{i+1} = \uparrow_2 [\mathbf{v}_i * \mathbf{h}] + \uparrow_2 [\mathbf{w}_i * \mathbf{g}], \quad i = i_0, i_0 + 1, \dots, i_1 - 1, \quad (2)$$

Оператор $\uparrow_2 [\mathbf{x}]$ осуществляет вставку нулевого элемента между элементами сигнала \mathbf{x} . Фильтры \mathbf{h} и \mathbf{g} называются соответственно *НЧ* и *ВЧ фильтрами синтеза*.

Очевидно, что формула (2) обеспечивает реальное восстановления сигнала только для фильтров, для которых:

$$\forall \mathbf{x} \quad \uparrow_2 [\downarrow_2 [\mathbf{x} * \tilde{\mathbf{h}}]] * \mathbf{h} + \uparrow_2 [\downarrow_2 [\mathbf{x} * \tilde{\mathbf{g}}]] * \mathbf{g} = \mathbf{x}.$$

Таким образом, конкретное ВП определяется четверкой фильтров $\tilde{\mathbf{h}}$, $\tilde{\mathbf{g}}$, \mathbf{h} и \mathbf{g} .

Простейшим ВП является *преобразование Хаара*, которому соответствуют следующие фильтры:

$$\begin{aligned} \tilde{\mathbf{h}} &= [1/2, 1/2], & \tilde{\mathbf{g}} &= [-1/2, 1/2]; \\ \mathbf{h} &= [1, 1], & \mathbf{g} &= [1, -1]. \end{aligned}$$

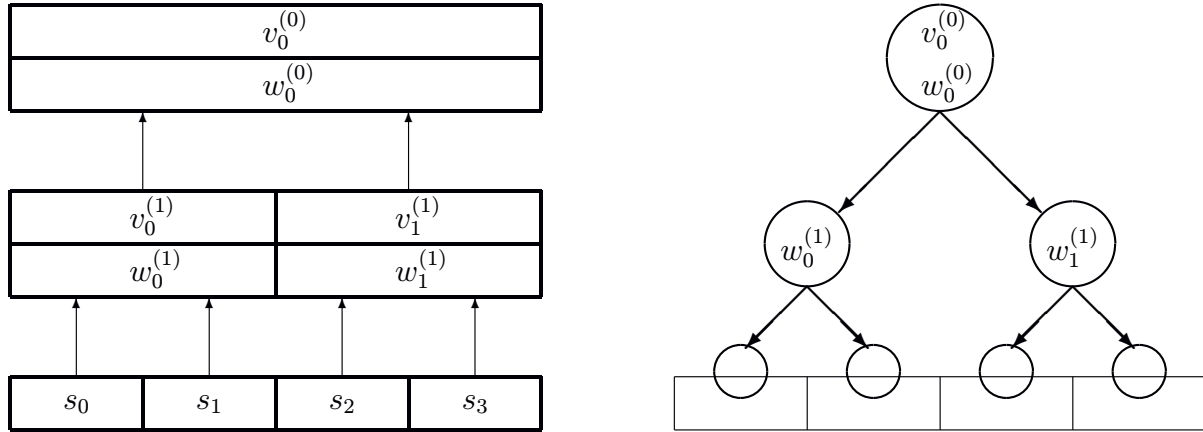


Рис. 1: Структура вейвлет-преобразования (слева) и дерево преобразования (справа).

Структура ВП такова, что каждой паре элементов сигнала, представленного с разрешением $i + 1$, ставится в соответствие один элемент НЧ составляющей разрешения i и один вейвлет-коэффициент того же разрешения (рис. 1, слева). Если построить граф таких соответствий (начиная с низшего уровня разрешения), то для конечных сигналов получится сбалансированное бинарное дерево, а в общем случае лес сбалансированных бинарных деревьев. Вершине дерева приписывается соответствующий НЧ элемент и вейвлет-коэффициент; остальным вершинам, кроме листовых, приписываются соответствующие вейвлет-коэффициенты; листовым вершинам, соответствующим элементам исходного сигнала, не приписывается ничего. Такое дерево будем называть *деревом преобразования* (рис. 1, справа).

Двумерное ВП может быть реализовано композицией одномерных. Пусть сигнал \mathbf{s} — матрица конечного или бесконечного размера. Применим к каждой строке матрицы один шаг одномерного ВП. В результате получится две матрицы, строки которых содержат НЧ и ВЧ составляющие строк исходной матрицы. К каждому столбцу обеих матриц также применим шаг одномерного ВП. В результате получается четыре матрицы, которые можно обозначить как НЧ/НЧ, НЧ/ВЧ, ВЧ/НЧ и ВЧ/ВЧ. Первая является НЧ составляющей исходного сигнала, остальные три содержат детализирующую информацию.

Во второй главе описывается алгоритм построения адаптивных треугольных сеток для представления графических объектов, параметризуемых на плоскости. Алгоритм основан на древовидной структуре многомас-

штабного анализа информации. Рассматривается модификация этого алгоритма для приложения — реконструкции функции освещенности по данным, полученным с помощью прямой трассировки лучей методом Монте-Карло.

В п. 1 говорится об актуальности задачи построения адаптивных треугольных сеток (т.е. сеток, учитывающих структуру объекта) для представления графических объектов, определяемых двумерными сигналами.

В п. 2 задача построения адаптивных сеток решается с помощью вейвлет-анализа, а именно, на основе древовидного представления двумерного ВП.

Пусть в качестве исходных данных имеется двумерный дискретный сигнал. Этот сигнал есть результат оцифровки некоторой поверхности, выполненной с фиксированным шагом, то есть элементы сигнала суть значения в узлах регулярной решетки. Если применить к исходному сигналу один шаг двумерного ВП, то полученную в результате НЧ составляющую можно также рассматривать, как результат оцифровки той же поверхности, но с удвоенным шагом (то есть с меньшим разрешением). Следующий шаг преобразования даст решетку еще более низкого разрешения и т.д.

Структура двумерного преобразования такова, что на каждом шаге четырем соседним элементам преобразуемого сигнала ставится в соответствие один элемент НЧ составляющей и три вейвлет-коэффициента. Дерево такого преобразования является квадродеревом. Корню приписывается одно значение НЧ составляющей и три вейвлет-коэффициента, необходимых для восстановления следующего уровня разрешения. Всем остальным вершинам, кроме листовых, также приписывается по три вейвлет-коэффициента.

Координаты на плоскости узла уровня разрешения i являются средним арифметическим координат четырех соответствующих ему узлов уровня $i + 1$. На рис. 2 показано взаимное расположение узлов решеток трех соседних уровней разрешения.

Далее производится операция, аналогичная той, которая имеет место при сжатии данных с помощью вейвлетов. Из дерева преобразований удаляются поддеревья, к которым приписаны вейвлет-коэффициенты, значения которых удовлетворяют некоторому критерию (например, их абсолютные величины меньше заданного порога). Такие поддеревья называют

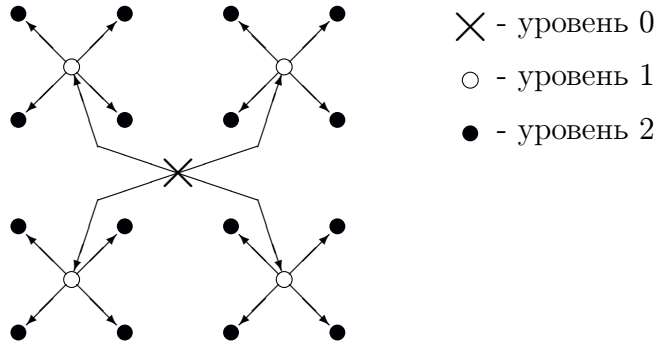


Рис. 2: Узлы решеток уровней 0, 1 и 2 (стрелками показаны ребра квадродрева).

«нуль-деревьями», поскольку удаление поддерева из дерева преобразований эквивалентно обращению в нуль всех приписанных поддереву вейвлет-коэффициентов. Обратное преобразование по модифицированному дереву производится так, чтобы восстановить значения только в тех узлах, которые соответствуют листовым вершинам этого дерева. Очевидно, что в таком случае будут восстановлены узлы вообще говоря разных уровней разрешения, что и обеспечит построение сетки, адаптированной к особенностям конкретного сигнала.

Триангулировать полученный набор нерегулярно расположенных узлов предлагается следующим образом. Создается двумерный массив ссылок на вершины, который имеет в точности такой же размер, что и исходная решетка. Если узлу решетки соответствует листовая вершина дерева, то в соответствующий узлу элемент массива заносится ссылка на эту вершину; если узлу решетки соответствует вершина нуль-дерева, то в элемент массива заносится ссылка на корень этого нуль-дерева. Таким образом, массив заполняется ссылками на листовые вершины дерева, причем часть элементов массива содержит ссылки на одни и те же вершины (кроме случая, когда нуль-деревья отсутствуют). Далее по полученному массиву строится триангуляция, как если бы триангулировалась регулярная решетка (эта операция тривиальна). В этом случае часть треугольников будет иметь по две или по три совпадающие вершины (за счет повторяющихся ссылок в массиве). Такие треугольники просто игнорируются. Оставшиеся же треугольники и образуют искомую сетку (рис. 3). Некоторые результаты работы метода показаны на рис. 4.

В п. 3 формулируется задача реконструкция функции освещенности на основе данных, полученных в результате прямой трассировки лучей мето-

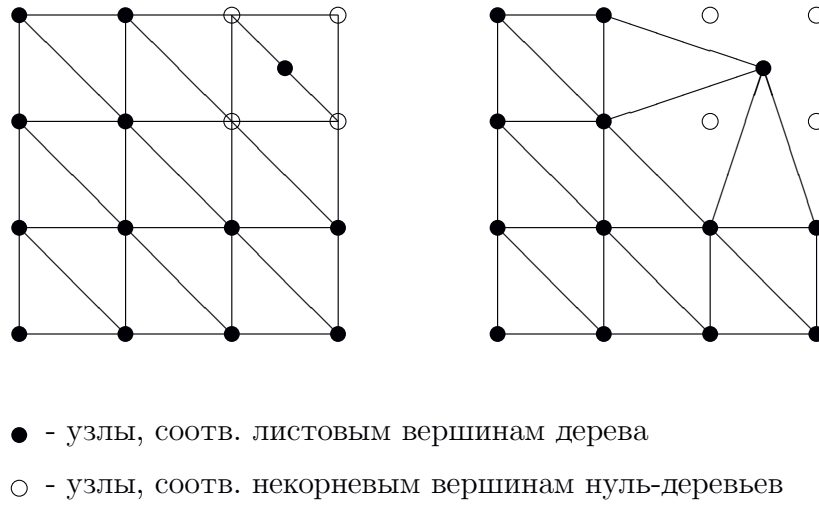


Рис. 3: Триангуляция массива ссылок (слева) и итоговая сетка (справа).

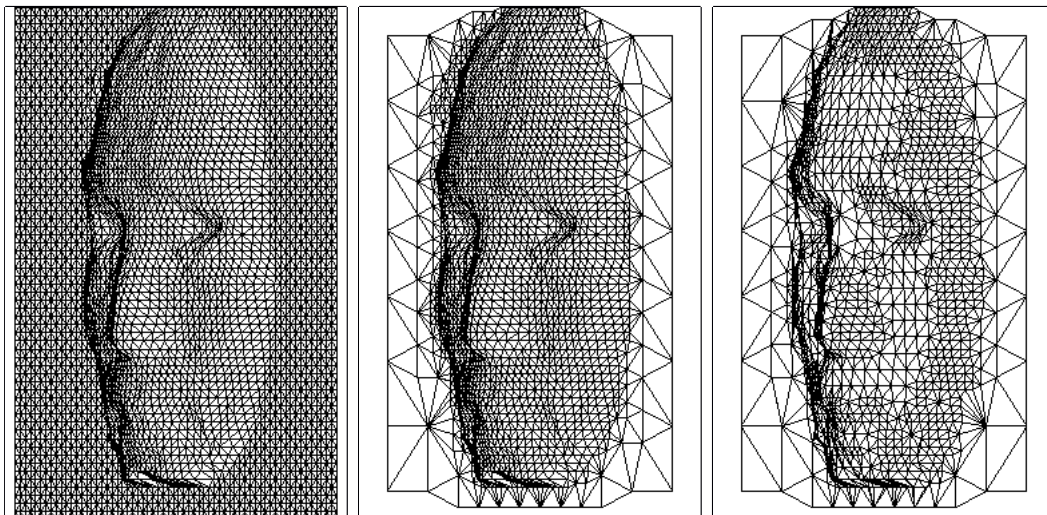


Рис. 4: Регулярная сетка максимального разрешения (слева), и адаптивные сетки, построенные с помощью дерева вейвлет-преобразования, с разным значением порога.

дом Монте-Карло.

Метод Монте-Карло трассировки лучей — распространенный метод синтеза фотореалистических изображений, моделирующий световой поток испусканием конечного числа лучей-фотонов, распространение которых определяется случайными величинами с заданными распределениями. При попадании луча на поверхность некоторого объекта сцены след луча регистрируется в специальной структуре данных. Освещенность поверхности является функцией плотности лучей, попавших на эту поверхность. Задача восстановления плотности распределения по конечной выборке известна как *задача оценки плотности*.

Функцию освещенности можно получить в виде растрового изображения (текстуры), однако в ряде случаев предпочтительно представление в виде треугольной адаптивной сетки. При этом интересна задача не просто построить сетку по рассчитанной каким-либо методом текстуре, а объединить фазы оценки плотности и построения адаптивной сетки в единый процесс.

В п. 4 предложен метод реконструкции и построения адаптивного сеточного представления функции освещенности на основе метода, рассмотренного в п. 2.

Вместо хранения вейвлет-преобразования сигнала, предлагается хранить в явном виде все полученные решетки. Дерево преобразований при этом практически не меняется, только его вершинам приписываются не вейвлет-коэффициенты, а значения в соответствующих узлах в явном виде.

Представление данных в таком виде позволяет существенно расширить круг используемых сглаживающих фильтров. При выборе фильтра можно не заботиться об обратимости преобразования, так как последнее не требуется. В частности, оказывается возможным применение неоднородных схем (то есть разных фильтров на одном уровне разрешения).

В качестве начального приближения используется следующий сигнал. На поверхности строится решетка с постоянным шагом, определенным пользователем. Каждый узел решетки получает значение — количество фотонов, которые оказались ближе всего к данному узлу. Другой вариант — распределять энергию фотона между четырьмя ближайшими к нему узлами.

Шаг прямого преобразования реализуется не композицией одномерных

шагов, а сверткой с двумерным НЧ фильтром-матрицей, с последующим применением оператора $\downarrow_2 [\bullet]$ к строкам и столбцам полученного сигнала. ВЧ составляющие не вычисляются.

Вместо обычной однородной свертки возможно применение более сложной операции: для расчета значений в каждом узле выбирается такой фильтр, чтобы суммарная энергия фотонов, оказавшихся в зоне действия фильтра, была не меньше заданной. То есть, если плотность фотонов вблизи рассматриваемого узла высока, достаточно воспользоваться фильтром малого размера, что будет препятствовать излишней размытости изображения. Если же плотность фотонов низка, то маленький фильтр не обеспечит достаточного шумопонижения, требуется фильтр большего размера. В настоящей реализации количество фильтров, одновременно участвующих в преобразовании, ограничивается тремя («малый», «средний» и «большой»). Ниже приводится одна из возможных троек фильтров:

$$\frac{1}{32} \begin{bmatrix} 1 & 2 & 2 & 1 \\ 2 & 3 & 3 & 2 \\ 2 & 3 & 3 & 2 \\ 1 & 2 & 2 & 1 \end{bmatrix}; \quad \frac{1}{124} \begin{bmatrix} 1 & 2 & 3 & 3 & 2 & 1 \\ 2 & 4 & 5 & 5 & 4 & 2 \\ 3 & 5 & 6 & 6 & 5 & 3 \\ 3 & 5 & 6 & 6 & 5 & 3 \\ 2 & 4 & 5 & 5 & 4 & 2 \\ 1 & 2 & 3 & 3 & 2 & 1 \end{bmatrix}; \quad \frac{1}{344} \begin{bmatrix} 1 & 2 & 3 & 4 & 4 & 3 & 2 & 1 \\ 2 & 5 & 6 & 7 & 7 & 6 & 5 & 2 \\ 3 & 6 & 8 & 9 & 9 & 8 & 6 & 3 \\ 4 & 7 & 9 & 10 & 10 & 9 & 7 & 4 \\ 4 & 7 & 9 & 10 & 10 & 9 & 7 & 4 \\ 3 & 6 & 8 & 9 & 9 & 8 & 6 & 3 \\ 2 & 5 & 6 & 7 & 7 & 6 & 5 & 2 \\ 1 & 2 & 3 & 4 & 4 & 3 & 2 & 1 \end{bmatrix}.$$

Если в базовом алгоритме, описанном в п. 2, критерии отсечения нуль-деревьев основаны на оценке вейвлет-коэффициентов, то в данной модификации оцениваются разности значений в узле решетки разрешения i и четырех соответствующих ему узлах решетки разрешения $i + 1$. Если максимум модуля этих четырех разностей не превышает заданного порога, то вершина объявляется корнем нуль-дерева.

На рис. 5 показан пример исходных данных, построенная по ним сетка и восстановленная функция освещенности.

В п. 5 анализируются результаты проделанной работы. Большинство методов упрощения сеточных представлений основаны на прореживании исходной сетки, т.е., пользуясь терминологией теории обработки сигналов, осуществляет точечную выборку. Предложенный метод создает новую сетку, вершины которой являются взвешенной выборкой вершин исходной сетки. Это улучшает аппроксимирующие свойства метода, кроме того, появля-

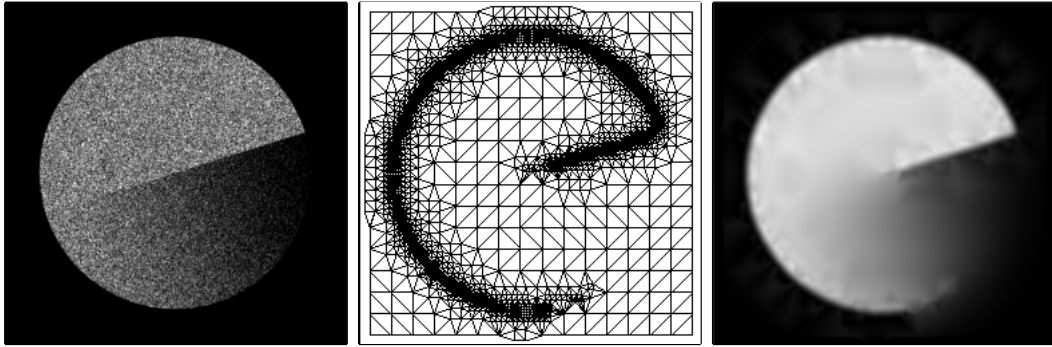


Рис. 5: Восстановление освещенности. Результат попадания 200 тыс. фотонов (слева), адаптивная сетка (в центре) и итоговое изображение (справа).

ется возможность на этапе построения сетки выполнять дополнительную обработку сигнала, например, сглаживание.

Метод лишен таких недостатков алгоритмов адаптивной триангуляции, как сложные критерии выбора удаляемых вершин, необходимость повторной локальной триангуляции, возможность «перехлеста» треугольников после удаления вершины и пр.

Базовый вариант метода может быть применен для широкого круга задач, например, для описания виртуальных ландшафтов. Он основан на обычном двумерном вейвлет-преобразовании, которое допускает достаточно эффективную реализацию и, после незначительных модификаций алгоритма, позволяет ввести дополнительные операции: автоматическое управление уровнем детализации, прогрессивную передачу объекта и др.

Вариант, разработанный для восстановления функции освещенности, кроме построения сетки выполняет специальную обработку сильно зашумленного сигнала — начального приближения искомой функции. Для реализации этого варианта выбрана схема, близкая к построению пирамиды гауссианов. Она требует большего расхода памяти, но снимает практически все ограничения на способы вычисления НЧ составляющих, что является особенно важным при решении данной задачи.

В отличие от других методов построения функции освещенности данный подход объединяет в единый процесс фазу собственно восстановления функции и фазу построения для этой функции сеточного представления, используя при этом одну и ту же структуру данных — дерево узлов. Это упрощает процедуру обработки и сокращает общий объем вычислений.

На примере реализации двух вариантов одного метода показано, как,

учитывая особенности конкретной задачи, можно модифицировать представление объекта и алгоритм его обработки в пользу того или иного свойства.

В третьей главе рассматривается применение вейвлет-анализа для решения задачи построения линий уровня (изолиний) на плоскости.

В п. 1 формулируется задача построения линий уровня (изолиний) с учетом некоторых дополнительных требований, предъявляемых к приложению. Например, исходные данные, по которым строятся линии, могут быть искажены шумом, и построенная по таким данным линия требует сглаживания. В современных операционных системах с графическим интерфейсом отображение объектов, как правило, осуществляется в области («окна») произвольного и легко меняющегося пользователем размера, поэтому желательно, чтобы объект мог быстро масштабироваться и перерисовываться в окне заданного размера.

В п. 2 описывается распространенный алгоритм нахождения управляющей последовательности, т.е. последовательности точек, по которым в дальнейшем будет построена кривая.

Исходными данными является таблично заданная функция

$$z(i, j) = z_{i,j}, \quad i = \overline{0, M_x - 1}, \quad j = \overline{0, M_y - 1}.$$

Требуется построить линию уровня некоторого значения Z этой функции.

Для любого квадрата, вершины которого имеют координаты (i, j) , $(i + 1, j)$, $(i, j + 1)$, $(i + 1, j + 1)$, $i = \overline{0, M_x - 2}$, $j = \overline{0, M_y - 2}$, можно определить, проходит ли через него линия (для этого значения в вершинах сравниваются с Z). Если линия проходит через квадрат, то ищется и запоминается точка пересечения линии с одной из сторон квадрата, и через эту сторону происходит переход в соседний квадрат. В этом квадрате ищется следующая точка пересечения, и происходит дальнейший переход. Так формируется управляющая последовательность.

В п. 3 рассматривается задача построения линии на основе имеющейся последовательности точек. В ряде случаев достаточно построить по этим точкам ломаную, но в общем случае такой подход не даст удовлетворительного результата. Решено строить по точкам более гладкую кубическую В-сплайновую кривую. Но, как нетрудно показать, даже такая кривая, построенная непосредственно по исходной последовательности точек, не всегда даст нужный результат. Возникает необходимость дополнительной обработки управляющей последовательности. Для этого цели предлагается

применять В-сплайновое ВП.

В-сплайновая кривая $\gamma(t)$ — задается в виде:

$$\gamma(t) = \sum_{j=0}^{N-1} \mathbf{c}_j \mathcal{B}_j(t), \quad t \in [0, 1], \quad (3)$$

где \mathbf{c}_j — управляющие точки, а функции $\mathcal{B}_j(t)$ — *элементарные В-сплайны*.

Представление В-сплайновых кривых в форме (3) позволяет применить к управляющей последовательности (которая является дискретным сигналом) В-сплайновое ВП. Например, кубическое В-сплайновое ВП (ему соответствуют кубические элементарные В-сплайны) реализуется фильтрами:

$$\begin{aligned} \tilde{\mathbf{h}} &= \left[\frac{3}{32} \quad -\frac{3}{8} \quad \frac{5}{32} \quad \frac{5}{4} \quad \frac{5}{32} \quad -\frac{3}{8} \quad \frac{3}{32} \right], \quad \tilde{\mathbf{g}} = \left[\frac{1}{16} \quad -\frac{1}{4} \quad \frac{3}{8} \quad -\frac{1}{4} \quad \frac{1}{16} \right]; \\ \mathbf{h} &= \left[\frac{1}{8} \quad \frac{1}{2} \quad \frac{3}{4} \quad \frac{1}{2} \quad \frac{1}{8} \right], \quad \mathbf{g} = \left[-\frac{3}{16} \quad -\frac{3}{4} \quad -\frac{5}{16} \quad \frac{5}{2} \quad -\frac{5}{16} \quad -\frac{3}{4} \quad -\frac{3}{16} \right]. \end{aligned} \quad (4)$$

Полученная на любом шаге прямого преобразования НЧ составляющая также оказывается управляющей последовательностью некоторой В-сплайновой кривой, которая является более грубым приближением исходной кривой.

Применение шагов прямого преобразования приводит к сокращению числа управляющих точек и потери части ВЧ информации — таким образом реализуется сглаживание кривой.

Выводить на графическое устройство предлагается не саму кривую, а приближающую ее ломаную. Очевидно, что, принимая во внимание конечное разрешение любого реального графического устройства, ломаная, надлежащим образом приближающая кривую, всегда должна существовать. Для В-сплайновой кривой это может быть ломаная, построенная по управляющим точкам, при условии, что соседние точки расположены достаточно близко друг у другу. Такую ломаную можно получить, если применять к кривой шаги обратного преобразования с нулевой ВЧ составляющей. При этом сам объект (кривая) не меняется, но количество управляющих точек растет, и они располагаются ближе к самой кривой. Достаточно потребовать выполнять шаги прямого преобразование до тех пор, пока максимум расстояний между парами соседних точек не окажется меньше заданной величины (определяемой характеристиками графического устройства), и затем просто построить по полученной последовательности ломаную. Определение нужного числа шагов происходит автоматически в зависимости от

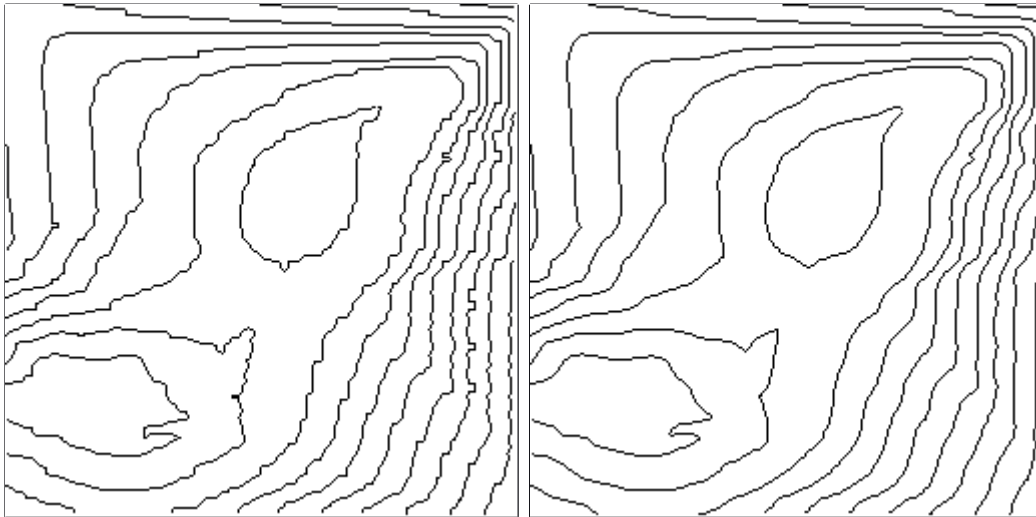


Рис. 6: Изолинии, построенные по исходным данным (слева) и после сглаживания (справа).

размера области вывода. Возможна реализация быстрого пересчета ломаной для отображения кривой в нужном масштабе, например, при изменении пользователем размеров окна графического вывода. Предложенный способ отображения применим и для векторных, и для растровых устройств.

Таким образом, с помощью вейвлет-анализа удалось решить все требуемые подзадачи, а именно, сглаживание кривой, выбор оптимально масштаба для отображения в область с заданными характеристиками и собственно отображение.

В п. 4 описаны некоторые результаты работы разработанного и реализованного метода и подводятся итоги проделанной работы. На рис. 6 слева показаны линии, построенные по исходным данным без дополнительной обработки, справа линии, полученные в результате сглаживания. В обоих случаях отображены управляющие ломаные, приближающие кубические В-сплайнове кривые.

Описанный выше способ позволяет не просто строить линии уровня таблично заданной функции, но и осуществлять дополнительную обработку (сглаживание) этих линий. Кроме того, он обеспечивает выбор точек, реально необходимых для отображения линии при заданных параметрах области отображения.

Важная особенность предложенного метода — использование одного представления и одного аппарата обработки (вейвлет-анализа) для решения всех поставленных подзадач обработки объекта. Это обеспечивает не

только эффективную реализацию метода, но и позволяет расширить набор возможных операций (например, ввести локальную обработку произвольного фрагмента кривой), внося в алгоритм минимум дополнений.

Метод не требует больших вычислительных затрат. Вычисления легко распараллеливаются: прежде всего, каждая кривая обрабатывается независимо от другой, кроме того, вычисление вейвлет-преобразований может выполняться параллельно для разных фрагментов кривой (прямое следствие пространственной локализации вейвлетов). Возможна аппаратная реализация метода (что актуально для фазы отображения).

В четвертой главе рассматривается многомасштабная модель для описания некоторого класса стохастических текстур. С помощью этой модели возможно создание как реалистических, так и «абстрактных» изображений-текстур.

В п. 1 формулируется задача генерации стохастических текстур, определяются требования, которым должны удовлетворять модель и алгоритм генерации:

- Представление текстуры должно быть существенно более компактным, чем изображение той же текстуры в явном виде, сжатое с помощью какого-либо алгоритма сжатия текстур.
- У генерируемой текстуры не должен наблюдаться периодический эффект, который возникает, если текстура получена простым размножением по поверхности объекта некоторого фиксированного изображения (образца).
- Представление должно допускать локальную генерацию (это означает, что для получения произвольного фрагмента текстуры не должно требоваться полное декодирование представления).
- Легко реализуется масштабирование текстуры.
- Алгоритм генерации текстуры по ее представлению должен работать в реальном времени и допускать аппаратную реализацию.

В п. 2 описывается иерархическая модель представления текстур.

В основе модели лежит идея о «детерминированно-случайной» природе стохастической текстуры. Обе компоненты такой текстуры предлагается

задавать в явном виде: детерминированную компоненту в виде изображения, а случайную — в виде распределения.

Базовым элементом называется некоторое произвольное растровое изображение.

Репликацией назовем одну копию (возможно, масштабированную и преобразованную перечисленными ниже способами) базового элемента, которая помещается в выходное изображение.

Будем говорить, что репликация помещена в некоторую точку выходного изображения, если в этой точке окажется начало, т.е. левый верхний угол репликации. Точку выходного изображения, в которую с отличной от нуля вероятностью может быть помещена репликация, назовем *точкой репликации*.

Предположим, что базовый элемент — квадратное растровое изображение с размером стороны $N = 2^I$ точек. Тогда масштабированные версии базового элемента — это квадратные растровые изображения с размерами сторон 2^i , $i = \overline{1, I}$ точек. Индекс i назовем *уровнем разрешения* или просто *разрешением*.

Координаты точек репликации одного уровня разрешения кратны половине размера стороны соответствующих репликаций.

В каждой точке репликации могут иметь место следующие *события*:

- *Позитивная репликация* (простое наложение), *негативная репликация* (наложение с инверсией), *нет репликации*.
- *Поворот* элемента на 0° , 90° , 180° , 270° .
- *Отражение* элемента, *нет отражения*.

Для каждой конкретной модели определяется вероятность каждого из этих событий в точках репликации. Возможны комбинации событий, например, негативная репликация с поворотом на 90° без отражения.

Базовый элемент может иметь точки как с положительной, так и с отрицательной интенсивностью. Точка с нулевой интенсивностью считается прозрачной. Нулевую интенсивность имеет фон базового элемента.

На первом шаге выходное изображение — это прямоугольник заданного пользователем размера с нулевой интенсивностью (прозрачный).

Далее в изображение добавляются репликации. Операция добавления может быть выражена обычным сложением интенсивностей точек формируемого изображения и репликации, но допустимы и другие способы наложения одного изображения на другое (в этом случае вероятности применения каждой из доступных операций также определяются для конкретной модели).

Репликации одного уровня разрешения формируют *слой* выходного изображения. Каждому слою может соответствовать один *весовой коэффициент*, который управляет степенью значимости слоя в формируемом изображении. Добавлять репликации в изображение предлагается послойно.

Последним шагом является добавление к каждой точке изображения фиксированного значения интенсивности фона.

Для задания конкретной модели требуется определить количество слоев и таблицы вероятностного распределения событий в точках репликации каждого слоя (*таблицы событий*). Каждая ячейка таблицы соответствует одной точке репликации. Размер таблицы задается пользователем. Генерация слоя любого размера обеспечивается простым размножением таблицы событий по вертикали и горизонтали в требуемом количестве.

Базовый элемент может либо включаться, либо не включаться в описание модели. Последнее значит, что модель предназначена для работы с различными базовыми элементами.

При описании модели также следует указать интенсивность фона всего изображения, весовые коэффициенты каждого слоя и порядок послойного формирования изображения (сверху-вниз или снизу-вверх).

Предлагаемая модель обеспечивает очень простой способ *масштабирования* изображений (в масштабе 1:2, 1:4, 1:8 и т.д.) Для этого достаточно лишь изменить соответствие между слоем изображения и разрешением базового элемента.

В п. 3 рассматриваются примеры текстур, полученных с помощью предложенной модели. На рис. 7 показаны несколько сгенерированных изображений и соответствующие базовые элементы.

В п. 4 модель модернизируется так, чтобы максимально соответствовать предъявляемым требованиям. Дело в том, что необходимость вычисления случайных значений делает невозможной локальную генерацию, что является серьезным ограничением.

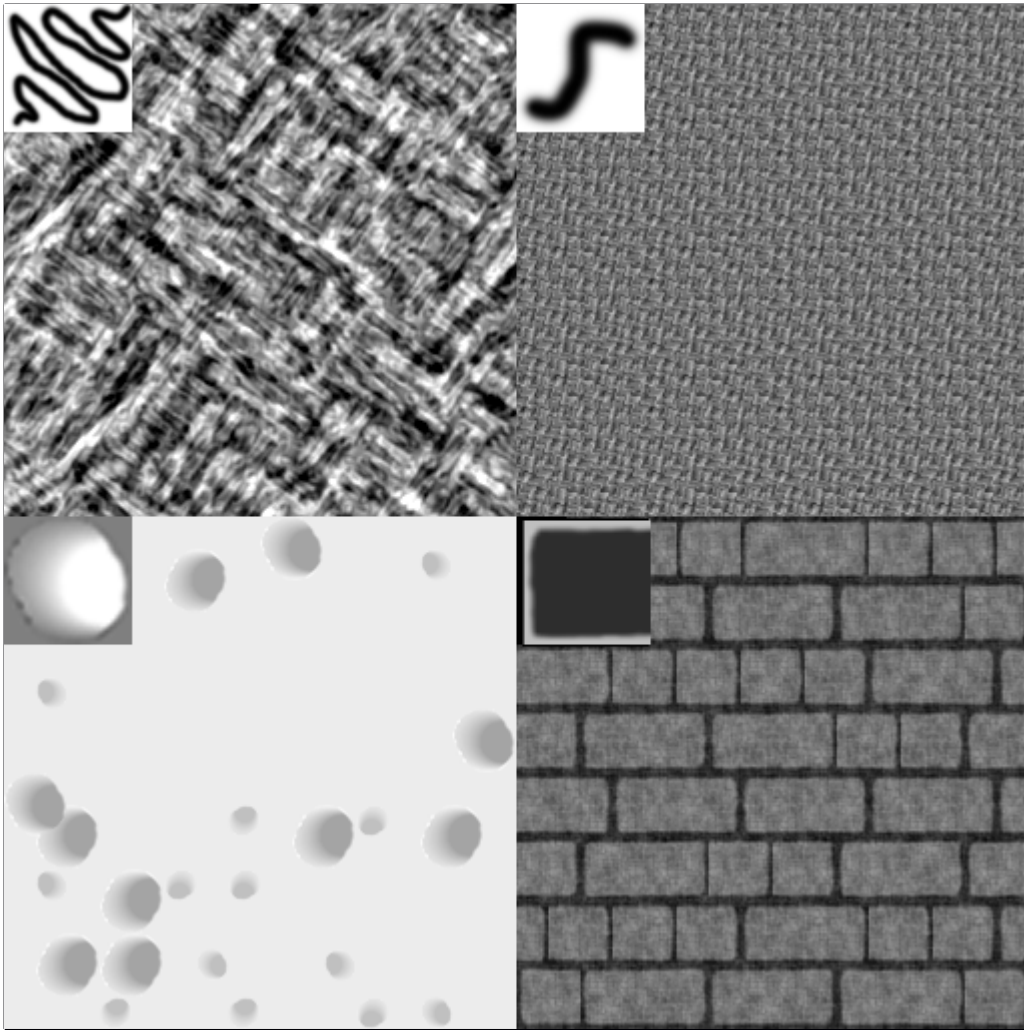


Рис. 7: Различные текстуры с базовыми элементами.

Предлагается с помощью имеющихся таблиц событий вычислять события в каждой точке репликации до начала фазы генерации и сохранять коды событий в специальных массивах — *управляющих масках слоев* (УМС).

Может показаться, что такой подход сведет на нет все попытки создать действительно компактное представление неперIODической текстуры. Появляются структуры данных конечного размера, в которых указаны уже не вероятности, а коды определенных событий. В этом случае приходится либо мириться с неизбежным периодическим эффектом, либо генерировать структуры большого размера, что соответствует большим периодам текстуры (а это приведет к увеличению объема памяти для хранения представления). Однако в работе показано, что возможен компромисс: в силу того, что предлагаемое представление оказывается достаточно компактным, хранение текстур с достаточно большими периодами требует относительно

малых объемов памяти.

Вместе с тем, применение УМС предоставляет следующие преимущества:

- Возможна локальная генерация текстуры по УМС.
- Генерация текстуры по УМС выделяется в отдельную фазу. Алгоритм генерации крайне прост, и допускает аппаратную реализацию. В то же время процесс описания и расчета самих УМС можно развивать и усложнять, поскольку его производительность уже не является существенным требованием.

В п. 5 рассматриваются вопросы представления данных и особенности реализации алгоритма генерации. На примере показывается, что предложенное представление может быть в 100 и более раз компактнее, чем представление сгенерированного изображения в виде битовой матрицы (алгоритмы сжатия текстур обеспечивают сжатие битовой матрицы не более чем в 10 раз).

П. 6 посвящен связи предложенной модели с теорией вейвлет-анализа. Описанная выше структура репликаций основана на простом двумерном обратном ВП непрерывных сигналов (которое рассматривается в приложении к диссертации). Запишем ее с помощью следующей формулы:

$$f(x, y) = v + \sum_{i=0}^{N-1} w^{(i)} \left(\sum_{j,k=-\infty}^{+\infty} W \left[\psi(2^i x - j, 2^i y - k), c_{j,k}^{(i)} \right] \right). \quad (5)$$

Начальным приближением является константа v — интенсивность фона. Аналогом вейвлета является базовый элемент $\psi(x, y)$. Аналогами вейвлет-коэффициентов выступают два объекта. Во-первых, это весовые коэффициенты $w^{(i)}$, определяемые по одному для каждого слоя. Во-вторых, коэффициент при каждой копии базового элемента меняется на функционал $W[\bullet, c]$, который преобразует аргумент в зависимости от значения числового параметра c . Параметр c — код ячейки УМС или значение случайной величины, распределение которой задается таблицей событий.

В п. 7 подводится итог проделанной работы. Рассмотрена модель для представления стохастических текстур, ориентированная на использование в графических устройствах на аппаратном уровне. Модель основана на обобщении разложения сигнала по вейвлет-базису. Указанное представле-

ние компактно, масштабируемо, допускает локальное декодирование. Разработан быстрый алгоритм декодирования такого представления и нанесения текстуры, допускающий аппаратную реализацию.

В конце главы рассматриваются возможные пути развития модели, которые позволят расширить класс представимых текстур.

В заключении формулируются основные результаты работы.

Приложение содержит справочную информацию по основам теории вейвлет-анализа — многомасштабному анализу и вейвлет-преобразованиям непрерывных сигналов.

Основные результаты работы

1. Разработан и реализован метод построения адаптивных треугольных сеточных представлений объектов, параметризуемых на плоскости. Метод основан на древовидной структуре многомасштабного анализа информации. С помощью предложенного метода решена задача расчета и представления функций освещенности.
2. Разработан и реализован метод построения линий уровня (изолиний) на основе B-сплайнового вейвлет-преобразования. Метод обеспечивает сглаживание кривых, построенных по зашумленным данным, и адаптацию к параметрам области отображения.
3. Предложена модель для представления масштабируемых стохастических текстур, основанная на обобщении разложения сигнала по вейвлет-базису. Разработан быстрый алгоритм декодирования такого представления и нанесения текстуры, допускающий аппаратную реализацию.
4. Применено на практике свойство многофункциональности многомасштабных представлений: одно и то же представление объекта используется для решения целого ряда подзадач (сглаживание, масштабирование, сжатие информации и пр.), что упрощает общую процедуру обработки и повышает эффективность ее реализации.
5. На примере разработанных методов показано, что разнообразие форм вейвлет-преобразований позволяет, в зависимости от структуры объекта и требований к приложению, делать выбор между простыми, про-

изводительными алгоритмами и более сложными методами, которые обеспечивают лучшие результаты обработки, но менее эффективны в реализации.

Публикации по теме диссертации

1. Переберин А.В. О систематизации вейвлет-преобразований // Вычислительные методы и программирование. — 2001. — Т. 2, № 2. — С. 133-158. (Электронная версия на сайте <http://num-meth.srcc.msu.su/>)
2. Переберин А.В. Построение изолиний с автоматическим масштабированием // Вычислительные методы и программирование. — 2001. — Т. 2, № 1. — С. 118-128.
(Электронная версия на сайте <http://num-meth.srcc.msu.su/>)
3. Pereberin A.V. From Photon Map to Irradiance Function via Wavelet Transform // GraphiCon'97 Proceedings. — 1997. — P. 38-43.
4. Pereberin A.V. Hierarchical Approach for Texture Compression // GraphiCon'99 Proceedings. — 1999. — P. 195-199.
5. Pereberin A.V. Fast Multi-Scaled Texture Generation and Rendering // GraphiCon'2000 Proceedings. — 2000. — P. 145-150.