

О ПОВЕРХНОСТЯХ СВЕРТКИ В МАШИННОЙ ГРАФИКЕ

Андрей Шерстюк

SquareUSA, Inc
andrei@squareusa.com

Поверхности свертки – это интегральное представление неявно заданных поверхностей, известных в машинной графике под следующими названиями: «капельные модели», «мета-сферы», «мягкие объекты». Поверхности свертки сочетают в себе гибкость капельных моделей с компактностью скелетных методов и представляют собой гораздо более мощное средство геометрического моделирования, чем традиционные модели неявных поверхностей. Тем не менее, практическое применение поверхностей свертки до сих пор ограничено узким кругом задач, связанных с моделированием объектов, содержащих плавные сочленения. Мы предлагаем более широкий взгляд на возможности геометрического моделирования поверхностями свертки.

1 Введение

1.1 Неявно заданные поверхности

История использования неявно заданных поверхностей в машинной графике ведет начало с пионерских работ Блинна [1], Нишимуры-сан и коллег [5], братьев Уайвилл [10] и насчитывает уже почти два десятка лет. Неявно заданные поверхности вошли в машинную графику как капельные модели (*blobby model*), мета-сферы (*metaballs*), мягкие объекты (*soft objects*). Несмотря на разнообразие названий, все эти модели описывают фактически один и тот же объект, а именно – изо-поверхность S уровня T в некоем скалярном поле $F(p)$:

$$S = \{p \in R^3 \mid F(p) - T = 0\} \quad (1)$$

где поле $F(p)$ задается в виде конечной суммы

$$F(p) = \sum_{i=0}^N f_i(p_i - p) \quad (2)$$

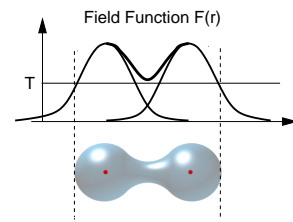


Рис. 1. Блоб: изо-поверхность в поле двух точечных компонент.

Каждая компонента $f_i(p)$ описывает поле точечного потенциала и обычно представлена какой-нибудь гладкой функцией, быстро затухающей с расстоянием от центра p_i . Поверхности, представленные уравнениями (1) и (2), иногда фигурируют под общим названием *классическая неявно заданная поверхность* или просто *блоб*. Типичный блоб показан на Рис.1.

1.2 Поверхности свертки

Поверхности свертки также являются изо-поверхностями, с той разницей, что скалярное поле $F(p)$ задается с помощью различных геометрических объектов. Если капельные модели (они же бобы), построены из точечных компонент-капель (Рис.1), то поверхности свертки базируются на более разнообразном выборе: точки, отрезки, дуги, треугольники и, в принципе, любые геометрические примитивы. Скалярное поле в таком случае вычисляется с помощью интегрального оператора свертки двух функций $g(p) * h(p)$ (отсюда и название класса поверхностей):

$$F(p) = g(p) * h(p) = \int_{R^3} g(p - r) h(r) dr \quad (3)$$

Функция $g(p)$ описывает геометрию примитива: точка, отрезок, дуга и так далее. Функция

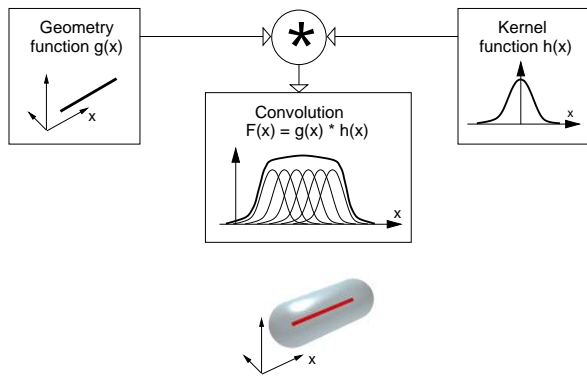


Рис. 2. Свертка с отрезком: слева – геометрическая функция $g(x)$, справа – ядро $h(x)$, в центре – функция свертки $f(x)$, внизу – изо-поверхность $f(x)$.

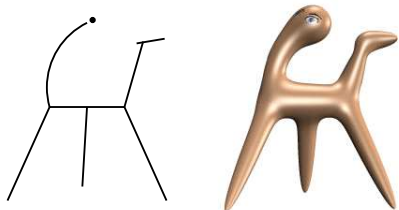


Рис. 3. Свертка с различными геометрическими примитивами: точка, отрезок, дуга. Прототипом этой модели является известная скульптура Деборы Халперн "Ангел" (Национальная Картинная Галерея, г.Мельбурн, Австралия).

$h(r)$ называется ядром свертки; она задает поле от точечного примитива. Пример свертки с отрезком показан на Рис.2; более сложная поверхность изображена на Рис.3.

Поверхности свертки были впервые применены в машинной графике Блюменталем и Шумейком в 1991 году [3].

1.3 Примеры использования неявно заданных поверхностей. Постановка задачи

Практические приемы и методы работы с неявными поверхностями во многом определяют ся их способностью к слиянию (Рис.1). В результате, большинство объектов, заданных

с помощью неявных поверхностей, содержат плавные сочленения (растения, тела людей и животных), либо демонстрируют некоторую мягкость (лица, губы, электронные облака в молекулах) или текучесть (кипящий суп). Все эти примеры и многие другие приведены в книге Блюменталя «Введение в неявные поверхности» [2].

Как легко видеть из Рис.2 и 3, поверхности свертки объединяют в себе гибкость капельных моделей с возможностью использования примитивов различных форм и размеров. Если представить процесс моделирования как описание объекта в некоем геометрическом языке, то словарный запас большинства популярных моделей неявных поверхностей состоит из одного слова: точка. В языке поверхностей свертки набор слов-примитивов ограничен только воображением дизайнера и, конечно, наличием программной реализации.

Несмотря на свои качественно более широкие возможности и солидный почти десятилетний возраст, поверхности свертки до сих пор не нашли широкого применения в машинной графике. Немногие примеры их использования по-прежнему не выходят за рамки традиционных задач неявных поверхностей, а именно: моделирование объектов, содержащих плавные сочленения. Примеры: человеческие ладони, деревья (также в книге [2]).

Цель этой работы – показать, что поверхности свертки способны на гораздо большее. Морщины, складки, шипы, зубцы, шероховатости – все эти атрибуты, столь типичные для органических объектов и столь нетипичные для классических неявных поверхностей, будут рассмотрены в качестве тестовых задач. Мы перечислим и коротко опишем на конкретных примерах практические приемы работы с поверхностями свертки в системе геометрического моделирования RATS.

2 Математическое представление: проблемы и решения

В определении поверхности свертки (уравнения 1 и 3) нет явных ограничений на геометрическую функцию $g(p)$, и, в принципе, любой геометрический объект может быть использован как источник скалярного поля. На практике, выбор таких объектов жестко ограничен соотношениями эффективности вычислений поля (3).

Главная трудность этих вычислений сос-

тоит в том, что для большинства примитивов $g(p)$ и ядер свертки $h(p)$ интеграл (3) не берется аналитически, т.е. не выражается в элементарных функциях. Такое неудобство представляет собой самое серьезное препятствие для практического использования поверхностей свертки в машинной графике. Приведем простой пример. Чтобы просчитать изо-поверхность для изображения размера M на N пикселей, нужно $M \times N$ раз решить уравнение $F(p) = T$. Для изображения размера телевизионного порядка, скажем, 640 на 480, это число составляет более 300 тысяч. Если при этом использовать итерационные методы, число вычислений интеграла поля $F(p)$ может легко достичь десятков миллионов. При визуализации поверхности в режиме повышенного качества эти десятки миллионов превращаются в сотни. Нетрудно представить, насколько важно иметь возможность быстро получать значение функции поля $F(p)$.

До последнего времени только точечные и линейные примитивы позволяли найти аналитическое решение интеграла (3). Более сложные примитивы не поддавались интегрированию, что неизбежно влекло использование косвенных методов вычисления поля. Как правило, такие методы опирались на двух- и трех-мерные сетки заранее просчитанных значений (3), полученных численным интегрированием, с последующей интерполяцией между узлов. Естественно, что точность оценки интеграла (3) при этом прямо зависит от размеров сетки, отсюда – высокие требования на память и время инициализации сеток. Более серьезный недостаток интерполяционных методов заключается в их чувствительности к взаимному расположению объекта и наблюдателя. Например, если некая сетка дает хорошее гладкое изображение поверхности при наблюдении издалека, то вблизи объекта шаг сетки может оказаться (и скорее всего окажется) слишком большим. В результате, непрерывная гладкая поверхность свертки получится в лучшем случае фасетчатой, и в худшем случае – дырявой.

Таким образом очевидно, что для эффективного и правильного построения поверхности необходимо уметь быстро и точно считать интеграл поля (3). Для этого мы использовали наши предыдущие результаты [6], где решения интеграла свертки выведены аналитически для ряда геометрических примитивов. Аналитическое представление позволяет быстро считать изо-поверхности $F(p) = T$

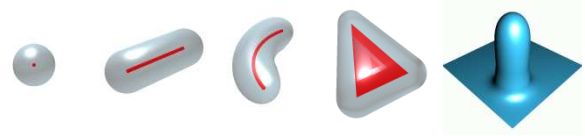


Рис. 4. Изо-поверхности неявных примитивов: точка, отрезок, дуга, треугольник и плоскость (в паре с отрезком).

с точностью машинного представления вещественных чисел во всех областях пространства модели.

3 Система моделирования

Мы перечислим и кратко опишем основные понятия и методы, использованные в нашей системе геометрического моделирования.

3.1 Неявные примитивы

Мы использовали пять различных скалярных функций $f(p)$, полученных при помощи оператора свертки со следующими геометрическими примитивами: точка, отрезок, дуга, треугольник и плоскость. По аналогии с неявно заданными поверхностями, мы называем такие скалярные функции неявными примитивами. Каждая из этих функций при подстановке в уравнение (1) дает свою собственную изо-поверхность (Рис.4). Подробнее об этих функциях можно прочитать в [6].

3.2 Скелетоны

Скелетон – это совокупность примитивов, которые вместе определяют общие очертания объекта. В терминах поверхностей свертки, скелетон задается суммой геометрических функций $g(p)$:

$$G(p) = \sum_{i=1}^N g_i(p) \quad (4)$$

Напомним, что суммарное поле $F(p)$ от всего объекта считается так:

$$F(p) = h * G = h * \sum_{i=1}^N g_i(p) \quad (5)$$

Поскольку оператор свертки обладает свойством линейности,

$$F(p) = \sum_{i=1}^N g_i(p) * h = \sum_{i=1}^N f_i(p) \quad (6)$$

что позволяет считать поле от всего скелетона по частям, а также строить сложные примитивы $F(p)$ из простых компонент $f_i(p)$. Например, из дуг легко можно составлять кольца и спирали, из треугольников – многоугольники и так далее. Это простой результат очень важен для практических целей.

Визуально скелетоны представляются геометрическим объединением примитивов, как показано на Рис.3 слева.

3.3 Материалы

Как следует из уравнений (1, 3, 6), поверхность свертки задается скелетоном $g(p)$, ядром $h(p)$ и скалярным уровнем T . Скелетон при этом задает глобальную форму объекта. Соответственно, локальные свойства поверхности должны определяться параметрами ядра $h(p)$ и значением T .

В своей исторической работе Блинн [1] использовал функцию поля $h(r) = b \exp(-a^2 r^2)$. Из высоты b и ширины a этой колоколообразной функции, а также значения T , Блинн вывел два параметра, которые он назвал *собственным радиусом* R и *липучестью* (*blobbiness*) B . Первый параметр задает размер сферы – изо-поверхности от отдельно взятого точечного примитива (см. Рис.4, крайний слева). Второй параметр определяет насколько охотно примитивы слипаются вместе, когда оказываются рядом. Внешне это проявляется в толщине перемычки между примитивами на Рис.1.

Параметры R и B , введенные Блинном для его точечной капельной модели (*blobby model*), можно легко перенести на более сложные примитивы, что мы и сделали для отрезков, дуг, треугольников и плоскостей. На Рис.5 показана серия Т-образных поверхностей с одним и тем же скелетоном и разными значениями R и B . (Заметим в скобках, что внешний вид букв «Т» в правом столбце на Рис.5 подсказывает, быть может, более удачный перевод слова *blobbiness* как «надутость». Однако, именно «липучесть» отдельных элементов поверхности вызывает их слияние, а надутость – лишь внешнее проявление, видимый результат.)

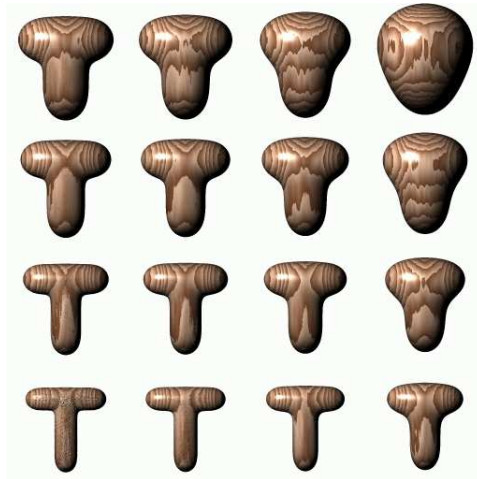


Рис. 5. Т-образная поверхность свертки. Увеличивание липучести (слева направо) и собственного радиуса (снизу вверх) позволяет варьировать форму поверхности.

Мы нашли удобным включить собственный радиус и липучесть в набор параметров, задающих оптические свойства поверхности, такие как цвет, коэффициенты преломления, отражения и так далее. Таким образом, мы расширяем понятие *материала*, который помимо чисто фотометрических характеристик (цвет, отражательная способность и т. п.), теперь еще несет и геометрическую информацию об объекте. Продолжая анатомическую аналогию со «скелетами», каждая «кость» (она же неявный примитив) оказывается покрытой своеобразной мягкой тканью с толщиной R и липучестью B , которая во многом напоминает эластичность.

3.4 Профилирование

На Рис.4 все неявные примитивы показаны в своей канонической, изначальной форме, которая легко может быть изменена для каждого конкретного случая. Существует бесконечное число способов исказить форму примитива; Рис.6 демонстрирует самый простой из них – умножение значения поля $f(x, y, z)$ на различные профилирующие функции от аргумента x . Поскольку эти функции меняют значение поля, изо-поверхности вынуждены следовать этим изменениям. Профилирующие функции могут быть заданы от одного, двух и большего числа переменных, включая время в анимации.

Как показывает практика, профилирование

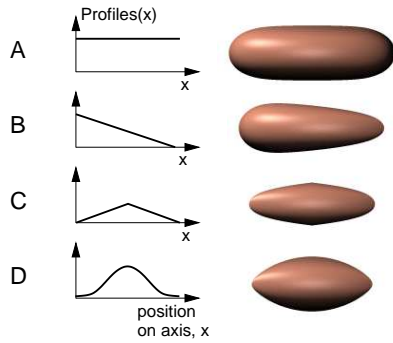


Рис. 6. Профилирование линейного примитива.

является чрезвычайно полезным расширением стандартной модели поверхностей свертки. Профилирование в той или иной форме использовано во всех примерах, которые будут рассмотрены ниже.

3.5 Офсетные поверхности

Как видно на Рис.4, отдельно взятые неявные примитивы генерируют изо-поверхности, очень похожие на свои офсетные поверхности. А именно, точка превращается в сферу, отрезок – в цилиндр, закрытый двумя полусферами, дуга – в часть тора и так далее. Такое сходство подсказывает естественный способ ускорить процесс моделирования, заменив настоящие неявные примитивы их двойниками – офсетными поверхностями. Полезность такой замены заключается в том, что офсетные поверхности легко поддаются визуализации, например, с помощью алгоритма трассировки лучей. Это позволяет легко манипулировать моделью, добавлять и убирать компоненты, менять их расположение интерактивно; при этом каждая модификация тут же отображается на экране. Иначе говоря, офсетные поверхности – это удобные черновики, позволяющие быстро построить модель в первом приближении. Когда черновик готов, пользователь заменяет офсетные поверхности настоящими неявными примитивами. В таком режиме, все компоненты модели (бывшие сферы, цилиндры и торы) рассматриваются как источники поля (3) в уравнении изо-поверхности (1). Офсетная поверхность (объединение офсетных примитивов) таким образом превращается в поверхность свертки. При этом начинают работать такие эффекты, как слияние компонент, профилирование – начинается тонкая настройка модели.

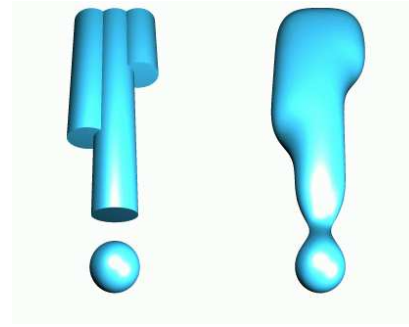


Рис. 7. Двойственное представление модели сосульки: слева – офсетная поверхность, справа – поверхность свертки. Обе поверхности заданы одним и тем же набором команд создания примитивов (одна точка и три отрезка).

В нашей системе моделирования и визуализации *RATS*, мы поддержали такую двойственность программно, используя расширенный синтаксис для всех неявных примитивов. Приведем пример. Командная строка

```
sphere Ice, r1, 0 0 – 0.85
```

создает примитив, который может быть использован как обыкновенная сфера из некоего заранее заданного материала *Ice*, с радиусом r_1 и центром в точке $(0, 0, -0.85)$. Если материал *Ice* содержит в списке своих параметров липучесть, то такая сфера в любой момент принять «неявный» вид, т.е. превратиться в точечную компоненту поверхности свертки. Для программы визуализации такое переключение задается простым флагом типа *да/нет*. Благодаря двойственности представления всех примитивов, перезагрузки модели в память при этом не требуется, что особенно удобно для быстрого переключения между явным и неявным режимами визуализации. Упомянутая сфераточка взята «живьем» из модели сосульки на Рис.7.

3.6 Переменные параметры

Для большего удобства моделирования мы использовали переменные следующих типов: скаляр, вектор, цвет, материал и текстура. В процессе работы с моделью переменные можно создавать и модифицировать на ходу. Переменные можно также присваивать и выражать друг через друга, соблюдая совместимость типов.

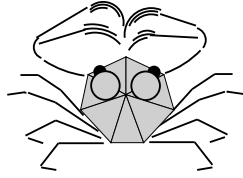


Рис. 8. Использование примитивов разных типов позволяет создавать компактные и выразительные скелетоны. В этой модели ручной работы использованы точки, отрезки, дуги и треугольники.

Благодаря использованию переменных, практически все параметры модели можно настроить интерактивно, т.е. без перезагрузки модели в память. Сюда входят: геометрическое расположение скелетных элементов, их толщина и липучесть, все фотометрические характеристики материалов, текстуры. Параметры камеры и источников света также можно задавать и настраивать с помощью переменных.

4 Практические примеры

Ниже мы приведем несколько практических примеров геометрического моделирования поверхностями свертки. Эти примеры сгруппированы вокруг методов, которые были при этом использованы: глобальное задание формы объекта (скелетоны), локальные изменения (профилирование) и мелкие детали.

4.1 Скелетоны

Все скелетоны, применимые для поверхностей свертки, можно примерно разделить на четыре вида.

Процедурные скелетоны задаются алгоритмически и часто используются при моделировании само-повторяющихся структур: морские раковины, кораллы, деревья и другие «растущие» объекты. Примеры: Рис.9 и 11. Эти модели основаны на утилитах Эрика Хайнса [4].

Скелетоны ручной работы. Пример такой модели приведен на Рис.12. Морской конек был нарисован от руки на бумаге и продублирован в графическом редакторе набором дуг (слева). Дуги затем были превращены в офсетную поверхность (центр), а те, в свою очередь, – в

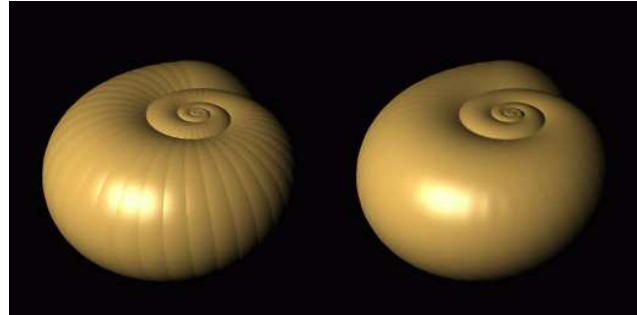


Рис. 9. Простая спиральная модель морской раковины: офсетная поверхность (слева), неявная поверхность (справа). Число элементов: 361.

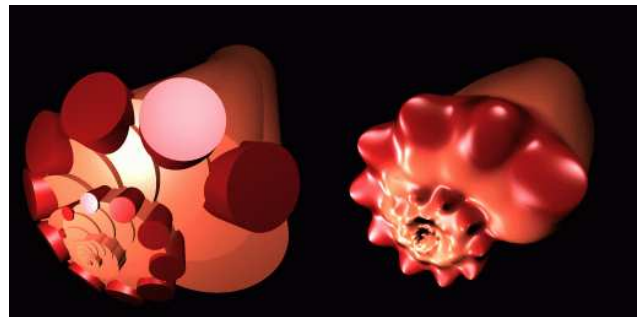


Рис. 10. Рогатая раковина, полученная наложением двух процедурных скелетонов. Число элементов: 70.

поверхность свертки (справа). Еще один пример скелетона такого вида приведен на Рис.8 и 13.

Скелетоны смешанного происхождения сочетают в себе процедурные и самодельные элементы. Скелетон рогатой раковины на Рис.10 получен на основе процедурной модели круглой раковины на Рис.9 путем замены точечных элементов линейными отрезками. Для этого достаточно было добавить одну общую вершину, расположенную над плоскостью спирали, и соединить ее с центрами всех сфер. Подобным образом был получен второй набор отрезков («рога»); оба набора затем были проряжены и слиты в единую изо-поверхность.

Импортированные скелетоны. Геометрическая модель бюста Нефертити (сетка треугольников на Рис.14, слева) была использована в качестве скелетона поверхности свертки без каких-либо изменений.

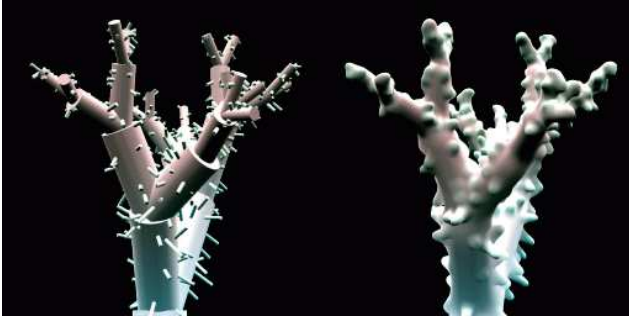


Рис. 11. Коралловое дерево. Ствол и ветви выращены процедурно, мелкие детали добавлены позже. Число элементов: 548.

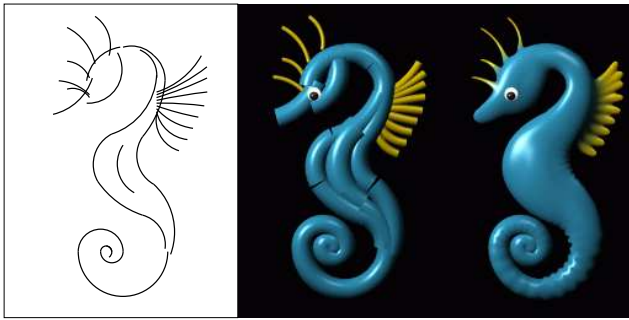


Рис. 12. Морской конек. Поверхность свертки собрана из 43 дуг, заданных вручную. Обратите внимание, как складки на хвосте и спине говорят о мягкости кожи.

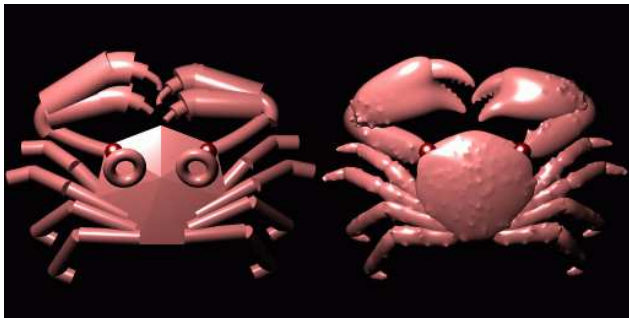


Рис. 13. Краб. Туловище, конечности и клешни смоделированы вручную (слева). Многочисленные неровности на всем панцире, а также зубцы на клешнях, добавлены с помощью утилит специального назначения. Число элементов: 641

4.2 Профилирование

Когда первый черновик модели готов (что соответствует офсетным поверхностям на левой стороне Рис.9-13), можно применять более тонкие методы работы с моделью. Как мы уже упоминали, одним из самых действенных приемов «обработки поверхности» является профилирование, во многом сходное с вырезанием формы из заготовки (см. Рис.6). Для аксиально симметричных заготовок-примитивов (например, цилиндров), профилирующие функции определяют поверхность вращения.

Все элементы рогатой раковины на Рис.10 были изменены функцией $\sin(x)$ (см. Рис.6, внизу) – иначе фронт раковины сливался в одну аморфную массу. Подобным же образом были заострены зубцы клешней и конечные сегменты ног краба на Рис.13.

4.3 О деталях

Реалистичное представление органических объектов требует очень высокой степени детализации их поверхностей. Кроме того, чем больше деталей на поверхности, тем она интереснее для зрителя. Все типы деталей, использованные в системе RATS, можно примерно разделить на три группы: структурные, функциональные и процедурные.

Структурные детали создаются тривиальным добавлением большого числа мелких элементов. Добавочные элементы можно задавать вручную или с помощью специальных утилит. Примеры: коралловое дерево (Рис.11), краб (Рис.13). Этот метод «грубой силы» может увеличить размер модели в десятки и сотни раз. Достоинства – простота реализации.

Функциональные детали – это профилирование отдельных примитивов высокочастотными функциями. Пример – складки на хвосте и спине морского конька заданы с помощью функции $t(x) = 1 + a \sin(bx)$, где x – положение на дуге хвоста и спины в радианах, параметры a и b задают глубину и частоту складок. Помимо элементарных профилирующих функций, можно использовать более интересные профили, например, двух- и трехмерный «шум», что может придать поверхности некоторую шероховатость. Заметим, что профилирующие функции лишь намечают направление изменений в уровне изо-поверхности и конечный результат во многом зависит от влияния соседних компонент

модели. Если поле примитива (в данном случае, дуги спины) невелико по отношению к вкладу соседей, то даже очень глубокие ($a = 1$) и даже отрицательные ($a > 1$) профили могут остаться незамеченными.

Процедурные детали достигаются программными средствами. Пример – морщины на коже Нефертити (Рис.14). Эта модель являет собой довольно необычный случай неявной поверхности и заслуживает отдельного пояснения.

Сначала исходная сетка треугольников модели Нефертити (Рис.14, слева) была превращена в поверхность свертки (Рис.14, центр), что вызвало значительные изменения в ее внешности. Эти изменения, впрочем, вполне нормальны: ядро свертки работает как фильтр низкой частоты и сглаживает все тонкие детали. При этом пострадали самые важные участки поверхности, сделав лицо Нефертити практически неузнаваемым. Чтобы уменьшить излишнюю размытость, поле от каждого треугольника было искусственно обрезано на некотором расстоянии от треугольника и плавно интерполировано эрмитным сплайном внутри. В результате такой операции, поле во внутренних областях каждого треугольника осталось практически неизменным, поле на краях было существенно уменьшено. В итоге, поверхность свертки оказалась прижатой к сторонам треугольников и приподнятым над их центральными областями. Внешний эффект – морщинистая поверхность на Рис.14, справа. Варьируя глубину морщин и толщину кожи (параметр R в описании материала, см. Рис.5), можно легко менять видимый возраст объекта.

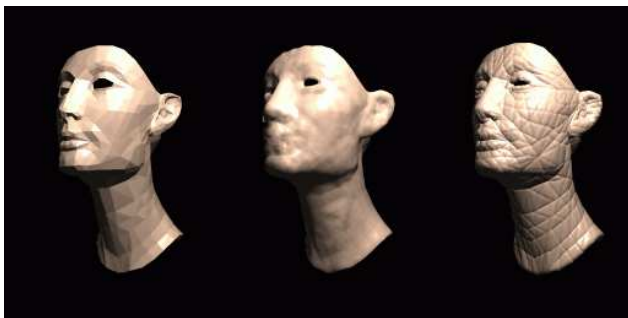


Рис. 14. Нефертити, до (слева) и после свертки (центр, справа). Число элементов: 1242

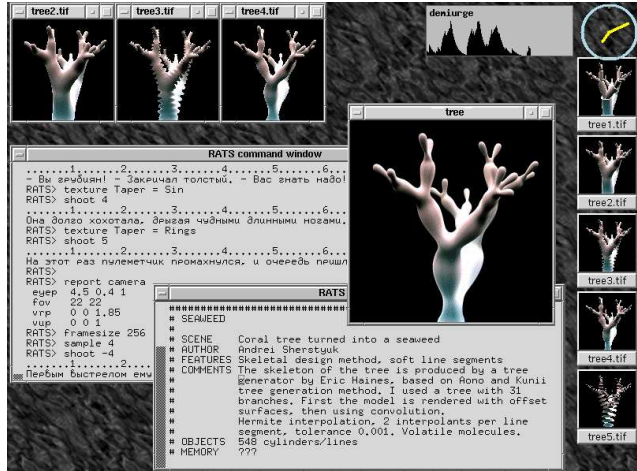


Рис. 15. Система RATS, снимок с экрана: окно ввода команд (слева), командный файл (внизу), набор пробных изображений (верхний ряд).

Модель	t_1	t_2	t_1/t_2
Круглая раковина	2:52	22:46	7.81
Рогатая раковина	12:54	41:16	3.20
Морской конек	5:25	11:52	2.19
Краб	14:03	46:13	3.29

Таблица 1. Время визуализации (мин:сек). t_1 – офсетные поверхности, t_2 – поверхности свертки.

5 Реализация

Все модели, описанные выше, были созданы в среде геометрического моделирования и визуализации RATS, написанной автором. Подробнее о RATS можно прочитать в Интернете [7]. Алгоритм трассировки лучей для неявных поверхностей описан в [8]. Офсетные поверхности также трассированы. В Таблице 1 приведены времена визуализации некоторых моделей. Размер изображений 512 на 512 пикселей, процессор Pentium 90, операционная система Linux. Все изображения созданы в режиме повышенного качества, максимальное число первичных лучей 16.

6 Заключение

Перефразируя известное высказывание, если единственным инструментом моделирования

является точечный примитив-капля, то дизайнер вынужден смотреть на мир как на один огромный блок.

Многообразие форм и размеров примитивов свертки устраняет такое неудобство и дает возможность применять неявные поверхности для решения широкого класса задач машинной графики. Реалистичное моделирование объектов органического происхождения – одна из них.

В заключении автор хочет сказать большое спасибо А.А. Пасько и В.В. Савченко за прекрасную организацию конференции «Shape Modeling International '99» [9] в университете Айзу-Вакаматсу, Япония, где эта работа была представлена впервые. Хочется надеяться, что события такого уровня полезности и приятности будут случаться и впредь. Также большое спасибо В.Д. Аджиеву за приглашение этой статьи на страницы «Открытых систем».

Источники

- [1] J.Blinn, "A Generalization of Algebraic Surface Drawing", ACM Transactions on Graphics, 1(3), 1982.
- [2] Editor J.Bloomenthal, "Introduction to Implicit Surfaces", Morgan Kaufmann Inc, 1997.
- [3] J.Bloomenthal and K.Shoemake, "Convolution surfaces", SIGGRAPH Proceedings, 1991.
- [4] E.Haines, Standard Procedural Databases (SPD), <ftp.princeton.edu:/pub/Graphics>
- [5] H.Nishimura, M.Hirai, T.Kawai, T.Kawata, I.Shirakawa and K.Omura, "Object Modelling by Distribution Function and a Method of Image Generation", The Transactions of the Institute of Electronics and Communication Engineers of Japan", J68-D, 4, 1985.
- [6] J.McCormack and A.Sherstyuk, "Creating and rendering convolution surfaces" Computer Graphics Forum, 17(2) 1998.
- [7] RATS site:
<http://www.cs.monash.edu.au/~ash/rats/>
- [8] A. Sherstyuk, "Fast Ray Tracing Of Implicit Surfaces", Computer Graphics Forum, to appear, 1999.

- [9] "Shape Modeling International'99"
<http://www.u-aizu.ac.jp/public/www/labs/sw-sm/SMI99/>
- [10] G.Wyvill, C.McPheeters and B.Wyvill, "Data structure for soft objects", The Visual Computer, 2(4), 1986.