

# ПОВЫШЕНИЕ КАЧЕСТВА И ПРОИЗВОДИТЕЛЬНОСТИ ТРЕХМЕРНОЙ ВИЗУАЛИЗАЦИИ ТОМОГРАММ

*Н.И. Гаврилов, В.Е. Турлапов*

*Нижегородский госуниверситет им. Н.И. Лобачевского*

## **Введение**

В научной визуализации часто приходится иметь дело со скалярными полями данных в трехмерном пространстве. Данные могут быть получены различными способами: численным экспериментом, сканированием с помощью магнитного резонанса, компьютерной томографией (КТ), сканированием с помощью ультразвука. Например, результатом КТ является множество слоев, т.е. двумерных массивов данных, которые вместе образуют трехмерное скалярное поле. Объем данных эксперимента значителен. Например, объем одного исследования в медицинской томографии, обычно равен 0.25-1 Гб. Обеспечение реального времени визуализации означает вывод на экран порядка 25 и более кадров в секунду.

С 90-х годов прошлого века техника прямого объемного рендеринга (Direct Volume Rendering) показала себя, как эффективный инструмент визуального анализа пространственных данных [2]. Различные общепринятые подходы [4] позволяют добиться визуализации в реальном времени путем использования параллельных вычислений на графических процессорах (GPU). Успехи в вычислениях общего назначения на GPU (GPGPU) также позволяют добиться интерактивной визуализации нескольких массивов данных одновременно [1].

## **1. Мотивация к созданию собственной системы для визуализации**

В ходе совместной работы с Институтом прикладной физики (ИПФ РАН) возникла потребность в визуализации результатов численного моделирования взаимодействия лазерного импульса с плазмой. Данные представляют собой набор трехмерных матриц чисел, характеризующих различные поля. Объемный рендеринг может быть использован для визуализации одного массива данных, тогда как для совместной визуализации нескольких массивов необходим мульти-объемный рендеринг, который не реализован ни в одной на сегодня доступной системе визуализации.

Кроме того, разработка собственной системы (рис. 1) позволила добавить некоторые полезные функции, такие как выделение области интереса полигональной сеткой, подавление артефактов рендеринга, оптимизация рендеринга для повышения производительности и др. В последствии проект стал частью разрабатываемой телемедицинской системы и авторы стали ориентироваться на медицинскую визуализацию.

## **2. Методы рендеринга**

Для реализации объемного рендеринга был выбран метод бросания лучей (Ray Casting, RC) [1, 4, 5, 8], как решение наиболее гибкое, высокопроизводительное, и дающее наилучшее качество визуализации. Благодаря гибкости метода RC, возможно множество вариантов того, как визуализировать данные. Алгоритм вычисляет цвет каждого пикселя искомого изображения, генерируя (бросая) луч, который в ходе прохода через объем с данными вычисляет цвет для пикселя.

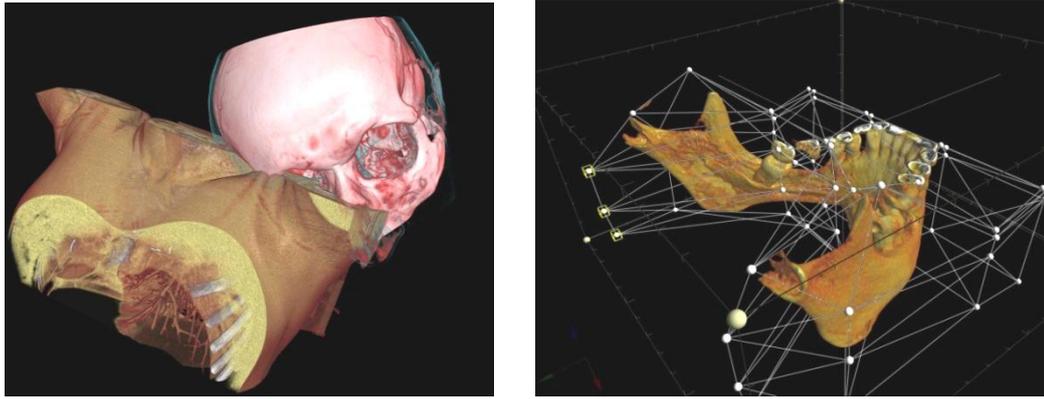


Рис.1. Возможности разработанной системы InVols (In the Volumes): мульти-объемный рендеринг (слева); сечение полигональной геометрией (справа)

В системе реализованы шесть различных техник визуализации, каждая из которых поддерживает режим мульти-объемности, т.е. во время движения луча происходит обращение к нескольким массивам данных. Местоположение массива данных определяется собственной системой координат, заданной матрицей преобразования. Кроме местоположения и поворота, матрица также определяет размеры вокселя (элемента трехмерной матрицы), поскольку воксель не всегда имеет форму куба (томограммы обычно представляют собой набор слоев, расстояние между которыми отлично от размера пикселя слоя).

Проекция максимальной интенсивности (Maximum Intensity Projection, MIP) широко используется в медицинской практике для визуализации сосудов [3]. Проекция минимальной интенсивности (MinIP) также может быть использована для исследования бронхов.

Метод MIP проектирует максимальные значения данных на экранную плоскость. Обычно для раскраски используют оттенки серого и алгоритм визуализации регулируется только окном (рис. 2) (в медицинской визуализации существует понятие окна, которое является парой чисел, задающих яркость и контрастность изображения).

Затененный прямой объемный рендеринг (sDVR) также используется в медицинской практике [2]. В отличие от MIP метода эта техника называется реалистичной, поскольку накопление цвета происходит с учетом накопленной лучом непрозрачности, т.е. видимые особенности данных могут быть заслонены другими особенностями. Поэтому данная техника позволяет оптимизацию путем раннего завершения луча (early ray termination).

Полупрозрачные изоповерхности обычно используются для научной визуализации. Качество визуализации именно изоповерхностей может быть заметно улучшено путем уточнения точки столкновения луча с поверхностью [5], тогда как в технике sDVR нет изоповерхностей, видимая среда непрерывна и полупрозрачна.

### 3. Повышение производительности рендеринга

#### 3.1. Раннее завершение луча

Данная стратегия типична для алгоритма бросания лучей. Благодаря гибкости архитектуры современных GPU возможен преждевременный выход из цикла движения луча (обычно при накоплении непрозрачности, близкой к единице). Однако, например, для техники MIP лучам необходимо двигаться на протяжении всего рассматриваемого отрезка внутри ограничивающего объема, т.к. максимум может находиться в любом месте отрезка.

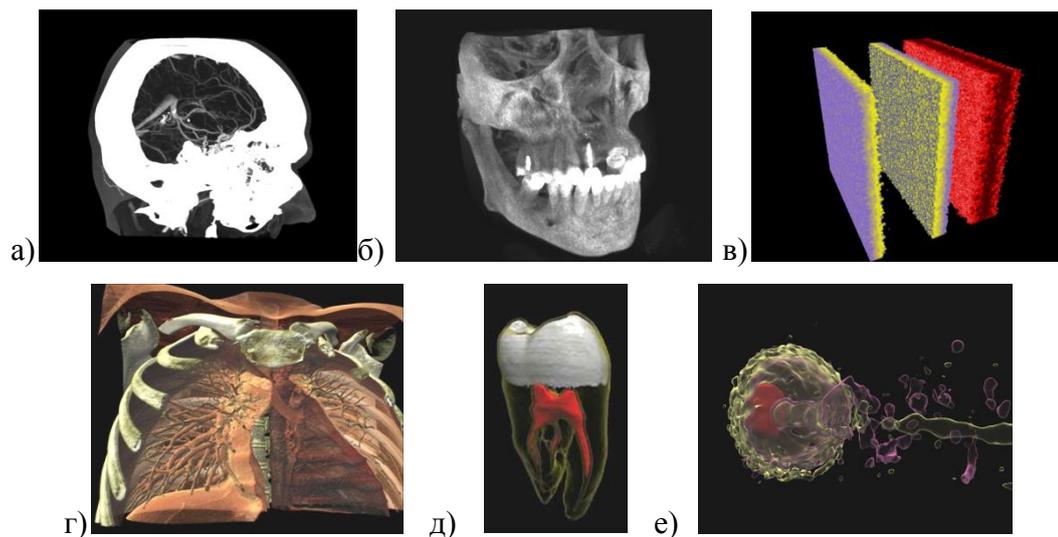


Рис. 2. Примеры разнообразных техник рендеринга: а), б) – MIP; в), г) – sDVR; д), е) – изоповерхности

### 3.2. Пропуск пустых областей (Empty space skipping) с использованием регулярной ускоряющей структуры

Особенно широко эта стратегия применяется для алгоритма бросания лучей, реализованного на центральном процессоре [8]. Однако некоторые техники могут быть применены и для случая реализации на GPU. Мы используем одноуровневую ускоряющую структуру: например, если визуализируем массив данных размером  $512 \times 512 \times 512$  вокселей, то используем дополнительный массив данных размером  $16 \times 16 \times 16$ . Каждая ячейка этого массива покрывает блок  $32 \times 32 \times 32$  исходных данных и содержит диапазон, в котором лежат значения данных из этого блока. Таким образом, построенная ускоряющая структура содержит информацию, достаточную для пропуска значительного числа блоков, которые заведомо прозрачны. Данный подход повышает производительность рендеринга в 1.5-2 раза.

### 3.3. Пропуск пустых областей при помощи сечения полигональной сеткой

Сечение данных замкнутой полигональной поверхностью может быть использовано как для повышения производительности [5], так и для сегментации, т.е. выделения области интереса. Причем в нашей реализации полигональная сетка не должна быть выпуклой, т.е. лучи могут иметь несколько отрезков, по которым они будут двигаться. Перед тем, как производить непосредственно бросание лучей, для каждого луча вычисляем список отрезков, по которым лучи будут двигаться.

Для этого мы сначала производим рендеринг лицевых граней нашей полигональной поверхности и записываем расстояния от наблюдателя до соответствующей пиксели точки поверхности в экранный буфер (рис. 3). Затем таким же образом записываем расстояние до задних граней. Таким образом, для каждого пикселя (т.е. луча) мы получили первый отрезок движения. Затем производим рендеринг тех точек лицевых граней поверхности, которые были заслонены при предыдущем рендеринге задними гранями – эти точки являются началами вторых отрезков движения лучей и расстояния до этих точек также записываются в очередной экранный буфер. Таким образом, количество отрезков (т.е. сегментов пути движения луча) ограничивается количеством используемых экранных буферов. В нашей реализации в качестве буферов используются две 4-канальные текстуры, т.е. для каждого луча доступно максимум 4 отрезка.

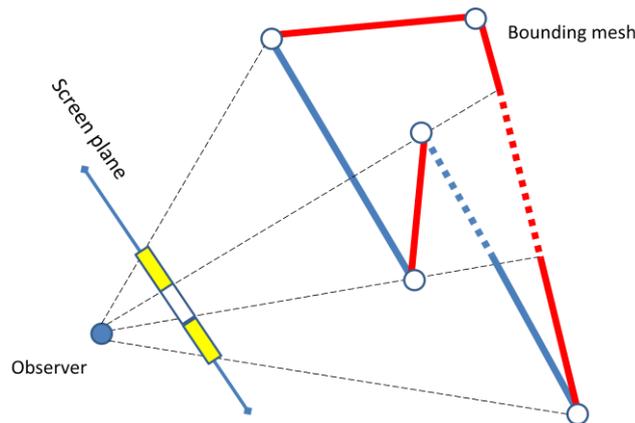


Рис 3. Иллюстрация рендеринга полигональной сетки в экранный буфер. Желтая область буфера показывает, где у лучей будет только один отрезок движения, белый – два отрезка

#### 4. Подавление артефактов рендеринга

Из-за конечного количества шагов движения луча, некоторыми из них могут быть пропущены важные видимые особенности, размеры которых, в силу использования интерполяции, могут быть много меньше размера вокселя. Причем из-за регулярности расположения стартовых позиций лучей артефакты также имеют регулярный вид и выглядят, как концентрические замкнутые линии на поверхностях. Путем сдвига стартовых позиций лучей на случайный вектор, сонаправленный их движениям, можно избавиться от регулярности артефактов. Дальнейшее накопление таких зашумленных изображений даст изображение без шума.

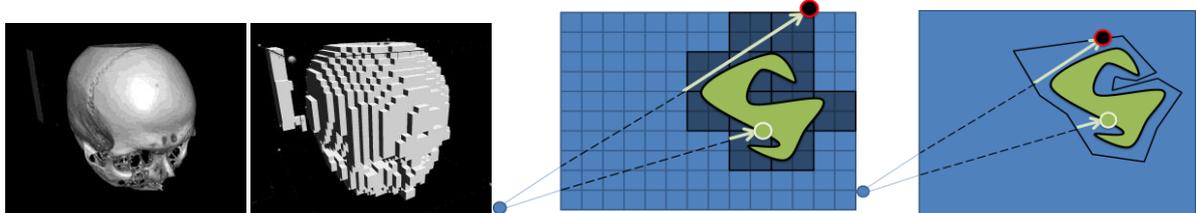


Рис 4. Слева направо: исходные данные; соответствующая ускоряющая структура; иллюстрация пропуска пустых областей при помощи структуры, записанной в GPU; пропуск областей путем отсечения полигональной поверхностью



Рис 5. Слева направо: артефакты из-за пропуска лучом особенностей в данных; результат накопления изображений; артефакты при три-линейной фильтрации; использование три-кубической фильтрации

Также качество визуализации может быть улучшено, если использовать три-кубическую интерполяцию при выборке данных вместо традиционной три-линейной. Мы расширили опубликованный алгоритм бикубической фильтрации [7] на трехмер-

ный случай. Учитывая возможность GPU эффективно производить три-линейную фильтрацию, три-кубическую выборку целесообразней составить из восьми три-линейных, чем производить 64 выборки из определенных ячеек данных.

## 5. Результаты



Рис 6. Тестовые данные, слева направо: Head, Dental, Feet

Таблица 1. Ускорение рендеринга методом пропуска пустых областей при помощи сечения полигональной поверхностью.

Данные	Размер	Три-линейная фильтрация	Три-кубическая фильтрация
Head	512 <sup>3</sup>	1.6	3
Dental	512x512x331	1.7	3.2
Feet	512x512x250	2	3.6

В табл. 1 представлено ускорение визуализации при использовании техники пропуска пустых областей при помощи сечения полигональной сеткой. В качестве сетки мы использовали поверхность ускоряющей структуры (рис. 4).

Работа выполнена при поддержке ФЦП «Научные и научно-педагогические кадры инновационной России», госконтракт № 02.740.11.0839.

## Литература

1. Kainz B. et al. Ray Casting of Multiple Volumetric Datasets with Polyhedral Boundaries on Manycore GPUs. Proceedings of ACM SIGGRAPH Asia 2009, Vol. 28, No 152.
2. Lundström C. Efficient Medical Volume Visualization: An Approach Based on Domain Knowledge. Linköping Studies in Science and Technology. Dissertations, 2007. 0345-7524; No. 1125.
3. Geoffrey D. Data explosion: the challenge of multidetector-row CT. In IEEE Transactions on European Journal of Radiology. 2000. Vol. 36, Issue 2, p. 74-80.
4. Klaus E. et al. Real-Time Volume Graphics, A.K. Peters, New York, USA. 2004.
5. Scharsach H. Advanced GPU raycasting. In Central European Seminar on Computer Graphics, 2005. p. 69-76.
6. Gordon L. Semi-automatic generation of transfer functions for direct volume rendering. A Thesis Presented to the Faculty of the Graduate School of Cornell University in Partial Fulfillment of the Requirements for the Degree of Master of Science. 1999.
7. Daniel R. et al. Efficient GPU-Based Texture Interpolation using Uniform B-Splines. In IEEE Transactions on Journal of Graphics, GPU, & Game Tools, 2008. Vol. 13, No. 4, p. 61-69.
8. Grimm S. et al. Memory Efficient Acceleration Structures and Techniques for CPU-based Volume Raycasting of Large Data. Proceedings of the IEEE Symposium on Volume Visualization and Graphics. 2004. P. 1-8.