

На правах рукописи

Мельман Сергей Владимирович

**РАЗРАБОТКА И ИССЛЕДОВАНИЕ  
АЛГОРИТМОВ И ПРОГРАММНЫХ СРЕДСТВ  
ВИЗУАЛИЗАЦИИ ОБЪЕМОВ**

05.13.11 – математическое и программное обеспечение вычислительных машин,  
комплексов и компьютерных сетей

**Автореферат**  
диссертации на соискание ученой степени  
кандидата технических наук



Владивосток — 2013

Работа выполнена в лаборатории машинной графики Института автоматизации и процессов управления Дальневосточного отделения РАН

Научный руководитель: Бобков Валерий Александрович,  
доктор технических наук

Официальные оппоненты: Грибова Валерия Викторовна,  
доктор технических наук, старший научный  
сотрудник, заведующая лабораторией  
интеллектуальных систем ИАПУ ДВО РАН

Фищенко Виталий Константинович,  
кандидат технических наук, старший  
научный сотрудник, заведующий  
лабораторией анализа океанологической  
информации ТОИ ДВО РАН

Ведущая организация: Институт вычислительной математики и  
математической геофизики Сибирского  
отделения РАН, г. Новосибирск

Защита состоится 22 марта 2013 г. в 12:00 часов на заседании  
диссертационного совета Д.005.007.01 в Институте автоматизации и процессов  
управления ДВО РАН по адресу: 690041, г. Владивосток, ул. Радио, 5.

С диссертацией можно ознакомиться в библиотеке Института автоматизации  
и процессов управления Дальневосточного отделения РАН.

Автореферат разослан \_\_\_\_ февраля 2013 г.

Ученый секретарь  
диссертационного совета Д.005.007.01, к.т.н.



А.В. Лебедев

## ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

**Актуальность темы.** Компьютерная графика является сегодня мощным и необходимым инструментом при решении широкого круга задач, как в научных исследованиях, так и в практических приложениях. Средства 3D визуализации используются для наглядного представления результатов компьютерного моделирования, для визуализации данных разнообразных измерений применительно к изучению природных явлений и к контролю производственных процессов, в медицине, при создании тренажеров и систем виртуальной реальности. Одним из актуальных разделов компьютерной графики является разработка программно-алгоритмических средств визуализации объемов, которые востребованы в научной визуализации, в метеорологии и океанологии, в медицинской диагностике и в ряде других приложений. На сегодняшний день усилиями отечественных и зарубежных разработчиков создан достаточно мощный методологический и программно-алгоритмический задел в этом направлении. Большой вклад в создание теоретической базы и в разработку программных средств 3D визуализации и визуализации объемов внесли коллективы разработчиков ИПМ РАН, ИВМиМГ СО РАН, ИАиЭ СО РАН, лаборатории компьютерной графики и мультимедиа Факультета ВМК МГУ имени М.В. Ломоносова и другие. Среди зарубежных исследователей можно отметить Стэнфордский университет, разработавший пакет прикладных программ 3D визуализации VolPack, Канадский университет Макгилла и их пакет программ Vis5D, а так же американских исследователей из проекта NOAA при поддержке NASA.

Текущий момент развития 3D графики и ее разнообразных приложений характеризуется постоянным ростом объемов визуализируемых данных и возросшими требованиями к графическим программным средствам, основными из которых являются реалистичность изображений и скорость обработки данных. С другой стороны, быстрый прогресс вычислительной техники стимулирует разработчиков к созданию более совершенных программно-алгоритмических средств. Поэтому дальнейшие усилия исследователей направлены на повышение реалистичности визуализации, повышение быстродействия программных средств и развитие возможностей визуального анализа данных. Для достижения указанных целей необходимо совершенствование существующих и разработка новых методов, алгоритмов, структур данных и реализация параллельных вычислений с использованием современных аппаратных возможностей.

В настоящей диссертационной работе исследуются и предлагаются алгоритмические решения 3D визуализации с применением октантных структур данных, рендеринга по базовым изображениям и аппаратных графических возможностей на базе OpenGL. Другой аспект исследований – реализация многопроцессорной параллельной обработки данных на базе существующих графических ускорителей средствами шейдер- и CUDA- технологий и с использованием многоядерной архитектуры центральных процессоров. В целом предлагаемые решения направлены на повышение реалистичности

изображений моделируемых сцен, наглядности визуализации объемов и на повышение скорости обработки больших объемов данных, отвечающей режиму реального времени. Практическая составляющая исследований включает разработки приложений для научной визуализации пространственных данных и визуализации данных медицинской диагностики. Самостоятельным результатом диссертационной работы является разработка системы визуализации физических полей синоптических объектов (прежде всего тропических циклонов), где акцент делается на обработку и анимационную визуализацию больших объемов данных с развитыми возможностями визуального анализа динамики полей.

**Целью диссертационной работы** является разработка и исследование методов и алгоритмов визуализации воксельной модели сцены и анимационной визуализации объемов с реализацией графических программных средств.

Для достижения поставленной цели в диссертационной работе ставились и решались следующие **задачи**:

1. Разработка и исследование октантной структуры данных, алгоритмов трассировки лучей в октантных деревьях и программных средств визуализации воксельной модели сцены с применением параллельных вычислений.
2. Разработка метода анимационной визуализации воксельной сцены с применением многопроцессорной обработки.
3. Разработка и исследование алгоритмов анимационной визуализации объемов с использованием графической аппаратной поддержки.
4. Разработка системы визуализации синоптических объектов с возможностями визуального анализа структуры и динамики физических полей синоптических объектов атмосферы и океана.

**Методы исследования.** В работе использовались методы компьютерной графики и возможности аппаратных графических средств, методы обработки изображений и компьютерного зрения, обработки спутниковых данных, объектно-ориентированного программирования, элементы векторной алгебры и вычислительной геометрии.

#### **Научная новизна.**

1. Предложен новый целочисленный алгоритм дискретной трассировки лучей в октантных деревьях, исследованы варианты его модификации с реализацией когерентности траекторий лучей.
2. Предложен метод, основанный на применении ограничений эпилеплярной геометрии и рельефных текстур для анимационной визуализации воксельной модели сцены.
3. Разработаны алгоритмы визуализации динамики скалярных полей, новизна которых состоит в комплексном применении шейдер-технологии, распараллеливании вычислений на многопроцессорной системе МВС и параллелизма на графических ускорителях по технологии CUDA.

**Практическая значимость диссертации.** Полученные в результате исследований методы, алгоритмы, структуры данных и программные средства могут быть использованы при создании прикладных систем различной проблемно-ориентированной направленности для статической и анимационной визуализации объемов и произвольных 3D-сцен.

Разработанная система «TomoView3D», предназначенная для визуализации данных компьютерного томографа, внедрена и используется в Дальневосточном окружном медицинском центре Росздрава (г. Владивосток). Система визуализации физических полей синоптических объектов в атмосфере и океане находится в опытной эксплуатации в центре спутникового мониторинга ИАПУ ДВО РАН и применяется при исследовании динамики тропических циклонов. Также система находится в опытной эксплуатации в Приморском управлении по гидрометеорологии и мониторингу окружающей среды и используется для визуализации результатов работы WRF модели, используемой для прогнозирования погоды.

Материалы исследований использовались в учебном процессе при чтении курса лекций «Компьютерная графика» в Дальневосточном государственном университете и Дальневосточном государственном техническом университете.

**Положения выносимые на защиту:**

1. Алгоритмы трассировки лучей в октантных деревьях.
2. Метод анимационной визуализации октантных сцен.
3. Технология визуализации физических полей синоптических объектов.
4. Алгоритмы параллельных вычислений на многопроцессорных системах с использованием аппаратной графической поддержки.

**Обоснованность и достоверность результатов** обеспечиваются корректным применением аппарата компьютерной графики, векторной алгебры, оптимизации, подтверждены экспериментальными исследованиями и результатами практической эксплуатации разработанных программных продуктов.

**Апробация работы.** Основные научные положения и результаты диссертационной работы докладывались и обсуждались на научных семинарах Института автоматизации и процессов управления ДВО РАН; на Дальневосточной математической школе-семинаре имени академика Е.В. Золотова (Владивосток, 2005, 2012); на региональной научно-технической конференции «Технические проблемы освоения мирового океана» (Владивосток, 2007); на международных конференциях по компьютерной графике и ее приложениям «GraphiCon» (Новосибирск, 2005, 2006, Москва, 2007, 2011) и «The First Russia and Pacific Conference on Computer Technology and Applications» (Владивосток, 2010).

**Публикация результатов работы.** По материалам диссертации опубликовано 16 печатных работ, из них 8 статей в журналах, входящих в перечень ВАК.

**Структура и объем работы.** Диссертация состоит из введения, 4-х глав, заключения и списка литературы. Содержание работы изложено на 146 страницах. Список литературы включает 119 наименований. В работе содержится 78 рисунков и 6 таблиц.

## СОДЕРЖАНИЕ РАБОТЫ

**Во введении** обоснована актуальность выполненных исследований, сформулированы цель и задачи диссертационной работы, раскрыты научная новизна и практическая значимость.

**Первая глава** посвящена обзору литературы. В ней анализируется современное состояние исследований в области визуализации объемов и воксельной графики. Первая часть главы посвящена обзорам методов воксельной графики, рассматриваются основные методы визуализации объемов. Акцент делается на производительности алгоритмов. Далее проводится обзор возможностей существующих систем визуализации синоптических данных. Рассмотрены приложения визуализации 2D и 3D данных полученных при помощи метеорологических станций, датчиков, спутниковых снимков, моделирования локальных и глобальных синоптических процессов. В конце главы обозначены направления исследований для повышения эффективности обработки и визуализации растущих объемов данных в контексте создания приложений.

**Вторая глава** диссертации посвящена исследованию и разработке алгоритмов воксельной графики. Приведены общие понятия, представлены применяемые в работе модели кодирования октантных структур, описаны алгоритмы квантизации цвета и нормали, сокращающие объемы требуемой оперативной памяти ЭВМ, и алгоритм генерации октантных структур. Основное внимание уделено разработке алгоритма трассировки октантных деревьев и его модификаций за счет использования когерентности цвета и траекторий, а так же за счет использования аппаратных возможностей современных программируемых графических ускорителей с применением технологии CUDA (Compute Unified Device Architecture). Реализация алгоритмов прямого рендеринга основана на использовании послойной структуры октодерева и адаптивной глубины.

В алгоритмах трассировки октантных структур использовалась структурная модель октантного дерева, в которой в каждом узле дерева хранятся ссылки на родительский и дочерние узлы. Узлу соответствует часть пространства сцены, ограниченная параллелепипедом (боксом). С учетом преимуществ объектно-ориентированного подхода эта модель, несмотря на некоторую избыточность, позволила добиться хороших скоростных показателей в описываемых алгоритмах. Для хранения на внешних носителях использовалась простая линейная модель, которая совместно с использованием

методов сжатия информации без потерь дает эффект сжатия воксельного пространства в 150 — 200 раз.

Для снижения затрат оперативной памяти, что весьма актуально при большой глубине октантных деревьев, был реализован метод «медианного сечения» применительно к квантизации цвета. Так же был предложен алгоритм квантизации нормалей, суть которого заключается в использовании фиксированного набора векторов для выбора требуемой нормали. Этот набор из векторов равномерно распределен по направлениям на единичной сфере. В этом случае максимальный угол отклонения нормали из набора и реальной нормали

можно вычислить по формуле:  $\beta_{\text{ошибка}} = \arcsin\left(\frac{2}{\sqrt{N}}\right)$ . При таком подходе достаточно 2 байта для кодирования нормали, а ошибка не приводит к заметным дефектам.

*Генерация октантных структур.* Разработаны алгоритмы генерации октантных структур по регулярной решетке и по полигональной модели объектов. Формулируя задачу в терминах принадлежности пространственной точки объекту, построение дерева можно провести в два этапа:

1. Генерация полного октантного дерева, за исключением вокселей, лежащих вне объекта;
2. Объединение внутренних вокселей в боксы минимального уровня.

При этом принадлежность вокселя объекту для регулярных решеток определяется по пороговому значению, а для триангуляционных сеток, в качестве признака принадлежности вокселя объекту, предлагается использовать порядок обхода вершин ближайшего треугольника. Построенные таким образом октантные структуры удовлетворяют требованиям точности. Единственное ограничение на триангуляционные объекты — они должны быть замкнуты.

*Целочисленный алгоритм трассировки октантных структур.* Идея алгоритма поиска боксов, пересекаемых лучом, заключается в рекурсивной обработке пространства сцены с переходом к новой системе координат на каждом уровне октантного дерева.

На первом этапе вычисляется главное направление — это координатная ось, проекция луча на которую имеет наибольшее значение по модулю. Все лучи в сцене приводятся к новой системе координат, в которой бокс сцены является кубом с гранью единичного размера. Нумерация октантов осуществляется согласно их положению вдоль осей координат, где первый бит показывает сдвиг вдоль оси X, второй — Y, третий — Z (рис. 1).

На втором этапе находится «байт состояния» для трех проекций бокса и луча на координатные плоскости. «Байт состояния» — это байт, в котором каждому пересеченному октанту соответствует бит, порядковый номер этого бита соответствует номеру октанта.

Для поиска «байта состояния» достаточно найти номер интервала, которому принадлежит точка пересечения луча и средней линии проекции бокса (рис. 2). Зная номер интервала и направление луча вдоль средней линии для каждой проекции, можно выбрать байт состояния из заранее заданного табличного набора вариантов. Далее, по байту состояния и байту проекции

луча, из таблицы выбирается последовательность пересеченных узлов. Байт проекции луча составляется из трех битов соответствующих наклону луча относительно каждой из координатных осей. Эксперименты показали высокую эффективность предложенного алгоритма.

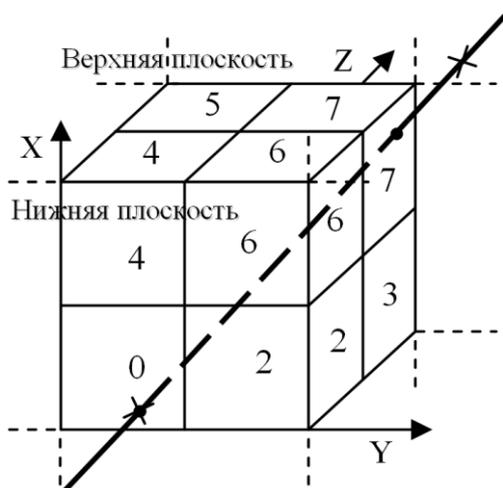


Рис. 1. Нумерация октантов в боксе.

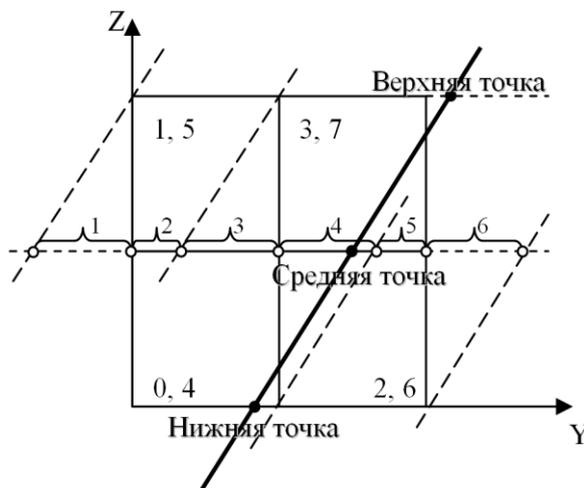


Рис. 2. Поиск интервала.

*Когерентность траекторий.* Для дополнительного повышения скорости рендеринга октантных структур был предложен подход, основанный на использовании когерентности траекторий в октантных деревьях. Идея оптимизации заключается в использовании общего пути в октантном дереве для двух лучей проходящих через соседние пиксели на экране (рис. 3).

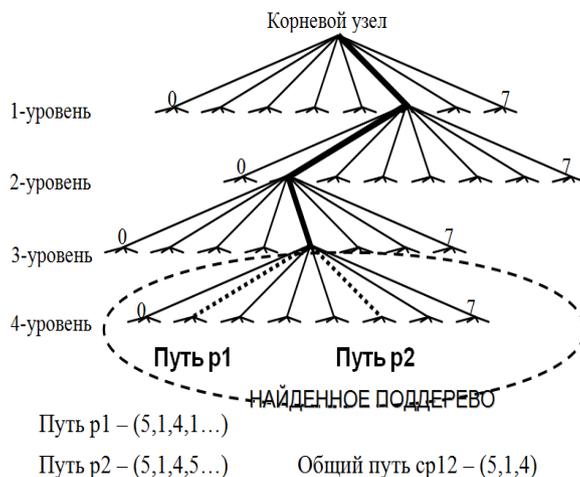


Рис. 3. Пути к граничным вокселям по опорным лучам. Общий путь.

В октантном пространстве близость соседних пикселей проявляется в наличии общего начального участка траектории в октантном дереве. Таким образом, чем ближе друг к другу расположены граничные воксели двух близких лучей, тем больше у этих лучей будет общий путь в октантном пространстве. Предложена схема использования этого типа оптимизации для последовательно расположенных пикселей на экране и для пикселей, сгруппированных в площадки. Экспериментально установлено, что наибольший эффект достигается при использовании площадок пикселей 5x5.

*Когерентность цветов.* Еще одно повышение скорости рендеринга было получено за счет использования когерентности цветов соседних пикселей, суть которой заключается в отказе от трассировки для пикселей, соседи которых не отличаются или отличаются незначительно по цвету. Совместное использование двух предложенных оптимизаций алгоритма визуализации октантных деревьев дает прирост в скорости более чем в два раза без потери качества.

*Прямой рендеринг октантных структур.* В работе также рассмотрен алгоритм прямого рендеринга октантных структур. В отличие от аналогов в предлагаемом алгоритме учитывается упорядоченность октантов в родительском боксе. Таким образом, при сканировании всего дерева, для вывода на экран используется заранее выбранный порядок перебора октантов, который обеспечивает выполнение условия: ближние октанты при проецировании перекрывают дальние. Это позволяет отказаться от традиционного Z-буфера и предварительной сортировки всего массива вокселей. В предложенном алгоритме используется введенное в алгоритме трассировки понятие байта проекции луча. Этот байт однозначно определяет последовательность октантов, которая обеспечивает перекрытие ближними дальних. Необходимое условие корректной работы алгоритма - наблюдатель находится вне бокса сцены. Описана схема, однозначно определяющая последовательность октантов для работы алгоритма прямого рендеринга в зависимости от направления луча зрения наблюдателя. Эта схема обеспечила табличный выбор последовательности на предварительном этапе и позволила избежать дополнительных вычислений на промежуточных уровнях. Для закраски пикселей на экране используется прямоугольная кисть, размеры которой так же рассчитываются на предварительном этапе. Для определения размеров кисти использовались минимальные размеры параллелепипеда содержащего проекцию самого дальнего от наблюдателя вокселя. В случае, когда размеры кисти оказываются меньше пикселя, используется неполный проход по дереву, то есть реализуется принцип «мультиразрешения» октантного дерева и адаптивного выбора глубины просмотра. Для реализации «мультиразрешения» разработан метод адаптивного выбора уровня просмотра дерева для текущего узла и заданного положения наблюдателя. С учетом размера кисти для самого дальнего и самого ближнего к наблюдателю вокселя, а так же задаваемого пользователем критерия качества, метод позволяет выбрать уровень дерева, ниже которого спускаться не имеет смысла (то есть кисть будет меньше пикселя). Поскольку для экономии оперативной памяти на промежуточных узлах не хранится информация о граничных свойствах объекта используется самый ближний к наблюдателю граничный воксель, содержащийся в выбранном узле. Процедура выбора ближайшего граничного вокселя идентична алгоритму послойного обхода дерева от дальнего вокселя к ближнему, только обход идет от ближнего к дальнему.

*Многопроцессорная обработка.* Для повышения быстродействия алгоритма трассировки октантных деревьев были проведены исследования по реализации параллельных вычислений на графическом ускорителе средствами языка CUDA.

Эксперименты показали, что прямой перенос целочисленного алгоритма на CUDA не эффективен, что объясняется интенсивным использованием относительно не быстрой глобальной памяти ускорителя. Поэтому была реализована специальная версия алгоритма для работы на графических ускорителях. Для повышения скорости рендеринга исходный алгоритм был модифицирован в двух направлениях. Во-первых, вместо ссылочной была выбрана линейная индексированная модель октантного дерева, в которой каждое поддереву представляется непрерывным участком памяти. Это позволило воспользоваться кэшируемой текстурной памятью. Во-вторых, эффективный для центрального процессора табличный выбор решений был заменен на вычисления «на лету», по причине медленного доступа к таблицам. В алгоритме для CUDA, в силу ограничений архитектуры, рекурсия была заменена условными переходами, с сохранением минимума информации для возврата на верхний уровень. Необходимый минимум — это индекс родительского бокса в структуре октодерева и номер текущего октанта для правильного обратного преобразования системы координат. При переходах между уровнями, преобразования системы координат приводили текущий бокс к единичному с вершиной в нуле. Такое преобразование позволило свести к минимуму количество обращений к глобальной памяти и количество вычислений пересечения луча и бокса. Эффективность реализованной версии алгоритма была проверена в ряде вычислительных экспериментов.

**Третья глава** диссертации посвящена анимационной визуализации октантных структур. В главе делается акцент на интерактивную анимацию, для реализации которой используются понятия базовых и промежуточных кадров. Для построения базовых кадров используются алгоритмы трассировки, представленные во второй главе. Для построения промежуточных кадров используются алгоритмы, основанные на предсказании новых положений наблюдателя, использовании рельефных текстур, эпиполярной геометрии (рис. 4), а так же алгоритмы, работающие с наборами базовых изображений. Так же в главе рассматриваются вопросы применения параллелизма на кластере для анимационной визуализации.

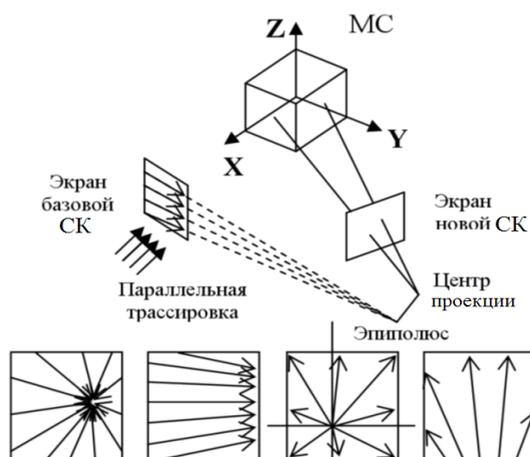


Рис. 4. Расположение камер и возможные схемы обхода.

В работе описывается методика построения матриц репроекции, которые используются для преобразования пикселей с «глубиной» из базовой камеры в

изображения для нового положения камеры. Использование рельефных текстур позволяет добиваться постоянной скорости рендеринга, независимо от сложности сцены и ракурса, но возникают традиционные сложности с перекрытием ближнего плана дальним, появлением артефактов и дыр. Для решения первой проблемы было предложено использовать сортировку пикселей, основанную на эпполярной геометрии. Схему обхода пикселей базового изображения определяет центр симметрии, который является эпполярной точкой (эпполюс) и вычисляется как пересечение плоскости экрана с линией, соединяющей центры проекций. Зная матрицу репроекции, центр симметрии и схему обхода, заданную этим центром, можно построить новое изображение без применения Z-буфера.

Проблема появления «дырок» решается применением схемы с использованием двух базовых изображений, эта схема обеспечивает повышение качества промежуточных кадров без увеличения вычислительных затрат, за счет использования половинной выборки из удвоенного количества пикселей. Из рассмотренных вариантов: регулярное разрежение и случайная выборка, выбор сделан в пользу регулярного разрежения, в силу меньшей вычислительной нагрузки. Подобный подход предлагается использовать и для заранее подготовленных фиксированных наборов базовых изображений. Задача сводится к выбору двух или трех наборов ближайших к новому положению камеры.

Для использования аппаратных возможностей графических ускорителей предложен алгоритм перебора всех граничных вокселей, формирующий массив трехмерных точек. Алгоритм предоставляет графическому ускорителю вывод на экран всех сплатов (splat), соответствующих найденным точкам. Закраска и Z-буфер обеспечиваются самим графическим ускорителем. Недостатком подобного подхода является зависимость скорости рендеринга от количества граничных вокселей.

Далее в главе описывается параллельная реализация рендеринга по базовым изображениям с использованием кластера (MBC 1000). Для обеспечения удобного интерфейса взаимодействия пользователя и приложения предложено использовать клиентскую часть, реализованную под управлением ОС Windows, тогда как построение изображения происходит на кластере под управлением ОС LINUX. Взаимодействие кластера и клиентской машины осуществляется по локальной сети. При помощи интерфейса на клиентской машине происходит выбор и передача базы данных базовых изображений на кластер. Далее на кластер в интерактивном режиме поступает положение камеры заданное пользователем. Сформированное на кластере изображение отправляется по сети обратно на клиентскую машину. Процесс рендеринга изображения на кластере обеспечивается двумя подпрограммами: диспетчер и обработчик. Диспетчер отвечает за получение набора данных базовых изображений, выбор необходимых изображений для формирования нового вида, расчет эпполюсов, разбиение базовых изображений на полосы и запуск обработчиков для всех полос. Количество полос выбирается равным количеству свободных процессоров на кластере. В заключении диспетчер собирает данные

от обработчиков, склеивает результат в изображение и передает на клиентскую машину для визуализации (рис. 5). В задачу обработчика входит преобразование полученных на входе полосы пикселей с глубиной в новый вид для новой камеры. В общем случае преобразованная полоса не является полосой на новом виде, поэтому после обработки ей ставится в соответствие целый текстурный слой.

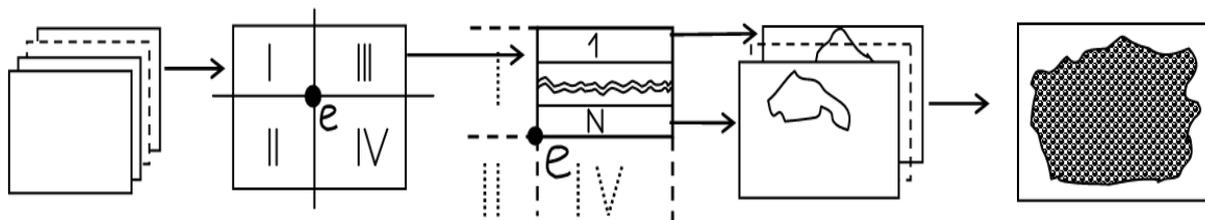


Рис. 5. Формирование изображения на кластер.

Такое статическое распределение работы оправдано, поскольку трудоемкость обработки всех пикселей одинакова для каждой полосы. А так как работа диспетчера и обработчиков строго последовательна, то диспетчер и один из обработчиков могут быть запущены на одном процессоре, чтобы не было простоя. Несмотря на большое количество задач возложенных на диспетчер, общее время работы всего алгоритма значительно больше времени работы диспетчера на всех этапах. В результате эффективность параллельного алгоритма находится на уровне 70% прироста скорости по отношению к линейному ускорению. Подобная схема применена и к распараллеливанию целочисленного алгоритма трассировки октантных деревьев представленного в главе 2. Эксперименты по распараллеливанию целочисленного алгоритма показали, что для более равномерной нагрузки на процессоры кластера количество полос, в случае трассировки, должно быть больше чем количество свободных процессоров. Это связано с неравномерным распределением лучей пересекающих бокс сцены в каждой полосе. Но с другой стороны слишком большое количество полос значительно увеличивает поток данных между процессорами. Наиболее оптимальным было выбрано утроенное количество полос по отношению к количеству свободных процессоров. Прирост скорости для предложенного алгоритма составил 95% по отношению к линейному ускорению. Формула расчета эффективности рендеринга на МВС:

$$E = \frac{T_1}{N \cdot T_N} \cdot 100\%$$

, где  $T_1$  - время рендеринга сцены на 1 процессоре;  $N$  - количество используемых процессоров;  $T_N$  - время рендеринга сцены на  $N$  процессорах.

**Четвертая глава** диссертации посвящена вопросам разработки системы визуализации синоптических данных, предназначенной для визуального анализа синоптических объектов. В качестве основного синоптического объекта рассматривался тропический циклон (ТЦ), данными для которого являются спутниковые измерения.

Разрабатываемая система должна отвечать следующим требованиям: удобный пользовательский интерфейс; работа с различными типами данных в рамках единого интерфейса; многоплановая анимация (в пространстве и во

времени); наращиваемый набор алгоритмов обработки тематической обработки данных и визуализации физических полей; высокая скорость обработки данных, отвечающая требованиям интерактивного режима работы и режима реального времени анимационной визуализации. Для удовлетворения сформулированным требованиям были решены следующие задачи:

1. Разработать схему организации больших объемов данных на внешних носителях, реализующую простой и быстрый доступ к данным.
2. Предусмотреть расширяемый набор модулей загрузки данных (для обеспечения работы с новыми форматами данных) и разработать схему управления загрузкой данных с подключением модулей визуализации и контролем сопутствующих параметров.
3. Разработать схему и программно-алгоритмические средства обработки данных на всех этапах (начиная от зашумленных, сырых и нерегулярных данных и заканчивая получением специальных структур, предназначенных для быстрой визуализации).
4. Разработать набор методов обработки данных (фильтрация, расчет аномалий, выделение ядра ТЦ и др.) и реализовать расширяемый набор пользовательских инструментов для исследования ТЦ.
5. Разработать алгоритмическую базу статической и анимационной визуализации физических полей синоптических объектов.
6. Разработать и реализовать схему многопроцессорной обработки данных для повышения скорости работы системы.

Структура системы визуализации тропических циклонов представлена на рис. 6. Основу ее составляет универсальная программная оболочка, посредством которой пользователь управляет загрузкой данных, выбором методов визуализации и инструментов для проведения визуального анализа данных.



Рис. 6. Структура программной оболочки

На этапе предобработки спутниковых данных выполняется регуляризация, восполнение, интерполяция данных, рассчитываются аномалии циклона с сохранением результатов во внутреннем формате системы.

В настоящей версии система работает с наборами статических и динамических скалярных и векторных пространственных полей, позволяет отображать траектории центров циклонов на равнопромежуточной проекции, рельеф и другие статические 3D-объекты для повышения наглядности.

Основным в системе является принцип независимости алгоритмов визуализации от набора данных. На вход модулю визуализации подается только ракурс и скорость анимации (используется в алгоритмах многочастичной анимации). Подобный подход позволяет использовать различные наборы методов и алгоритмов визуализации на различных наборах данных. Выделяются три группы обрабатываемых данных: скалярные поля, векторные и специфические данные, такие как центры циклонов, рельеф и т. д. В системе реализован принцип универсальности внутренних форматов данных, таким образом, работа модулей визуализации указанного типа данных не зависит от выбора модуля загрузки. Каждый модуль загрузки работает только с определенной группой данных, с определенным типом и от определенного источника, что необходимо для точного учета их специфики при обработке. Результатом работы модуля загрузки является не только загрузка, но и интерпретация данных и приведение их к формату принятому в приложении. Существенным плюсом подобного подхода является возможность совместной обработки данных, полученных из разных источников, например, от нескольких орбитальных спутников одновременно.

К основным прикладным функциональным возможностям системы относятся:

- визуальное обнаружение пространственно-временных характеристик атмосферы, в частности, влажности (интегральной и на различных уровнях), температуры, положение тропопаузы;
- визуализация изоповерхностей различных параметров атмосферы;
- интерактивный анализ атмосферно-океанических аномалий, определение их сущности, происхождения, положения и времени жизни.

Для обеспечения указанных проблемно-ориентированных возможностей в интерфейсе системы предусмотрены следующие графические функции:

- темпоральные и пространственные срезы;
- гибкое управление интерфейсом;
- интерактивное управление модулями загрузки и визуализации;
- интерактивная обработка данных.

К инструментарию относятся такие режимы как:

- управление камерой («сфера», свободный полет, плоский режим, автоматическое слежение за динамическими объектами);
- управление данными (отсечение скаляров, фильтрация);
- управление параметрами модулей визуализации;
- управление динамическими полями.

*Алгоритмы визуализации.* В системе реализован принцип расширяемого набора алгоритмов визуализации. В настоящей версии системы реализованы следующие алгоритмы: метод объемной текстурной визуализации, метод объемной многочастичной визуализации, метод визуализации изоповерхностей,

метод трассировки объемов на шейдерах, метод многочастичной анимации векторных полей.

Алгоритм, основанный на методе объемной текстурной визуализации, использует семейство секущих плоскостей параллельных экрану, 3D-текстуры поддерживаемые современными графическими ускорителями и попиксельную закраску с использованием программируемых шейдеров. Предложена методика, основанная на применении CLUT (Color Look Up Table) для интерактивного управления закраской.

Метод многочастичной визуализации скалярных полей реализует принцип «мельтешения» на экране частиц (в нашем случае это сферы малого диаметра). Вероятность появления и цвет определяется значением скаляра в точке объема. Для достижения наилучшего визуального эффекта пользователь может выбирать количество частиц и их размер.

Для визуализации изоповерхностей применен алгоритм «марширующих кубиков», что обеспечило аппаратную поддержку визуализации.

Алгоритм трассировки объемов на шейдерах является развитием метода 3D-текстурной визуализации. Идея заключается в отображении одной плоскости вместо набора, а изображение строится путем трассировки луча через объем с накоплением цвета в пикселе с использованием CLUT. Алгоритм реализован на шейдерах. Так же в этом алгоритме для повышения информативности данных используется диффузная модель освещения с бликами. В качестве вектора нормали в точке берется вектор градиента, а источник освещения совмещается с наблюдателем. Вводится понятие ориентации пространственной точки, таким образом, в зависимости от направления вектора нормали в точке к или от наблюдателя, присваиваются разные цвета бликам. Это обеспечивает возможность пользователю судить о внутренней структуре скалярного поля. Предварительное преобразование системы координат позволило значительно повысить скорость работы алгоритма.

*Многопроцессорная обработка.* В работе предложена схема многопроцессорной обработки данных, согласно которой осуществляется распределение нагрузки между многоядерным ЦП и графическим ускорителем (CUDA и шейдеры) (см. рис. 7).

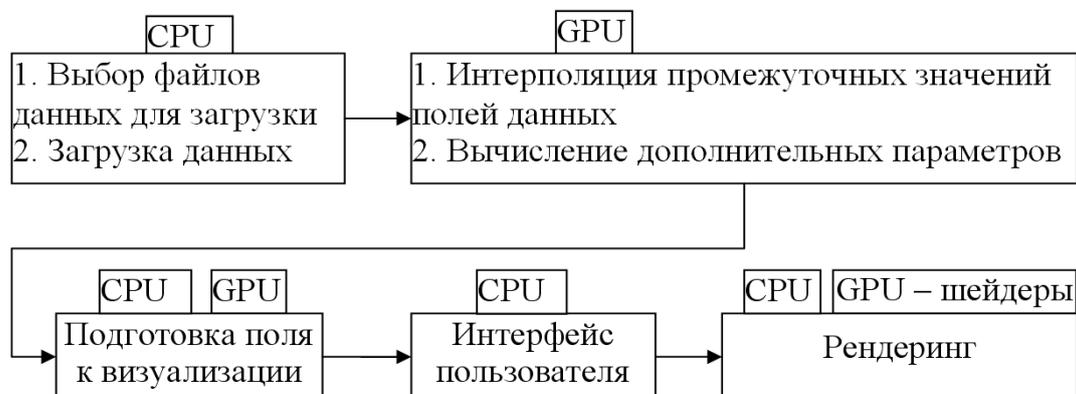


Рис. 7. Распределение вычислений между CPU (центральный процессор) и GPU (графический процессор).

На шейдеры, главным образом, возлагается задача визуализации. На CUDA решается задача предобработки больших массивов данных, вычисление новых параметров по имеющимся данным, интерполяция промежуточных данных для более плавной анимации динамических пространственных полей. Центральный процессор управляет загрузкой данных, обеспечивает управление параметрами модулей в рамках пользовательского интерфейса, подготовкой к визуализации и связями между модулями и системой. Так же на центральном процессоре решаются задачи, которые в силу своей специфики не могут быть реализованы на CUDA.

*Экспериментальная проверка.* Тестирование системы проводилось как на модельных, так и на реальных данных. В качестве реальных данных были взяты данные тропического циклона Melor со временем жизни с 29.09.2009 по 10.10.2009. Профили циклона получены по данным AMSU спутников NOAA 15, 16, 18, 19. Траектория тайфуна взята из базы данных KITAMOTO Asanobu @ National Institute of Informatics. Примененная для визуализации температурных аномалий ядра циклона палитра цветов позволила выделить температурные аномалии. Желтым цветом показаны теплые аномалии более 4.5К (градусов Кельвина), красным цветом — теплые аномалии от 1К до 4.5К, синим цветом — холодные аномалии от 1К до 4.5К, зеленым цветом — холодные аномалии более 4.5К. Шкала серого цвета определяет прозрачность. Траектория циклона показана на рис. 8. А пример визуализации ядра циклона на рис. 9.

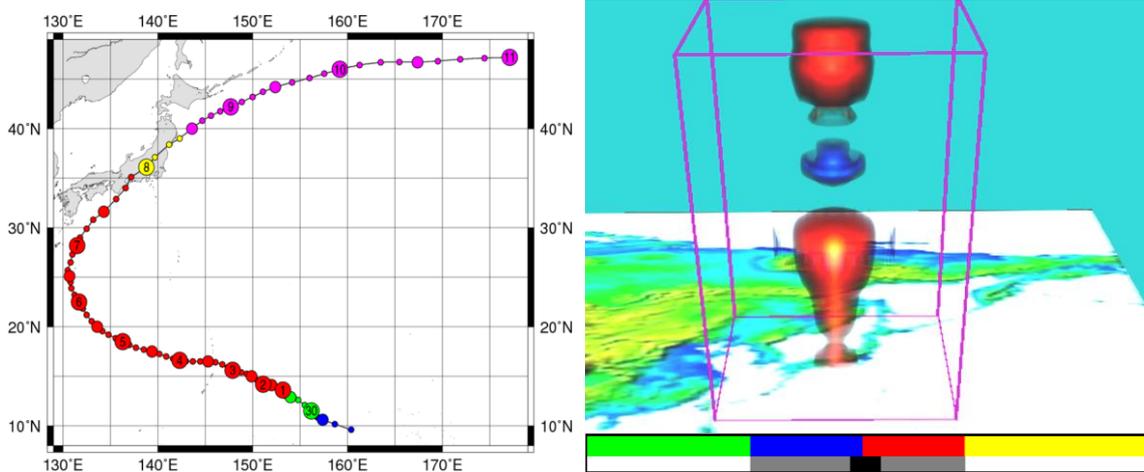


Рис. 8. Траектория циклона Melor. Рис. 9. Циклон Melor над японскими островами и соответствующая CLUT.

Анимационная визуализация динамики циклона позволила увидеть суточное «дыхание» циклона. При прохождении над японскими островами отчетливо видно формирование глаза циклона (зона спокойствия). Ярко выраженное теплое ядро (красного цвета) находится на 20 уровне, что соответствует высоте 10 км. Также в тропосфере отчетливо наблюдается формирование температурных аномальных зон над ядром циклона.

Результаты тестирования системы в целом позволяют говорить об эффективности заложенных в нее решений и удобстве пользовательского интерфейса.

## ОСНОВНЫЕ РЕЗУЛЬТАТЫ РАБОТЫ

1. Разработаны и исследованы алгоритмы трассировки лучей в октантных деревьях в рамках воксельного подхода к генерации и визуализации 3D сцен.
2. Предложен метод анимационной визуализации октантных сцен.
3. Разработаны алгоритмы статической и анимационной визуализации, обеспечивающие возможности визуального анализа структуры и динамики физических полей синоптических объектов. Предложены архитектурные и интерфейсные решения при реализации системы визуализации синоптических объектов.
4. Разработаны алгоритмы и программные средства для организации параллельных вычислений на многопроцессорных системах, включая применение шейдеров и технологии CUDA на графических процессорах.
5. Разработан комплекс программ для визуализации томографических данных.
6. Разработана система анимационной визуализации физических полей синоптических объектов.
7. Проведены экспериментальные исследования предложенных программно-алгоритмических средств с оценкой их эффективности.

## ПУБЛИКАЦИИ ПО ТЕМЕ ДИССЕРТАЦИИ

1. Бобков В.А., Казанский А.В., Морозов М.А., Мельман С.В. Определение векторов ветра водяного пара по данным геостационарных ИСЗ: сравнительный анализ корреляционного и контурного методов // **Исследование Земли из космоса.** – 2004. – №3. – С. 52-59.
2. Бобков В.А., Роньшин Ю.И., Мельман С.В. Визуализация воксельных сцен // **Информационные технологии.** – 2005. – №6. – С. 16-19.
3. Бобков В.А., Роньшин Ю.С., Борисов Ю.С., Мельман С.В. Интерактивная визуализация сложных сцен // **Вестник ДВО РАН.** – 2005. – №6. – С. 87-92.
4. Бобков В.А., Мельман С.В. Рендеринг октантных сцен // **Информационные технологии.** – 2006. – №3. – С. 39-46.
5. Мельман С.В. Визуализация объемов в задачах анализа физических полей синоптических объектов // **Информационные технологии.** – 2008. – №1. – С. 62-66.
6. Мельман С.В., Май В.П. Система объемной визуализации объектов компьютерной томографии // **Информационные технологии.** – 2010. – №5. – С. 68-73.
7. Мельман С.В., Бобков В.А. Параллельная трассировка октантных деревьев на языке CUDA // **Информационные технологии.** – 2011. – №4. – С. 30-36.
8. Бобков В.А., Мельман С.В. Визуализация динамики циклонов // **Информационные технологии.** – 2012. – №2. – С. 49-54.

9. Мельман С.В., Гриняк Т.М. Анимация трехмерных стационарных векторных полей // Электронный журнал «Исследовано в России». – 2004. – №202. – С. 2149-2155.
10. Мельман С.В., Гриняк Т.М. Анимация статических векторных полей в пространстве // Материалы Дальневосточной математической школы-семинара имени академика Золотова. – Владивосток. – 2004. – С. 153-154.
11. Бобков В.А., Мельман С.В., Роньшин Ю.И. Оптимизация трассировки лучей в октантных деревьях // Материалы 15-ой Международной Конференции по Компьютерной Графике и Зрению, ГрафиКон'2005. – Новосибирск. – 2005. – С. 187-195.
12. Мельман С.В., Бобков В.А. Система визуализации пространственных полей синоптических объектов // Материалы 17-ой Международной Конференции по Компьютерной Графике и Зрению, ГрафиКон'2007. – Москва. – 2007. – С. 264-268.
13. Melman S., May V. The system for volume visualization of computed tomography // Proceedings of The First Russia and Pacific Conference on Computer Technology and Applications. – Vladivostok. – 2010. – Pp. 391-393.
14. Melman S., Bobkov V., May V. The system for visualization of synoptic objects // Proceedings of 21st International Conference on Graphics and Vision, GraphiCon'2011 – Moscow. – 2011. – Pp. 103-106.
15. Мельман С.В., Бобков В.А., Май В.П. Система визуализации синоптических объектов // Электронный журнал «Научная визуализация», Национальный Исследовательский Ядерный Университет «МИФИ». – 2011. – Квартал 4. – Том 3. – №4. – С. 2-9.
16. Черкашин А.С., Мельман С.В. Программный комплекс визуализации синоптических данных // Сборник материалов XXXVI Дальневосточной Математической Школы-Семинара имени академика Е.В. Золотова. – Владивосток. – 2012. – С. 404-407.

**Личный вклад автора.** Все результаты, составляющие основное содержание диссертации, получены автором самостоятельно. В работе [1] автором разработан оригинальный целочисленный алгоритм трассировки лучей в октантных структурах, а также его оптимизация с использованием когерентности траекторий. В работах [9, 10] автору принадлежит идея создания алгоритма прямого псевдопослойного рендеринга октантных структур без применения Z-буфера и предварительной сортировки. В работе [2] автором предложен и реализован алгоритм одновременной визуализации набора рельефных текстур без применения Z-буфера с использованием эпиполярной геометрии. В работах [3, 11] автору принадлежит идея создания модифицированного алгоритма многочастичной визуализации стационарных и динамических пространственных векторных полей, а так же реализация системы визуализации динамических векторных полей. В работе [4] автором разработан и реализован алгоритм трассировки октантных деревьев для параллельной архитектуры графических ускорителей на CUDA. В работах [12, 14, 15] автору принадлежит разработка и реализация системы визуализации

пространственных (динамических и статических) полей применительно к синоптическим объектам, реализация алгоритмов для достижения режима реального времени, также автором разработаны и реализованы алгоритмы параллельной предварительной обработки данных полей синоптических объектов средствами графических ускорителей на CUDA. В работах [6 и 13] автору принадлежит реализация приложения для визуализации медицинских данных КТ сканирования. В работе [7] автором реализован алгоритм трассировки октантных деревьев на CUDA. В работе [8] автором реализован метод пространственно-темпоральной визуализации динамики циклона с помощью вычисляемых по спутниковым данным температурных аномалий, так же автору принадлежит реализация алгоритмов обработки и визуализации данных с применением шейдер и CUDA технологий. В работе [16] автору принадлежит разработка архитектуры и интерфейса программного комплекса визуализации синоптических данных.

Мельман Сергей Владимирович

РАЗРАБОТКА И ИССЛЕДОВАНИЕ  
АЛГОРИТМОВ И ПРОГРАММНЫХ СРЕДСТВ  
ВИЗУАЛИЗАЦИИ ОБЪЕМОВ

Автореферат

Подписано к печати 12.02.2013

Усл. печ. л. 1,0

Уч.-изд. л. 0,8

Формат 60x84/16

Тираж 100

Заказ 1

---

Издано ИАПУ ДВО РАН. Владивосток, Радио, 5

Отпечатано участком оперативной печати ИАПУ ДВО РАН

Владивосток, Радио, 5