

Программно-аппаратный комплекс интерактивных мультимедийных презентаций

Б.С. Мазурок, Б.С. Долговесов, Е.И. Коростелев, Т.Н. Артиков, А.Н. Артиков
Институт Автоматики и Электростроения СО РАН, Новосибирск, Россия
kore3d@gmail.com

Аннотация

В данной работе представлены особенности реализации современного программно-аппаратного комплекса интерактивных презентаций. Разработанная презентационная система предоставляет широкие возможности для подготовки и демонстрации различных мультимедийных материалов. Обеспечение интерактивного совмещения виртуальных трехмерных объектов и мультимедийных данных (видео, презентации, текст) с изображениями реальных персонажей позволяет существенно улучшить качество подачи материала.

Ключевые слова: виртуальный презентер, мультимедиа, визуализация реального времени.

1. ВВЕДЕНИЕ

В последнее десятилетие происходит развитие и адаптация систем виртуальной реальности для подготовки демонстрационных и образовательных материалов: учебные демонстрации, демонстрация товаров, визуализация бизнес процессов. Разрабатываемые системы строятся на базе технологии интегрированной виртуальной реальности. Данная технология предполагает интерактивное объединение в виртуальной среде моделей пространственных объектов, явлений и процессов с изображением реального персонажа (лектора, экскурсовода). Такое решение способствует существенно лучшему восприятию и пониманию излагаемого материала.

Авторами работы ранее разработана и представлена соответствующая система [1,3], успешно применяемая в образовательных целях (подготовка видеопособий, учебный процесс в специализированных школах). Основываясь на многолетнем опыте ее разработки и применения, а также современных требованиях, разработан программно-аппаратный комплекс виртуального презентера.

Разработанное решение обеспечивает формирование и визуализацию изображений (включая стереоизображения) виртуального окружения различной тематической направленности в комбинации с видеоизображением докладчика, интерактивно взаимодействующего с моделями демонстрируемых образцов. Интерфейс пользователя (рис. 1) ориентирован на использование сенсорного экрана и позволяет в несколько касаний выбрать демонстрируемый медиа-контент. Таким образом, докладчик, без привлечения сторонних специалистов, легко управляет ходом презентации, а также может переключать режимы микширования, медиа-источники и использовать вещание в сеть.



Рис. 1: Интерфейс управления презентацией

2. СТРУКТУРА КОМПЛЕКСА

Программно-аппаратный комплекс включает следующие аппаратные компоненты:

Система визуализации, выполненная на базе современного ПК – компьютера с высокопроизводительным графическим процессором.

Система отображения результирующего изображения (виртуальной среды, совмещенной с видеоизображением докладчика) на сенсорный монитор, проектор.

Интерактивные средства управления презентацией: джойстик, манипулятор «мышь», планшетный ПК.

Съемочное оборудование для записи видеоизображения докладчика, специализированный одноцветный фон (синий или зеленый) система освещения и звукозаписи.

Программное обеспечение комплекса также состоит из набора компонентов. Приведем некоторые из них:

Менеджер ресурсов обеспечивает экспорт и загрузку ресурсов сцены, управление интерактивными средствами.

Подсистема визуализации осуществляет отображение трехмерных сцен, текстурирование поверхностей трехмерных моделей, выполнение шейдерных операций.

Менеджер сцены осуществляет управление содержимым сцены в процессе отображения.

В следующем разделе рассмотрены особенности и специфичные для решаемых задач функции подсистемы визуализации. Их производительность существенно влияет на производительность всей системы, а набор реализованных в нём функций определяет применимость разработанной системы визуализации.

При разработке основы программных компонент существенное внимание уделялось гибкости системы: расширения системы новыми компонентами и конфигурирование всех параметров системы.

Расширяемость современных систем визуализации обеспечивается благодаря использованию компонентной архитектуры. В разработанной системе был расширен данный подход и предложена унифицированная структура хранения и обработки данных. В работе используется разработанный формат XQL, иерархическая база данных и алгоритмы ее функционирования [4]. Особенностью выбранного подхода является: межмодульное взаимодействие через базу данных (текстовый тегируемый протокол обмена), хранение данных произвольного типа (в том числе последовательности команд). Таким образом, интерфейсы модулей отделены друг от друга и зависимости между ними определяются только данными. При таком подходе модули более ранних версий интерпретируют только известные им данные, а неизвестные (добавленные для более поздних версий) просто игнорируют.

База данных используется для доступа к произвольным свойствам системы. Допускается частичное дублирование данных в программных модулях для более эффективного использования аппаратных ресурсов (отображение в видеопамять, предварительная компиляция эффектов, кэширование крупных структур). Для того, чтобы обеспечить обновление этих данных, применяется система обработки событий – каждый модуль регистрируется на уведомления о чтении и записи определенного набора структур в базу данных.

3. ПОДСИСТЕМА ВИЗУАЛИЗАЦИИ

Современные системы визуализации поддерживают десятки визуальных спецэффектов [2,6]. Кроме этого, используемые в сцене изображения могут обрабатываться различными фильтрами (например, фильтром гамма-коррекции, выделения актёра из монохромного фона). А на последнем этапе построения кадра используется последовательность спецэффектов (глубина резкости, свечение и другие).

Спецификой визуализации разработанного комплекса является способность формировать изображение интегрированной виртуальной среды (схема представлена на рис. 2). Изображение реального персонажа (лектора), взаимодействующего с изучаемыми виртуальными моделями, совмещается с тематическим виртуальным фоном и виртуальной моделью изучаемого объекта.

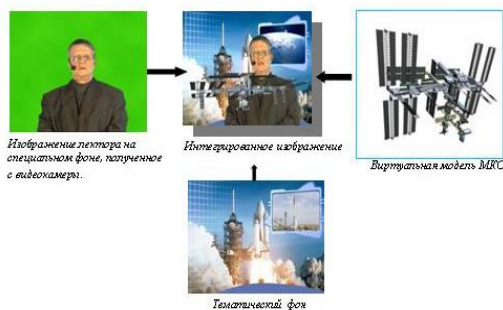


Рис. 2: Принцип формирования интегрированной виртуальной среды

В следующих подразделах рассматриваются основные компоненты подсистемы. Предлагаемый метод выделения актёра вынесен в соответствующий раздел.

3.1 Виртуальная сцена

Описание трехмерной сцены включает: описание трехмерных объектов, последовательностей их изменения во времени, настройки эффектов (параметры шейдеров), свойства отображения (фиксированное разрешение, аспект) и другие параметры. Все эти данные располагаются в иерархическом виде в разработанной базе данных.

В начале подготовки нового кадра производится обход графа сцены (иерархических связей), в процессе которого определяются видимые объекты (отсечение по пирамиде видимости с использованием ограничивающих сфер), строится список объектов для растеризации, производится их сортировка по материалам и в зависимости от удаленности от текущей виртуальной камеры. В узлах графа сцены располагаются виртуальные камеры, источники света и объекты. При этом виртуальный объект сцены может быть, либо обычной трехмерной геометрией (сеткой из треугольников), либо плоскостью, ориентированной на наблюдателя, либо ортогонально проецируемой поверхностью (визуализация элементов пользовательского интерфейса либо режима «оверлей») с текстурованием «тексель-в-пиксель». Все объекты сцены могут быть заданы в Autodesk 3ds Max и загружены в систему визуализации по необходимости.

Любые объекты сцены могут быть анимированы. Так, с течением времени могут меняться свойства объекта, траектории движения и позиция на траектории. Изменение анимационного времени, в зависимости от требований, рассчитывается по среднему времени кадра (для сохранения плавности анимации при буферизации кадров), либо синхронизировано с видеопотоком.

3.2 Система материалов

Материал - совокупность визуальных свойств поверхности объекта. Изображение вычисляется на основе описания его геометрии, материала, свойств сцены (например, освещение и туман). В ходе вычислений блоки графического процессора исполняют специальный код (шейдер), определяющий методы обработки поверхности объекта. Для всех трёхмерных объектов может быть указан универсальный шейдер, но это неэффективно, поскольку приводит к существенным потерям производительности. Вместо этого реализована автоматизированная генерация кода шейдера, в зависимости от свойств материала и ситуации на сцене.

В разработанной системе вводятся новые типы шейдеров: шейдер поверхности, освещения. Кроме этого, при их сборке в шейдер для графического конвейера (пиксельный шейдер) используется условная компиляции. Такой подход позволяет описывать поверхность объекта (цвет, нормаль) и модель расчета освещения независимо друг от друга и комбинировать их произвольным образом. При этом за счет условной компиляции из кода шейдера исключаются лишние инструкции, что улучшает производительность,

При разработке системы освещения основной задачей было повторение локальной модели освещения Autodesk 3ds Max (модель освещения Фонга). Также реализованы более сложные модели – например, освещение по Куку-Торренсу.

Для визуализации теней используются несколько методов (в зависимости от требований): статические (предвычисленные тени) и динамические - метод карт теней и тени в экранном пространстве (Screen Space Ambient Occlusion). Список объектов, отбрасывающих и принимающих тень, может быть задан в программе трехмерного моделирования.

3.3 Конвейер обработки изображений

Реализованная подсистема визуализации поддерживает возможность конфигурирования последовательной обработки изображений – набора последовательно применяемых к изображениям на графическом процессоре эффектов. Описание задается в виде графа фильтров, обрабатывающих изображения [2]. При этом входных и выходных изображений у фильтра может быть несколько. Для каждого изображения могут быть указаны различные параметры: абсолютные или относительные размеры, цветовое пространство, формат пикселей. В качестве входных изображений могут использоваться изображения с предыдущих кадров (временная фильтрация). На рисунке (Рис.3) представлен эффект глубины резкости.

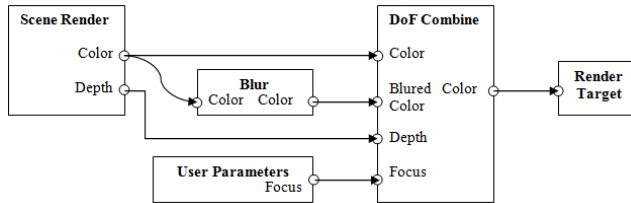


Рис. 3: Граф эффекта глубины резкости

Последовательная обработка текстур применяется как до визуализации (преобразование форматов, выделение актёра), так и после: цветовая коррекция, имитация расфокусировки камеры, туман, эффект смазывания при движении.

3.4 Воспроизведение HD-видео

Обработка видеоданных высокой четкости требует серьезных вычислительных ресурсов – необходима буферизация и декодирование большого объема данных (например, из YUV2 в RGB формат). Для проигрывания видеопотоков реализована асинхронная обработка потока данных (декодирование средствами DirectShow, передача в видеопамять).

Для каждого видеопотока выделяется отдельная нить исполнения и в базе данных размещаются: описание, команды управления, идентификатор очереди кадров. Идентификатор очереди кадров используется для запроса дескрипторов разделяемой видеопамети и информирования о состоянии кадров (их готовности). Таким образом, модули-источники видеоданных (с камер, диска) пишут кадры без дополнительных копирований через системную память.

В настоящий момент широко распространены источники чересстрочного сигнала – профессиональные и любительские видеокамеры. Для правильного отображения на устройствах с прогрессивной разверткой в разработанной системе используется метод устранения чересстрочности YADIF [7]. Метод адаптирован для исполнения на графическом процессоре и используется в варианте для формата YUV2.

Любой видеопоток может быть назначен в качестве «живого фона». При наличии в видеопотоке информации о глубине, может быть задано произвольное расположение актёра и элементов фона относительно друг друга. В обоих случаях применяется точная синхронизация кадров динамического фона со всей анимацией виртуальной сцены, поскольку без нее возникают неприемлемые визуальные дефекты (например, при движении виртуальной камеры).

3.5 Рисование на 3D объектах

Одним из поддерживаемых средств интерактивного взаимодействия является возможность рисовать поверх произвольного объекта сцены. Автоматически определяется какому объекту принадлежит выбранная пользователем точка

и какие текстурные координаты ей соответствуют. В итоге, создается дополнительная динамическая текстура, используемая как слой во время визуализации объекта.

Для определения выбранного в сцене объекта используется отдельный проход визуализации трехмерной сцены. Каждый объект рисуется со специальным шейдером, записывающим в цвет пикселя код: номер рисуемого объекта (либо группы треугольников) и текстурные координаты. Результат этого прохода считывается перед началом прохода визуализации следующего кадра. После этого информация передается в подсистему генерации динамического рисунка пользователя.

4. ВЫДЕЛЕНИЕ АКТЁРА

Для построения маски прозрачности по видео-изображению актёра был разработан оригинальный алгоритм, ориентированный на упрощенные условия съёмки. Для выделения актёра из фона необходимо для каждого пикселя определить какая его часть (α – вес пикселя) должна попасть в результирующее изображение [5]. Предполагается, что актер снимается на одноцветном фоне (допустимы дефекты освещения, неровности, вкрапления) и цветовой тон фона отсутствует на актёре. Таким образом, цвет фона представляют собой некоторую область в цветовом пространстве YUV, размер которой в плоскости UV соответствует неоднородности цвета, а в направлении Y (яркости) – неоднородности освещения. Так как неоднородность освещения даёт более значимый вклад, область цветов фона получается вытянутой вдоль оси Y, поэтому для удобства расчётов эту область можно ограничить эллипсоидом, вытянутым по оси Y.

Для разделения всех трёх областей (фона, актёра и перехода актёр-фон) используются два эллипсоида, оси которых совпадают. Внутренний эллипсоид ограничивает цвета фона, внешний цвета актёра, пространство между ними – переход актёр-фон. Таким образом, в области цветов фона каждому пикселю будет поставлена в соответствие $\alpha = 0$, в области актёра $\alpha = 1$, а для цветов не принадлежащих двум этим множествам α лежит в пределах от 0 до 1, в зависимости от того, к какой области цветов ближе находится цвет пикселя.

Пусть $C_{и}$ – цвет исходного пикселя, $C_{а}$ – цвет актёра, $C_{ф}$ – цвет фона, то можно представить каждый пиксель входного изображения в виде:

$$C_{и} = C_{а} * \alpha + C_{ф} * (1 - \alpha)$$

Для нахождения α находим расстояние L от цвета фона до цвета текущего пикселя (длина вектора в кубе YUV). Используем два параметра r_0 и r_1 (радиусы), где $0 < r_0 < r_1$ и $r < r_0$ задает область фона, а $r > r_1$ – область актёра. Параметры задаем вручную либо автоматически так, чтобы для цветов, не принадлежащих ни актёру, ни фону, L лежало в пределах от r_0 до r_1 . В итоге вычисляем α :

$$\alpha = \min\left(1, \max\left(0, \frac{L - r_0}{r_1 - r_0}\right)\right)$$

Тогда цвет актёра (без фона) вычисляется по формуле:

$$C_{р} = (C_{и} - C_{ф} * (1 - \alpha)) / \alpha$$

А цвет на этапе визуализации:

$$C_{вых} = (C_{и} - C_{ф} * (1 - \alpha)) + C_{сцена} * (1 - \alpha)$$

Для улучшения качества к α -маске применяются фильтры обработки изображения (выделение границ, размытие), а также для цвета применяется цветокоррекция.

В совокупности эти методы, реализованные для графического процессора, позволяют в реальном времени совмещать

изображения реальных объектов с виртуальным без видимых дефектов в различных съёмочных условиях (рис. 4).



Рис. 4: Отображение актёра на произвольном фоне

5. ВЫВОД ИЗОБРАЖЕНИЯ

В разработанной системе реализована поддержка вывода результирующего видеопотока на внешние устройства, вещание в сеть, запись на диск, а также отображение на мониторы (выход видеокарты). Для обеспечения вертикальной синхронизации кадры буферизируются и переключаются в отдельных потоках исполнения. При этом между потоками передаются только идентификаторы на ресурсы графической видеокарты (используются средства видеодрайвера для разделения видеопамати между потоками). В качестве устройств для организации вещания поддерживаются платы SoftLab-NSK (семейство плат FD, http://www.softlab-nsk.com/rus/forward/hardware_all.html).

Различные форматы имеют разное разрешение и соотношение сторон пиксела (аспект). Растеризуемое изображение может иметь фиксированный размер и тогда к нему применяются корректирующие методы (вписывание, обрезка, масштабирование). Так, при необходимости, к изображению могут быть добавлены чёрные поля. Например, при несовпадении разрешений (либо аспектов) видеовыхода (монитор, плата и т.д.) с окном превью, изображение в окне «превью» автоматически масштабируется и обрамляется чёрными полями.

6. ЗАКЛЮЧЕНИЕ

В ходе разработки системы выделены ключевые особенности презентационных систем реального времени. Разработаны и реализованы решения для соответствующих вопросов, и, как результат, разработано программно-аппаратное решение для создания презентаций различной сложности. Применяемая архитектура системы позволяет расширять и дополнять ее новыми функциональными возможностями, необходимыми широкому кругу пользователей.

Сферой применения комплекса являются: всевозможные презентации для выставок, лекционных залов, ситуационных центров, музеев, обучение персонала управлению сложными технологическими процессами и техническими системами, дистанционное обучение и телеконференции.

7. БИБЛИОГРАФИЯ

- [1] Ванданов В.Г. и др. Программно-аппаратный комплекс 3d презентаций на основе виртуальной студии и виртуального окружения // Труды конференции «Трёхмерная визуализация научной, технической и социальной реальности. Кластерные технологии моделирования». 2009. Том 1. С. 73-77.
- [2] Городилов М.А. и др. Управление конфигурациями конвейера обработки изображений и спецэффектов систем визуализации реального времени // Сборник трудов «Технологии Microsoft в теории и практике программирования». 2011. – С. 85-87.
- [3] Долговесов Б.С. и др. Система обучения и презентаций с использованием технологии «погружения» лектора в интерактивную виртуальную среду // Сборник трудов «Online EDUCA 2007, Moscow». 2007.
- [4] Долговесов Б.С. и др. Объектно-ориентированная база данных в интерактивных системах 3D визуализации // Вестник НГУ. Серия: Физика (том 6, выпуск 3) . 2011. – С.59-63
- [5] Ковальков М.А. и др. Разработка и реализация алгоритмов рирпроекции на базе современного графического акселератора // Труды конференции Графикон-2006. 2006. – С. 360-362
- [6] Akenine-Moeller T. et. Real-time rendering, 3ed, 2008.
- [7] Balakhnin A., Port of Yet Another DeInterlacing Filter, // AviSynth's Documentation. 2007 (<http://avisynth.org.ru/>)

Integrated solution for interactive multimedia presentations

Abstract

This paper presents features of the modern hardware/software integrated solution for interactive presentations. Developed presentation system provides wide opportunities for training and demonstration of various multimedia data. Also the paper describes implementation details and presents useful techniques for providing an interactive integration both images of a lecturer and media (video, presentations, text) into virtual scene. Integrated multimedia content improves the quality of presentations.

Keywords: virtual presenter, multimedia, real-time rendering.

About the author

Boris S. Mazurok is a scientific researcher of Synthesizing Visualization Systems Laboratory at Institute of Automation and Electrometry SB RAS. His contact email is boris@albatros.iae.nsk.su

Boris S. Dolgovesov (Ph.D.) is a head of Synthesizing Visualization Systems Laboratory at Institute of Automation and Electrometry SB RAS. His contact email is bsd@iae.nsk.su

Evgeny I. Korostelev is a Ph.D. student at Institute of Automation and Electrometry SB RAS. His contact email is kore3d@gmail.com

Timur N. Artikov is a student of Faculty of Information Technologies at Novosibirsk State University, an engineer-programmer of Synthesizing Visualization Systems Laboratory at

Institute of Automation and Electrometry SB RAS. His contact email is t.artikov@gmail.com

Artur N. Artikov is a student of Faculty of Information Technologies at Novosibirsk State University, an engineer-programmer of Synthesizing Visualization Systems Laboratory at Institute of Automation and Electrometry SB RAS.. His contact email is a_artikov@mail.ru