

**ВОСПРОИЗВЕДЕНИЕ ТРЕХМЕРНОГО ВИДЕО НА РАЗЛИЧНЫХ ПРОГРАММНО-АППАРАТНЫХ ПЛАТФОРМАХ****А.А. Цыганов**Самарский государственный технический университет,  
443100, Самара, ул. Молодогвардейская, 244.

E-mail: hitrolisk@gmail.com

*При записи и воспроизведении трехмерного видео основной проблемой является преобразование записанной информации к универсальному формату, совместимому со всем спектром целевых платформ. В статье рассматриваются способы решения проблемы, применяемые в существующих программных продуктах, и предлагается метод, основанный на определении техник и графических эффектов, используемых в записываемом приложении.*

**Ключевые слова:** *трехмерное видео, совместимость, видеоадаптер, преобразование, высокоуровневый формат*

Создание видеозаписей работы с различными приложениями является достаточно распространенной задачей. Примером служит запись обучающего видео работы с трехмерными редакторами и проектировочными программами, создание роликов из компьютерных игр, а также удаленный терминальный доступ к приложениям, при котором на удаленном компьютере воспроизводится изображение, захватываемое на управляемой машине.

В зависимости от записываемого приложения может отличаться техническая реализация захвата изображения, но ее продуктом в большинстве случаев является двухмерная видеозапись. Разработка технологии записи и воспроизведения трехмерного видео из программ, использующих трехмерную графику, позволяет расширить возможности обычного видео. Такая форма записи имеет ряд преимуществ, обусловленных векторным форматом хранимой информации. Это проявляется в лучшем качестве изображения, произвольном масштабировании без значительной потери качества, а также возможности обработки и трансформации видеопотока. Например, становится возможным выборочно убирать различные объекты, изменять способ освещения трехмерной сцены, менять графические параметры визуализируемых поверхностей или накладывать дополнительные эффекты.

Архитектура системы записи и воспроизведения трехмерного видео включает в себя компоненты операционной системы, записываемое приложение, программное обеспечение захвата трехмерного видеопотока и аппаратные средства, как представлено на рисунке. Приложение, оперируя объектной моделью трехмерной сцены, формирует набор управляющих воздействий в виде ресурсов и команд к графическому программному интерфейсу (API). Этот поток команд перехватывается специальной библиотекой – компонентом системы записи. Он перенаправляется в файл, а также передается программному интерфейсу, который отправляет их видеоадаптеру.

У конечных пользователей могут отличаться многие части описанной системы, начиная от настроек перехватываемого приложения и заканчивая различным аппа-

---

*Цыганов Александр Анатольевич – ассистент кафедры «Вычислительная техника», аспирант.*

ратным обеспечением. Это может привести к невозможности проиграть ролик, записанный на другом компьютере, или к появлению отклонений от исходного изображения. Также ввиду различий в вычислительной мощности разных видеокарт видеоролик может быть замедленным или, наоборот, слишком быстрым. Различия в реализации эффектов и графических техник на разных видеокартах также приводят к визуальным отклонениям от оригинального изображения.



Архитектура системы

В современных трехмерных играх и симуляторах используется множество сложных визуальных эффектов, которые реализуются как с помощью фиксированных функций графических адаптеров, так и с использованием программ для графического процессора, называемых шейдерами [1]. Обычно настройки приложения позволяют снизить качество графики и, соответственно, требования к видеокарте. Но в случае, если видео записано с приложения при высоких настройках, оно может не воспроизводиться на других компьютерах, где исходное приложение может работать только на низких настройках.

Исследование описанных проблем осуществляется путем сбора информации через обратную связь с пользователями во время тестирования программного продукта. При невозможности воспроизведения из-за несовместимости система составляет отчет с технической информацией о компьютере, проигрываемом ролике, приложении, наборе команд и состоянии виртуальной машины графического конвейера. Кроме отчетов об ошибках система осуществляет сбор информации об оборудовании и применяемом программном обеспечении. На основе статистики по этим данным формируется круг используемых пользователями приложений и оборудования. Это необходимо для определения путей решения проблемы совместимости.

Существует ряд технологий записи видео из трехмерных приложений и ряд решений проблемы совместимости. Программы захвата видео из игр, например D3Dgear и Fraps, получают финальное двухмерное изображение, формируемое видеокартой, то есть записывают обычный двумерный видеопоток. В утилитах анализа производительности трехмерных приложений, таких как PIX for Windows и NVIDIA

PerfHUD [2], сессия работы, записанная на одном компьютере, воспроизводится на другом без каких-либо изменений. Если часть записанных команд несовместима с используемой платформой, то воспроизведение не может быть осуществлено. Это решение обуславливается тем, что подобные программы предназначены для разработчиков и не используются для распространения сделанных записей. В приложениях терминального доступа основная задача заключается в удаленном воспроизведении информации, записываемой на другой машине. Для обеспечения совместимости на обеих сторонах используется заранее обговоренный набор команд, с гарантией поддерживаемый каждой стороной.

Подобная методика может быть применена и к трехмерному видео. Суть заключается в том, чтобы привести получаемый от приложения набор графических данных и команд к стандартизированному виду. Во-первых, это касается различий в API [3]. Формат файла записи должен содержать набор команд, представляющих собой надмножество всех используемых программных интерфейсов. Но кроме этого при воспроизведении такого файла нужно преобразовать записанные данные к виду, совместимому с используемой видеокарткой и графическим API.

Большинство приложений оперирует высокоуровневыми понятиями, такими как трехмерный объект, материал, графический эффект и тому подобными [4]. При перехвате команд графического API получаемые данные являются результатом преобразования этих высокоуровневых сущностей и параметров в набор конкретных низкоуровневых управляющих воздействий. Обратное преобразование низкоуровневой информации в высокоуровневую позволяет записывать в файл данные в универсальном формате, совместимом с любым целевым оборудованием. Таким образом, решение проблемы совместимости заключается в распознавании этих высокоуровневых сущностей.

Большинство применяемых в трехмерной графике техник и эффектов можно определить по набору используемых ими команд графического API. Для их уверенного определения необходимо опираться на присутствие характерных команд и ресурсов, их параметры и последовательность использования. Однако технологии трехмерной графики реального времени постоянно развиваются, и необходимо обеспечивать поддержку современных приложений и используемых ими приемов трехмерной графики реального времени.

Для сопоставления различных наборов команд одной высокоуровневой сущности наиболее удобным способом может стать обучение системы на тестовых запусках приложения с различными настройками. Обучение должно производиться с использованием характерных целевых приложений, таких как распространенные трехмерные редакторы, игры и системы проектирования. Кроме того, важны гибкость и простота создания новых правил преобразования информации в высокоуровневый формат. Это позволит добиться совместимости с большинством распространенных платформ и приложений.

### **Заключение**

Решение проблем совместимости трехмерного видео позволит сделать его таким же доступным, как и видео двумерное. Областью его применения является создание обучающих записей работы с трехмерными проектировочными и дизайнерскими программами, обмен роликами между пользователями трехмерных игр, удаленный терминальный доступ к приложениям. Дальнейшее исследование работы системы с

различными программными продуктами позволит добиться лучшего соответствия оригинального изображения и воспроизводимого видео.

#### БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *David Kirk, Randima Fernando*. GPU Gems: Programming Techniques, Tips and Tricks for Real-Time Graphics. – Pearson, 2004. – 816 p.
2. NVIDIA Developer Zone / NVIDIA company developer web site. URL: <http://developer.nvidia.com>
3. *А. Евченко*. OpenGL и DirectX: программирование графики. – СПб.: Питер, 2006. – 352 с.
4. *Эндрю Роллингз, Дэйв Моррис*. Проектирование и архитектура компьютерных игр. – М.: Вильямс. – 2005. – 1040 с.

*Статья поступила в редакцию 16 июня 2009 г.*

UDC 681.3.05

### THREE-DIMENSIONAL VIDEO PLAYBACK ON DIFFERENT HARDWARE AND SOFTWARE PLATFORMS

*A.A. Tsyganov*

Samara State Technical University,  
244, Molodogvardeyskaya str., Samara, 443100.

*The main problem during three-dimensional video recording and playback is conversion of recorded information into universal format compatible with the entire range of target platforms. This article observes problem solutions used in existing software products, and proposes method based on detection of techniques and graphic effects used in recording application.*

**Keywords:** *three-dimensional video, compatibility, video adapter, conversion, high-level format.*