

УДК 004.946

ОТСЛЕЖИВАНИЕ И РЕНДЕРИНГ ИЗОБРАЖЕНИЙ В КОНТЕКСТЕ ДОПОЛНЕННОЙ РЕАЛЬНОСТИ С ИСПОЛЬЗОВАНИЕМ СМАРТФОНА

Маслов А.С., Козина А.В., Белов Ю.С.

*Московский государственный технический университет имени Н.Э. Баумана, филиал, Калуга,
e-mail: maslow.tema@yandex.ru*

Дополненная реальность (AR) – это концепция, которая была введена в 1960-е гг., но с тех пор никаких крупных приложений AR не появилось. Это было в основном из-за аппаратных и программных требований AR, но ситуация изменилась в недавнем прошлом, с успехом смартфонов. Современные смартфоны оснащены всем необходимым оборудованием, а также достаточной вычислительной производительностью для отслеживания маркеров и визуализации объектов для дополненной реальности. В настоящее время существует несколько различных видов отслеживания маркеров для достижения эффекта дополненной реальности. В данной статье рассмотрены следующие варианты отслеживания маркеров: невизуальное отслеживание, визуальное отслеживание и визуальное отслеживание без маркеров, которое совмещает оба подхода. Рассмотрен процесс рендеринга изображения. Приведен пример технологии, которая позволяет достичь фотореалистичного изображения, а также технологии рендеринга изображений в реальном времени. Сделаны сравнения текущих технологий, которые показывают, что фотореалистичного рендеринга изображений в реальном времени пока что невозможно достичь из-за недостаточной вычислительной мощности, но благодаря развитию Vulkan API эта возможность может появиться в ближайшем будущем.

Ключевые слова: дополненная реальность, смартфон, рендеринг, фотореалистичный рендеринг, отслеживание маркера

TRACKING AND RENDERING IMAGES IN THE CONTEXT OF AUGMENTED REALITY USING A SMARTPHONE

Maslov A.S., Kozina A.V., Belov Yu.S.

Bauman Moscow State Technical University, Kaluga branch, Kaluga: e-mail: maslow.tema@yandex.ru

Augmented reality (AR) is a concept that was introduced in the 60s of the 20th century, but no major AR applications have appeared since. This was mainly due to the hardware and software requirements of AR, but the situation has changed in the recent past, with the success of smartphones. Modern smartphones are equipped with all the necessary equipment, as well as sufficient computing performance to track markers and visualize objects for augmented reality. Currently, there are several different types of tracking markers to achieve the effect of augmented reality. This article discusses the following options for tracking markers: non-visual tracking, visual tracking, and visual tracking without markers, which combines both approaches. The process of image rendering is considered. An example of a technology that allows you to achieve a photorealistic image, as well as a technology for rendering images in real time. The comparison of current technologies shows that the achievement of photorealistic image rendering in real time is not yet possible due to insufficient computing power, but thanks to the development of the Vulkan API, this feature may appear in the near future.

Keywords: augmented reality, smart-phone, rendering, photo-realistic rendering, tracking

Чтобы получить достоверный опыт дополненной реальности, визуализированный дополненный объект должен быть правильно наложен на изображение камеры. Для достижения регистрации изображения используется процесс, называемый отслеживанием [1]. Отслеживание может быть достигнуто несколькими способами, отличающимися входными данными, используемыми для этого.

В данной статье будут приведены различные методы отслеживания, а также методы рендеринга объектов, полученных в результате отслеживания с использованием смартфона.

Материалы и методы исследования

Идеальный алгоритм отслеживания должен поддерживать наложение дополненного объекта таким образом, чтобы объ-

ект не был распознан как дополненный. Визуализируемое изображение должно отображаться точно в нужном месте, даже когда наблюдатель ходит вокруг, чтобы увидеть его под разными углами. Подходы к отслеживанию можно разделить на две группы, в зависимости от использования входного изображения с реальной сцены. Невизуальное отслеживание способно работать без обработки изображения реального мира, в то время как визуальное отслеживание зависит от распознавания паттернов во входном изображении, чтобы перекрыть дополненный объект.

Отслеживание

Невизуальное отслеживание. Отслеживание без визуального ввода должно использовать другие данные для реализации регистрации изображения. Реальный мир,

камера и дополненный объект должны быть проецированы в одну общую систему координат. Знание местоположения и поворота камеры имеет важное значение для определения угла обзора дополненного объекта. Также важно желаемое положение дополненного объекта. Расположение камеры смартфона измеряется многими датчиками. Получить местоположение можно по сети GSM, Wi-Fi или по глобальным системам позиционирования, предоставляемым спутниками из космоса. Последний упомянутый метод является наиболее точным. Настройка местоположения дополненного объекта с невидуальным отслеживанием осуществляется путем установки желаемого местоположения объекта по геологическим координатам реального мира. Таким образом, одна общая система координат для регистрации изображений – это геологические долгота, высота и широта.

Когда задано желаемое положение объекта, необходимо измерить местоположение и поворот камеры (в данном случае камеры мобильного устройства). Для измерения местоположения прибора используются встроенные геолокационные чипы, обычно GPS, ГЛОНАСС и GALILEO [2].

Все эти локационно-измерительные системы работают по одному и тому же принципу. Спутники в космосе, вращающиеся вокруг Земли, передают сигналы с временной информацией, позволяя антенне целевого устройства принимать этот сигнал и вычислять расстояние от спутника. Для фиксации точного расположения в трехмерном пространстве необходим сигнал по меньшей мере от трех спутников. Точность определения местоположения увеличивается с увеличением количества спутников, пример показан на рис. 1.

Из-за расстояния, которое проходит сигнал, точность этого метода обычно составляет менее 3 м. С ухудшением погодных условий или определением в районах с узкой видимостью неба точность становится намного хуже. Однако одновременное использование нескольких навигационных систем для определения местоположения одного предмета улучшает точность. Смартфоны также оснащены антеннами A-GPS (Assisted GPS), когда подключение к интернету используется для загрузки последних сообщений о местоположении спутников, чтобы получить более точные результаты и более быструю блокировку местоположения.

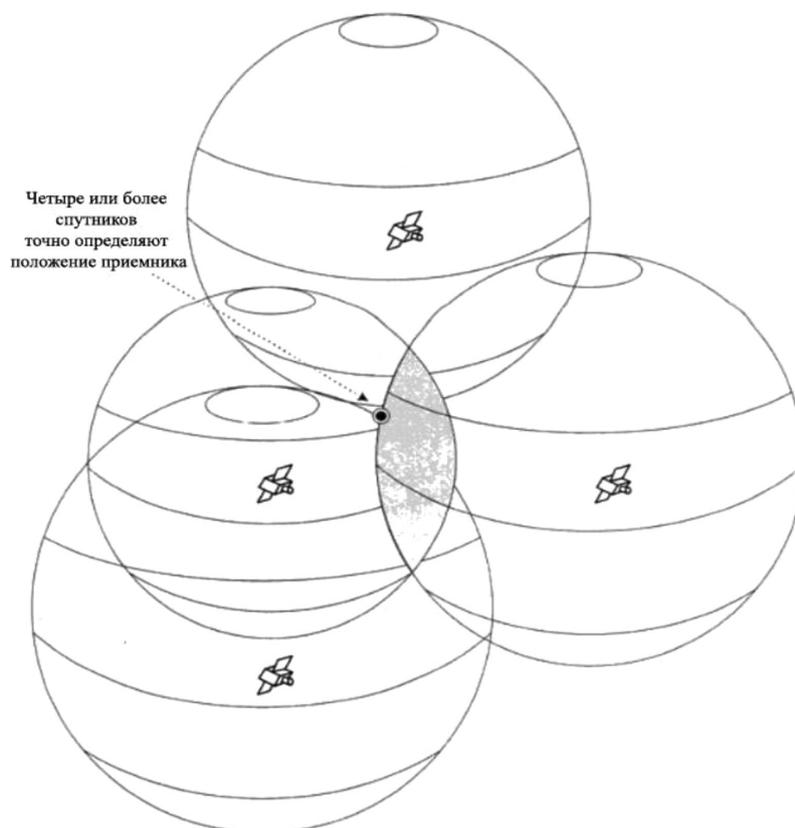


Рис. 1. Определение местоположения с помощью четырех спутников

Когда местоположение устройства определено, необходимо измерить наклон камеры, чтобы правильно сделать регистрацию изображения. Для этого используются три датчика: акселерометр, гироскоп и цифровой компас. Микросхема акселерометра измеряет ускорение устройства по трем осям, предоставляя информацию о незначительных перемещениях устройства. Гироскоп вычисляет наклон прибора, также по трем осям, а цифровой компас определяет относительное вращение прибора от северного магнитного полюса Земли.

Визуальное отслеживание. Визуальные методы отслеживания не используют информацию из других источников. Единственным вводом данных, необходимым для достижения регистрации изображения, является само изображение камеры. Алгоритмы такого типа слежения должны реализовывать распознавание изображений. Для этого используются маркеры [3–4]. Маркеры – это паттерны, обычно объекты, размещенные в реальной сцене, которые используются для определения положения дополненного объекта. Наиболее распространенным случаем является использование QR-кодов, так как они легко распознаются на изображении. Для более точного размещения также возможно использование 3D-маркеров (рис. 2).

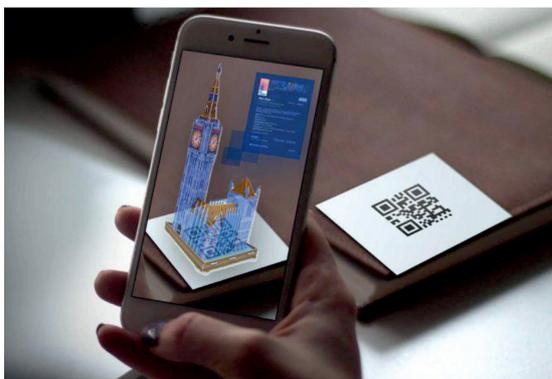


Рис. 2. Визуальное отслеживание с помощью маркера

Визуальное отслеживание без маркеров. Безмаркерный визуальный трекинг – это сочетание визуального и невизуального трекинга, когда входное изображение используется без маркеров, а также данные с датчиков устройства для достижения регистрации изображения. В этом случае вращение камеры смартфона используется как имитация стереоскопического зрения для получения карты глубины сцены, позволяющей распознавать 3D-объекты на реальной сцене [5].

Рендеринг изображения

Рендеринг – это процесс преобразования 3D-моделей и сцен, сохраненных в компьютере, в 2D-изображение, которое впоследствии отображается на устройстве отображения, обычно мониторе [6]. Рендеринг – это наиболее сложный и требовательный к производительности сегмент компьютерной графики, используемый во многих областях (медицина, архитектура, машиностроение, видеомонтаж и др.). Каждая из этих дисциплин требует от визуализируемого изображения различных параметров (качество изображения, скорость рендеринга), что приводит к появлению многих методов рендеринга. Рендеринг в реальном времени – это рендеринг, который способен взаимодействовать с пользователем таким образом, что пользователь не распознает отдельные изображения, доставленные ему в руки с минимальной задержкой. Это требует быстрого прохождения через конвейер рендеринга, что является очень трудоемкой задачей, превышающей возможности производительности процессора. Эта потребность была удовлетворена с помощью специального аппаратного блока, используемого для геометрических вычислений, графического процессора (GPU).

Конвейер рендеринга. Концепция конвейера используется во многих различных формах, где процесс делится на отдельные этапы, чтобы предотвратить запаздывание всей системы [6]. Конвейер рендеринга, также называемый графическим конвейером или графическим конвейером рендеринга, разделен на три концептуальных этапа для увеличения скорости всего рендеринга (рис. 3):

- Приложение.
- Геометрия.
- Растеризация.

Каждая стадия может быть разделена на несколько стадий, что означает, что каждая стадия также является конвейером. Некоторые части конвейера могут быть распараллелены, чтобы соответствовать высоким требованиям к производительности. Скорость конвейера рендеринга измеряется в fps (кадрах в секунду) и определяется самым медленным этапом. Эта скорость обычно изменяется в зависимости от сложности вычисляемой сцены, которая меняется кадр за кадром, поскольку пользователь взаимодействует с расположением. Этап применения конвейера управляется программным обеспечением центрального процессора, позволяющим осуществлять параллельную обработку благодаря множеству потоков центрального процессора.

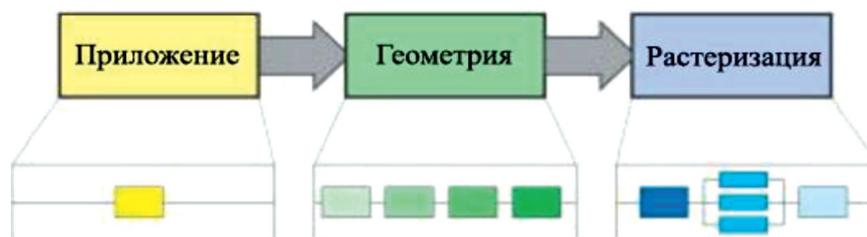


Рис. 3. Стадии конвейера рендеринга

```

for each pixel do
  compute viewing ray
  if (ray hits an object with  $t \in [0, \infty)$  ) then
    Compute lighting equation and set pixel to that color
  else
    Set pixel color to background color

```

Рис. 4. Алгоритм трассировки BasicRay

На этом этапе выполняются различные задачи: обнаружение столкновений, алгоритмы глобального ускорения, анимация, моделирование физики (необязательно) и многие другие, в зависимости от типа приложения. Этап геометрии вычисляет преобразования, проекции и т.д. Определение того, что, как и где должно быть нарисовано, выполняется, как правило, на графическом процессоре (GPU). Последний этап, этап растеризации, полностью обрабатывается на графическом процессоре, визуализируя данные, выводимые с предыдущего этапа, а также любые необходимые вычисления на пиксель. Использование специализированного оборудования для вычислительной графики необходимо для достижения плавного рендеринга в реальном времени без задержек, что является фундаментальным также для приложений дополненной реальности. К счастью, все современные смартфоны и планшеты оснащены мощными графическими процессорами, что позволяет использовать дополненную реальность на мобильных устройствах.

Фотореалистичный рендеринг. Для достижения правдоподобного опыта дополненной реальности дополненный объект должен быть представлен в наилучшем доступном качестве и реалистичности. Для этого используются фотореалистичные методы рендеринга. Фотореалистичный рендеринг, также называемый непредвзятым рендерингом, – это метод, который физически точно визуализирует сцену, обычно отслеживая свет с помощью алгоритмов

на пиксель. Непредвзятые методы рендеринга включают в себя:

- Трассировка пути.
- Трассировка света.
- Двухнаправленная трассировка пути.

Результаты исследования и их обсуждение

Одним из наиболее широко используемых методов фотореалистичного рендеринга является трассировка лучей. Трассировка лучей – это алгоритм для каждого пикселя, который использует метод трассировки пути, отбрасывая лучи от камеры в пространство, чтобы определить цвет пикселя. Он также используется в Z-буферных реализациях визуализаторов для достижения обнаружения столкновений объектов и щелчка мыши по их выбору. Щелчок мыши привязывается к одному пикселю, из которого луч отбрасывается в пространство, выбирая первый попавшийся объект. При рендеринге трассировка лучей проста в вычислении теней и отражений. Алгоритм трассировки BasicRay представлен на рис. 4.

В этом примере цвет пикселя вычисляется только в соответствии с материалом и применением нетрассирующей световой модели (т.е. модели Phong [7]) к пикселю поверхности объекта. В результате качество изображения сравнимо с предвзятыми визуализаторами. Полный механизм трассировки лучей вычисляет освещение сцены лучами. После того как луч попадает на объект, он отражает и отслеживает луч итеративно,

пока луч не попадает на источник света. Тот же принцип используется и для вычисления теней. Когда луч, идущий из точки на поверхности объекта, попадает на другой объект по пути к источнику света, вычисленная точка оказывается в тени. Эта техника физически точна, потому что она описывает и вычисляет поведение света в реальных сценах. Результирующие изображения находятся в особых условиях, их трудно отличить от фотографий. Nvidia Iray – это фотореалистичный движок рендеринга, который генерирует фотореалистичные изображения, имитируя физическое поведение света и материалов (рис. 5) [8].

является Vulkan API. Vulkan – это низкоуровневый API для графического и вычислительного оборудования. Это набор команд, которые позволяют определять спецификацию шейдеров, ядер и используемых ими данных [9]. С другой точки зрения, Vulkan – это конвейер рендеринга, который имеет некоторые программируемые этапы и некоторые управляемые состоянием этапы фиксированной функции, которые вызываются конкретными операциями рисования. Эти настройки позволяют намного больше контролировать графическое вычислительное оборудование, позволяя им писать более оптимизированный код.

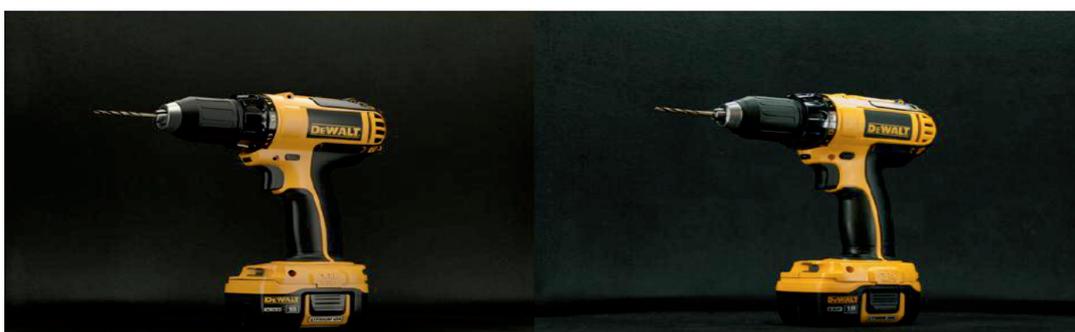


Рис. 5. Слева модель, справа её визуализация с помощью Nvidia Iray

Требования к производительности. Движок рендеринга Nvidia Iray, упомянутый выше, способен работать только на настольных компьютерах с мощной видеокартой, да и то не в режиме реального времени. Трассировка лучей – это алгоритм для каждого пикселя, поэтому его вычислительные требования быстро возрастают с более высокими разрешениями. На мобильных устройствах до сих пор не удается добиться такого качества рендеринга изображения. Рендеринг в реальном времени также необходим для применения дополненной реальности, а также как резерв вычислительной мощности для обработки изображения камеры и датчиков устройства. До сих пор нет никаких шансов для обычных устройств выполнять рендеринг фотореалистичных сцен в реальном времени, направленных на создание стилизованной реальности, но их производительность сходится, чтобы удовлетворить этим требованиям.

Vulkan API. Аппаратное обеспечение – это не единственная область, которая переживает значительный прогресс. Наряду с аппаратным обеспечением, программные API и драйверы графических процессоров также претерпели значительные улучшения. Одним из самых важных изменений

Использование API Vulkan уменьшает количество вызовов рисования, отправляемых на центральный процессор, что приводит к значительному снижению накладных расходов центрального процессора. Это решение обеспечивает повышение производительности на всех существующих платформах, даже на мобильных устройствах. Также поддерживаются устройства Android, но только с аппаратной поддержкой OpenGL ES 3.1 и более поздних версий. Vulkan API все еще относительно новый, поэтому реализовано всего несколько приложений, а на мобильных устройствах ситуация еще хуже. Однако бенчмарковые тесты показывают огромный потенциал этого API, и в ближайшем будущем мы определенно увидим гораздо больше приложений, использующих этот API. Основываясь на последних бенчмарках, повышение производительности API Vulkan может достигать более 80% по сравнению с решениями OpenGL [10].

Заключение

Существуют различные методы отслеживания, и каждый метод применяется под свои конкретные задачи. Для задач с геопозиционированием хорошо подходит

метод, основанный на невизуальном отслеживании, для задач с выводом информации об объекте хорошо подойдет метод, основанный на визуальном отслеживании. Рендеринга объектов, полученных в результате отслеживания с фотореалистичными изображениями, пока что не удаётся достичь с текущими технологиями, но Vulkan API даёт надежды, что в скором времени этого будет возможно.

Список литературы

1. Erkan Bostanci S.E., Nadia Kanwal, Clark A.F. User Tracking Methods for Augmented Reality. *International Journal of Computer Theory and Engineering*. 2013. Vol. 5. No. 1. P. 93–98.
2. Rao G.S. *Global Navigation Satellite Systems: With Essentials of Satellite Communications*. New Delhi: Tata McGraw Hill, 2010. 470 p.
3. Маслов А.С., Белов Ю.С. Различные виды мишеней и их распознавание в фреймворке дополненной реальности Vuforia // Научное обозрение. Технические науки. 2020. № 1. С. 15–19.
4. Klein G. *Visual Tracking for Augmented Reality*. PhD. Thesis. University of Cambridge, 2006. 193 p.
5. Andrew I. Comport, I. E. M. M. I. M. P., Member; François Chaumette, I., Member: Real-Time Markerless Tracking for Augmented Reality: The Virtual Visual Servoing Framework. *IEEE TRANSACTIONS ON VISUALIZATION AND COMPUTER GRAPHICS*. 2006. Vol. 12. No. 4. P. 615–628.
6. Akenine-Moller T. Haines E. *Real-Time Rendering*, Fourth Edition. CRC Press, 2018. 158 p.
7. Wikipedia: Phong reflection model. [Electronic resource]. URL: https://en.wikipedia.org/wiki/Phong_reflection_model (date of access: 16.11.2020).
8. Nvidia: NVIDIA IRAY. [Electronic resource]. URL: <http://www.nvidia.com/object/nvidia-iray.html> (date of access: 16.11.2020).
9. KHRONOS: Vulkan 1.0.17 – A Specification. [Electronic resource]. URL: <https://www.khronos.org/registry/vulkan/specs/1.0/pdf/vkspec.pdf> (date of access: 16.11.2020).
10. Pirazda U. Khronos Group's 'Vulkan' API, Performance Numbers Revealed. [Electronic resource]. URL: <http://wccftech.com/khronos-vulkan-api-performance-numbers-89-star-dust-benchmark/> (date of access: 16.11.2020).