УДК 004.08

К.А. Ручкин, Е.А. Шевченко, В.А. Акчурин
Институт информатики и искусственного интеллекта, ДонНТУ

РАЗРАБОТКА ИНТЕРАКТИВНОГО КЛИЕНТ-СЕРВЕРНОГО ПРИЛОЖЕНИЯ РАСШИРЕННОЙ РЕАЛЬНОСТИ

Розглянуто проблему інтерактивних додатків розширеної реальності. Проаналізовано додатки розширеної реальності у різних сферах людської активності. Наведена архітектура клієнтського та серверного додатків багатозадачної платформи на основі клієнт-серверної технології.

Рассмотрена проблема многозадачных платформ расшренной реальности. Проанализированы приложения расширенной реальности в различных сферах человеческой деятельности. Приведена архитектура клиентского и серверного приложений многозадачной платформы на основе клиент-серверной технологи.

In this work the problem of interactive applications of augmented reality is investigated. Augmented reality applications in various fields of human activity are being analyzed. Architecture of client and server applications of multitask platform based on client-server technology is represented as result of research.

Ключевые слова: расширенная реальность, интерактивность, маркер, манипулятор, платформа, распознавание.

Введение. Одним из перспективных направлений в программном обеспечении (ПО) на сегодняшний день являются естественные интерфейсы. Цель данного направления – разработка интерфейсов приложений, отличных от стандартных двухмерных интерфейсов. К ним относятся речевые интерфейсы, виртуальная реальность, а также расширенная реальность.

Расширенной реальностью является наложение дополнительных ощущений (чаще всего зрительных) на естественное окружение. Для этого используются системы распознавания изображений и графические средства для добавления дополнительного графического контента. Контент может быть как статическим, так и динамическим, интерактивным.

Архитектура приложения расширенной реальности (рис. 1) имеет следующий вид. В систему распознавания поступает видео с камеры, в котором присутствуют предварительно установленные изображения – маркеры. Маркерами могут быть двухмерные черно-белые изображения, цветные изображения или окружающие объекты. Система распознавания находит позицию маркеров относительно камеры и передает в графическую библиотеку, где видео, на основе позиций маркеров, дополняется графическим контентом и выводится на экран.

Таким образом, с точки зрения распознавания имеется, с одной стороны, система распознавания, обладающая алгоритмами, работающими с некоторой точностью, а с другой стороны – система маркеров, выделяющихся на фоне окружающей среды.

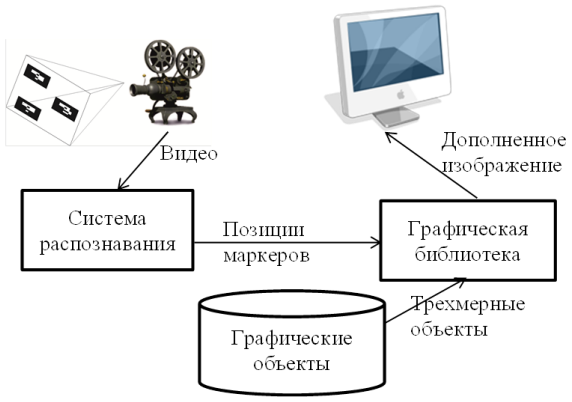


Рис. 1. Архитектура приложений расширенной реальности

По данным компании Adventure Design, расширенная реальность активно используется в таких сферах человеческй деятельности, как образование, реклама, презентации, компьютерные игры.

В последние десятилетия было проведено много исследований, посвященных изучению эффекта 3D компьютерных технологий в сфере обучения с положительными и обнадеживающими результатами. Есть уже много AR систем с функциями, которые допускают двустороннее взаимодействие между пользователем и системой, разработанных для образовательных целей [5]. Примером такой системы является проект

© *К.А. Ручкин, Е.А. Шевченко, В.А. Акчурин, 2013*

«Магическая книга солнечной системы», созданный Недимом Слайджепчевичем и Ванджу Хавнгом. Он представляет собой книгу, снабженную черно-белыми маркерами. При попадании маркера в поле камеры, на экране появляется планета с ее описанием [5].

Еще одним проектом является “Physics playground”. Он использует последние физические ядра, разработанные для рынка компьютерных игр с целью имитации физических экспериментов. Студенты получают возможность строить собственные эксперименты и изучать их [6].

Примером игровых проектов является расширение виртуальным контентом настольных игр. С помощью простой веб-камеры и методов компьютерного зрения стандартные фишки настольной игры преобразуются в игру с расширенной реальностью. Эти объекты используются как маркеры и дополняются визуальными эффектами [7].

Проекты, представленные выше, как и основная часть проектов расширенной реальности, не предусматривают многопользовательскую, клиент-серверную архитектуру. Однако в этой области также имеются разработки. В рамках проектов ARTag [4] и Studierstube [3] были проведены разработки локальных и удаленных многопользовательских проектов расширенной реальности. Примером таких проектов являются телеконференции с использованием расширенной реальности. С помощью видеокамеры и очков расширенной реальности пользователь может видеть собеседников на маркерах с их именами [2].

Каждый проект имеет свои преимущества, однако общим их недостатком является «жесткое» встраивание системы распознавания в приложение на этапе его создания. Одно приложение предлагает один описанный набор функций, не подлежащий расширению. *Однако при использовании очков расширенной реальности необходимо изменять набор функций, предлагаемых приложением в зависимости от окружения, а также потребностей пользователя, то есть работать с несколькими приложениями расширенной реальности по очереди или одновременно. Таким образом, одной из недостаточно исследованных актуальных задач в данный момент является разработка платформ расширенной реальности, поддерживающих многозадачность.*

Постановка задачи. Целью данной работы является разработка интерактивной многозадачной платформы расширенной реальности. Для обеспечения многозадачности была выбрана клиент-серверная технология, проведено исследование различных инструментов взаимодействия пользователя с приложением расширенной реальности.

Проект расширенного рабочего стола, рассматриваемый в данной статье, предлагает многозадачную платформу, на основе которой могут запускаться различные клиентские приложения расширенной реальности. Она берет на себя все функции распознавания изображений, позиционирования и вывода графической информации на экран. Клиентское приложение реализует только логику, а также операции с графическим контентом (трехмерным интерфейсом программы).

Целью данного исследования является также проектирование и реализация интерактивного приложения расширенной реальности на основе клиент-серверной технологии. Данная технология позволит ему обмениваться информацией с другими приложениями и, таким образом, сделает возможным существование нескольких приложений в одном дополненном пространстве.

Практическим применением данной разработки может быть расширенный рабочий стол – платформа, аналогичная рабочему столу Windows, однако работающая в расширенной реальности. Таким образом, пользователь в расширенной реальности может одновременно работать с текстовыми документами, играть и т.д.

Для реализации клиент-серверной технологии в полной мере (не только локального сервера) проект предусматривает многопользовательскую технологию, а также сетевую прозрачность. В результате с одним рабочим столом могут работать сразу несколько пользователей, в том числе и удаленно.

Метод решения. Разработка многозадачной интерактивной платформы расширенной реальности имеет следующие этапы:

1) исследование различных методов взаимодействия и разработка интерфейса для даной работы;
2) разработка архитектуры платформы;
3) разработка механизмов взаимодействия клиентского и серверного приложений.

Если говорить об обеспечении взаимодействия пользователя с приложением, на наиболее простом уровне это происходит с помощью поворота камеры или маркера. Дополнительное изображение поворачивается вместе с маркером. Однако для более сложных взаимодействий используются дополнительные инструменты.

В качестве дополнительных манипуляторов может использоваться лист с нанесенными управляющими маркерами (рис. 2,а). Некоторые из них закрываются рукой, представляя механизмы интерактивности.

Еще одним средством обеспечения интерактивности являются специальные предметы, манипуляторы с нанесенными маркерами (рис. 2,б). Система распознавания может определить его позицию в расширенном пространстве. Таким образом, можно определить его взаимодействие с виртуальными объектами.

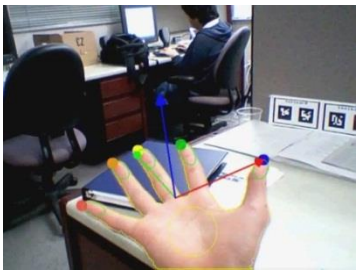
Наиболее сложным инструментом взаимодействия является тело человека, к примеру человеческие руки (рис. 2,в). Данное направление еще разрабатывается, так как стабильное распознавание рук представляет проблему для систем распознавания.



а



б



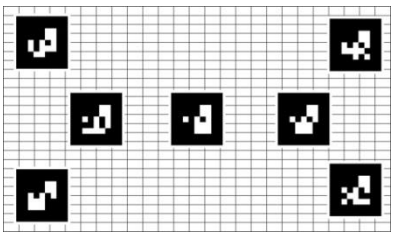
в

Рис. 2. Инструменты взаимодействия пользователя с приложением:
а – закрытие маркеров; б – манипулятор «ручка»; в – распознавание руки

Пользовательский интерфейс взаимодействия состоит из плоского предмета, снабженного маркерами (расширенный рабочий стол), и манипулятора (рис. 3).

Дизайн и разметка расширенного рабочего стола определяются отдельно для особенностей запроса каждого пользователя. Это может быть скатерть с нанесенными на нее маркерами либо лист бумаги разного размера.

Маркеры служат для позиционирования рабочего стола в пространстве, а разметка прямыми линиями – для формирования переднего плана изображения. Расширенный рабочий стол задает систему координат, в которой рисуются трехмерные объекты. К объектам, если это необходимо, привязываются физические сущности, которые после этого регистрируются в физическом мире и в серверном приложении. В физическом мире обрабатываются их взаимодействия, а серверное приложение посылает сообщение клиентскому о взаимодействии манипулятора и объекта приложения. Манипулятор основан на разработке Studierstube.



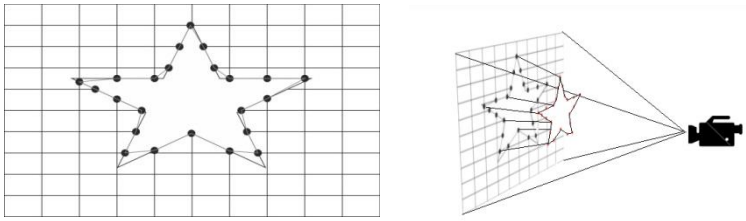
а



б

Рис. 3. Элементы взаимодействия:
а – расширенный рабочий стол; б – манипулятор

Проблема перекрытия расширенной трехмерной сцены различными предметами решается разработками расширенной реальности по-разному. В проекте Studierstube трехмерная сцена делается полупрозрачной, таким образом показывая руки пользователя. В технологии YVision разработано глубинное компьютерное видение. В данной разработке предлагается формирование переднего плана изображения (рис. 4). Оболочка чужеродного элемента, попавшего на фон расширенного рабочего стола, «обрисовывается» с помощью поиска точек разрыва линий разметки (рис. 4,а). После этого из камеры направляются лучи к точкам разрыва и вычисляются точки пересечения лучей с плоскостью, параллельной плоскости рабочего стола, но находящейся ближе к камере (рис. 4,б).



а б

Рис. 4. Формирование переднего плана:
а – выделение оболочки; б – перенос оболочки

Архитектура платформы (рис. 5) состоит из серверного и набора клиентских приложений, обменивающихся информацией.

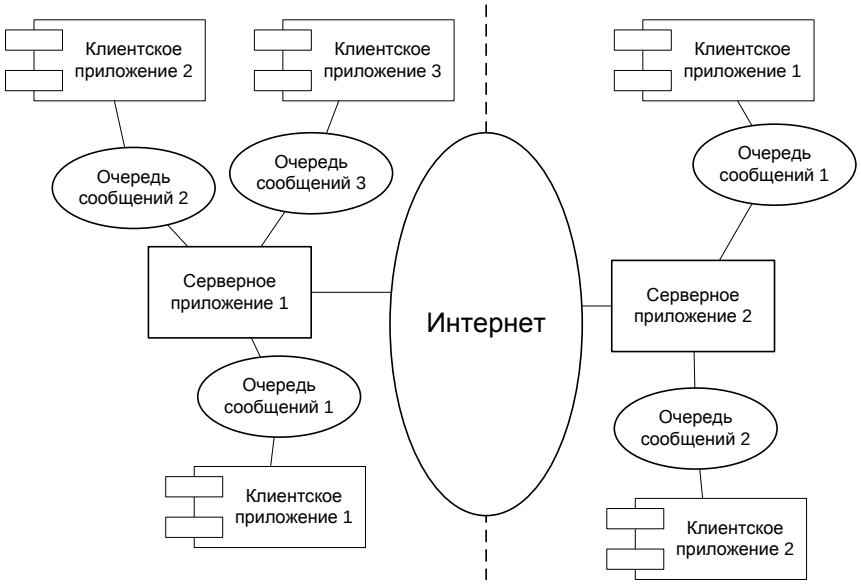


Рис. 5. Архитектура платформы расширенного рабочего стола

Серверное приложение выполняет следующий набор функций:

1) запуск и закрытие клиентских приложений;
2) позиционирование глобальной системы координат в соответствии с набором маркеров расширенной реальности;
3) позиционирование манипулятора;
4) обработка взаимодействий манипулятора с объектами расширенной реальности;
5) обмен информацией о положении и взаимодействиях манипулято-ра, нажатиях клавиш на клавиатуре и т.д. с клиентскими приложениями;
6) вывод на экран пространства расширенной реальности и всех его объектов.

Каждое клиентское приложение, кроме реализации своей своей логики, выполняет следующий набор функций:

1) создание и преобразование трехмерных объектов своей трехмерной сцены;
2) обработка информации, поступающей от серверного приложения.

Исходя из функциональности платформы, можно заметить, что клиентское и серверное приложения должны обмениваться большими объемами информации о трехмерных объектах. Так как серверное приложение также имеет большую вычислительную сложность, связанную с распознаванием маркеров, желательно избежать копирования этой информации из участка памяти клиентского приложения в серверное или сделать его максимально быстрым.

На основании этих требований было решено использовать библиотеки динамической линковки в качестве клиентских приложений.

При использовании данного метода клиентские приложения будут компилироваться в библиотеки динамической линковки [1] (dll для ОС Windows) и динамически загружаться и выгружаться серверной программой. Данный подход не требует копирования или какого-либо другого обновления разделяемого графического контента, является кроссплатформенным. Минусом является некоторая незащищенность сервера от вредоносных программ. Данный подход был выбран как наиболее подходящий и удобный.

Каждая клиентская программа компилируется в библиотеку динамической линковки, имеющую точку входа – функцию “mainFn”, запускаемую серверным приложением в отдельном потоке. В качестве параметров данная функция должна принимать объект, хранящий в себе данные, необходимые для клиент-

серверного взаимодействия, и представляющий функциональность обмена информацией между клиентским и серверным приложениями.

Объект должен содержать следующий набор данных:

1) идентификатор (хендл) загруженного приложения;
2) путь к приложению;
3) идентификатор потока, в котором выполняется загруженное прило-жение;
4) очередь сообщений приложения;
5) физический мир, обрабатывающий взаимодействие манипулятора и виртуальных объектов, создаваемых приложением;
6) указатель на локальную систему координат приложения.

Из схемы взаимодействия серверного и клиентского приложений (рис. 6) видно, что серверное приложение обменивается с клиентским сообщениями с помощью очереди сообщений, аналогичной очереди Windows с некоторыми исключениями. Каждое приложение имеет свою очередь сообщений, и каждое сообщение имеет два параметра: идентификатор вида сообщения и указатель на класс, содержащий дополнительные параметры сообщения. Каждый вид сообщения имеет свой набор пара-метров.

Серверное приложение состоит из трех основных модулей: разделяемый мир расширенной реальности, очередь сообщений серверного приложения и ядро. Клиентское приложение также имеет свою очередь сообщений и ядро, где реализована его логика. Серверное приложение при запуске клиентского приложения создает локальный мир для данного приложения, очередь сообщений для него, а после этого запускает само приложение.

При инициализации клиентское приложение вносит изменения в разделяемый трехмерный мир, с которым взаимодействует напрямую, а именно в свою локальную трехмерную сцену. Клиентское приложение также имеет основной цикл, в котором происходит считывание сообщений из очереди и обработка их.

Одним из вариантов завершения работы программы является посылка сообщения M_CLOSE в соответствующую очередь. При закрытии клиентское приложение удаляет динамические объекты, свои трехмерные объекты из разделяемого мира, освобождает ресурсы и т.д. После этого посылается сообщение M_POSTCLOSE в очередь сообщений сервера, по которому серверное приложение очищает память, занятую клиентским приложением, удаляет соответствующую очередь сообщений и локальную трехмерную сцену.

Взаимодействие по сети происходит между двумя серверными приложениями. Так как передача по сети всей трехмерной сцены является очень затратным процессом, между двумя серверными приложениями передаются только сообщения в очередях сообщений. Все перемещения объектов происходят локально.

Были спроектированы следующие виды сообщений и следующие наборы параметров:

1) закрытие программы (M_CLOSE);
2) активация программы (M_ACTIVATE);
3) программа становится неактивной (M_DEACTIVATE);
4) перемещение манипулятора (M_MAN_MOVE). Параметром являет-ся матрица локальной системы координат манипулятора;
5) взаимодействие манипулятора с виртуальным объектом (M_COLLISION). Параметрами являются: матрица локальной системы координат манипулятора; ссылка на виртуальный объект взаимодействия; массив точек взаимодействия;
6) нажатие клавиши на клавиатуре (M_KEYPRESSED). Параметром является нажатый символ.

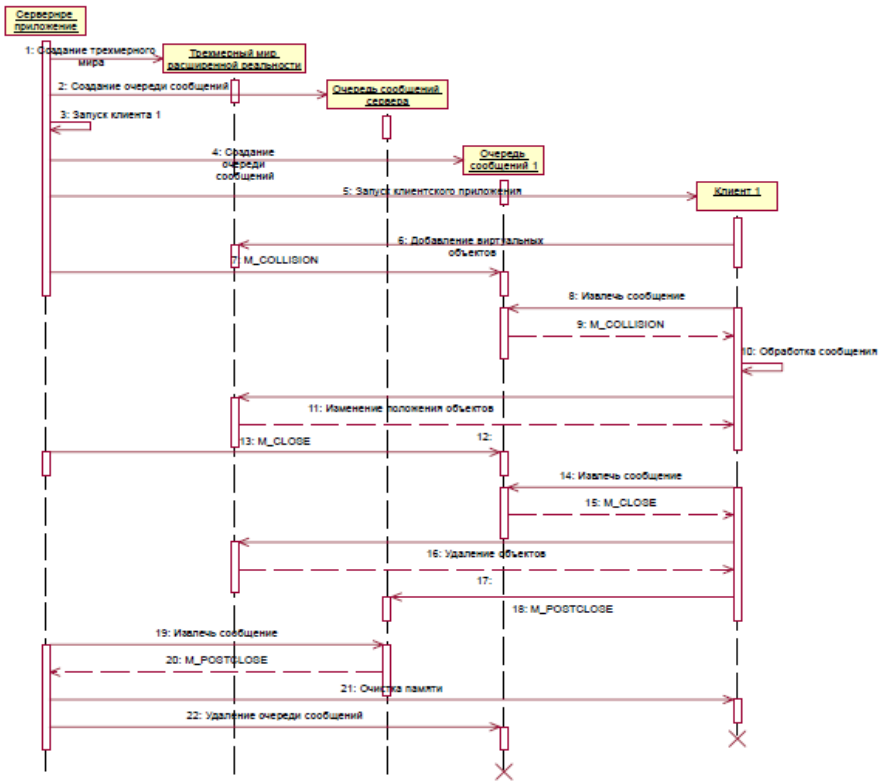


Рис. 6. Схема взаимодействия клиентского и серверного приложений

Анализ результатов. На основе схемы платформы реализовано тестовое серверное приложение и клиентское, которое изменяет цвет кубика в зависимости от положения манипулятора (рис. 7).

Приложение было протестировано несколькими пользователями: им было предложено опробовать стабильность обнаружения коллизии, распознавания манипуляторов и стола. В итоге тестирования были обнаружены недочеты программного обеспечения.

Преимуществами интерфейса является легкость в его использовании, естественная форма манипулятора. Недостатки связаны с несовершенством алгоритмов распознавания видеопотока. Решением этой проблемы является реализация платформы на основе более точных систем распознавания, к примеру библиотеки SmartAR, анонсированной компанией Sony. Данная платформа будет способна распознавать даже быстро движущиеся маркеры.



а б

Рис. 7. Демонстрация работы тестового приложения:
а – без коллизии; б – с коллизией

Выводы. В результате исследования была разработана многозадачная платформа для интерактивных приложений расширенной реальности, обладающая набором достоинств по сравнению с аналогами:

1) возможность различных разработчиков создавать приложения рас-ширенной реальности;
2) возможность нескольких приложений функционировать и обмени-ваться информацией в одном и том же расширенном пространстве;
3) отсутствие ограничений на формы создаваемых окон приложений.

Проект позволяет в полной мере реализовывать трехмерные интерфейсы, а также работать с набором приложений расширенной реальности одновременно (изменять несколько текстовых документов, работать в интернете и т.д).

Дальнейшими перспективными направлениями являются:

1) подключение других видов манипуляторов (человеческие руки);
2) создание интерфейсов межсетевого взаимодействия;
3) реализация набора клиентских приложений для данной платформы.

Библиографические ссылки

1. **Рихтер Дж.** Windows для профессионалов: создание эффективных Win32-приложений с учетом специфики 64-разрядной версии Windows / Дж. Рихтер – М., 2008.
2. **Billinghurst M.** Real World Teleconferencing / M. Billinghurst Cheok A., H. Kato, S. Prince // CHI EA'99 – 1999. – P. 194 – 195.
3. **Schmalstieg D.** The Studierstube Augmented Reality Project /D. Schmalstieg // Presence: Teleoperators and Virtual Environments. – 2007. – No 11 – P. 33 – 54.
4. **Fiala M.** Artag, a fiducial marker system using digital techniques. / M. Fiala // CVPR'05 - IEEE Computer Society, San Diego. – 2005. – No 1 – P. 590 – 596.
5. **Gebril Z**. Innovative Learning of Solar System using Augmented Reality for Primary School Children / Z. Gebril, K. Ah-Choo, Y. May-Chan, B. Parhizkar // ICEMT2012. – 2012. – No 41 – P. 156–160.
6. **Kaufmann H.** Physics Playground: Physics Education in Virtual Reality / H. Kaufmann // Virtual Reality Systems for Psychology – 2006. – P. 227 – 240.
7. **Molla E.** Augmented Reality for Board Games / E. Molla, V. Lepetit // Mixed and Augmented Reality (ISMAR) – Seoul, Korea. – 2010. – No 9.

Надійшла до редколегії 26.02.2013 р.