

Система удаленного присутствия с использованием шлема виртуальной реальности*

А.Г. Старостин, В.В. Афанасьев, А.В. Игнатенко

fsstar.mas@gmail.com | {vafanasjev|ignatenko}@graphics.cs.msu.ru

Москва, Россия, Московский государственный университет имени М.В.Ломоносова

Данная статья описывает систему удаленного присутствия, состоящую из удаленного робота, представленного трехосевой подвижкой с двумя камерами и манипулятора, закрепленных на колесной платформе, и локального компьютера с подключенным шлемом виртуальной реальности, от которого через Интернет передаются команды управления, генерируемые в том числе с помощью определения положения шлема. В шлем выводится стереоизображение, получаемое от робота, создавая у пользователя ощущение восприятия глубины и эффект присутствия. Ключевой особенностью системы является использование общедоступных аппаратных компонентов.

Ключевые слова: виртуальная реальность, удаленное присутствие, стерео-видео, эффект присутствия

1. Введение

Система удаленного присутствия предполагает наличие удаленного управляемого устройства, способного передавать информацию об окружающем пространстве пользователю, который может находиться на значительном расстоянии от самого устройства.

Роботы удаленного присутствия как правило обладают камерами, закрепленными на подвижных конструкциях, позволяющих менять направление обзора. Робот может также обладать возможностью перемещения в пространстве и механического воздействия на окружающее пространство с помощью управляемых конечностей.

При работе робота в динамическом окружении ключевой характеристикой качества работы является скорость реакции на команды пользователя и оперативность передачи информации, в общем называемая общей задержкой работы системы, выражающейся в секундах между отдачей команды пользователем и восприятием результата ее выполнения.

Системы удаленного присутствия имеют множество применений. В области медицины используются телеуправляемые роботы-хирурги, позволяющие выполнять сложные операции. Роботов удаленного присутствия можно использовать для выполнения опасных или трудновыполнимых для человека задач, таких как военная разведка, исследование глубин океана, работа в открытом космосе. В повседневной жизни роботов можно использовать для образовательных и познавательных целей – виртуального путешествия в отдаленные уголки планеты.

Технология виртуальной реальности, рассматриваемая в данной работе, предполагает полное замещение визуального восприятия реального мира вир-

туальным с помощью специального шлема, проецирующего изображения напрямую в глаза пользователя. В последнее время шлемы виртуальной реальности постепенно набирают массовую популярность, в связи с чем прослеживается тенденция снижения цен на данные устройства при увеличении качества, что позволяет расширять область их применения.

Применение шлема виртуальной реальности в системе удаленного присутствия позволяет создавать у пользователя более сильное ощущение нахождения в удаленной точке по сравнению с использованием экранов для вывода изображений. Описанное ощущение называется эффектом присутствия [4]. Целью разработки данной системы является создание программно-аппаратной платформы для исследований в сфере виртуальной и дополненной реальности, в том числе для обеспечения возможности реализации студенческих проектов в указанных сферах.

У разрабатываемой системы существует ряд аналогов, использующих шлемы виртуальной реальности и стереокамеры на подвижках. Например, имеется ряд проектов (в том числе коммерческих) с неопубликованным исходным кодом: Skydrone FPV¹, DORA², AR-Rift³. Среди аналогичных систем существуют такие, которые используют альтернативные каналы связи, например, аналоговую передачу видео, как проект Oculus FPV⁴. Проект stereo_raspi⁵ во многом схож по пользовательским функциям с разрабатываемой системой, однако использует одноплатный компьютер Raspberry

¹<http://www.skydrone.aero/>

²<http://doraplatform.com/#about>

³<http://willsteeptoe.com/post/67571388490/ar-rift-ar-showcase-part-7>

⁴<https://github.com/Matsemann/oculus-fpv>

⁵https://github.com/coreyshuman/stereo_raspi

Работа опубликована по гранту РФФИ №16-07-20482.

PI вместо пары из Arduino и портативного компьютера.

2. Структура системы

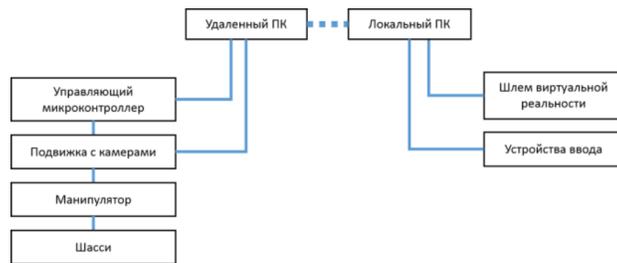


Рис. 1: Общая схема системы в разрезе аппаратных компонентов.

Разрабатываемая система состоит из следующих аппаратных компонентов, представленных на общей схеме на рис. 1:

- Подвижный робот
 - Портативный компьютер на базе процессора Intel и ОС Windows
 - Управляющий микроконтроллер Arduino
 - Трехосевая подвижка с камерами Ximea (рис. 2)
 - Колесная платформа
 - Рука-манипулятор
- Терминал управления
 - Локальный компьютер на базе процессора Intel и ОС Windows
 - Шлем виртуальной реальности Oculus Rift DK2

Камеры выбраны исходя из требуемого разрешения и частоты кадров. Разрешение 1280 x 1024 – на момент выбора камер доступное и вполне достаточное для комфортного восприятия человеком, кроме того, оно близко к разрешению используемого шлема виртуальной реальности (960 x 1280 на один глаз). Частота кадров 60 Гц также достаточная для комфортного восприятия видео через шлем: экспериментальным путём было выяснено, что при частоте кадров ниже 40 Гц человек наблюдает дрожание картинки в шлеме виртуальной реальности.

Выбор операционной системы обусловлен особенностями программной реализации. Выбор процессора и шлема виртуальной реальности – соображениями доступности.

Терминал и робот связываются через Интернет, причем робот оснащен модулем Wi-Fi, встроенным в портативный компьютер, что позволяет ему свободно перемещаться в пространстве.

На программируемых устройствах – портативном компьютере, локальном компьютере и микроконтроллере, работают соответственно программные компоненты «Сервер», «Клиент» и «контроллер»,



Рис. 2: Трехосевая подвижка.

обеспечивающие взаимодействие пользователя и робота.

Взаимодействие заключается в передаче команд управления (поворот головы, изменение выдержки, передвижение, команды манипулятору) от локального компьютера на удаленный и передачи стереовидеопотока в обратном направлении.

Для передачи стереовидеопотока получаемые с камер изображения на портативном компьютере объединяются в одно (что препятствует возникновению временной рассинхронизации, увеличивающей дискомфорт от восприятия стереовидео [5]) и сжимаются кодеком h.264 с помощью Intel Media SDK. Образующийся поток байт передается по сети и распаковывается на локальном компьютере тем же средством. Полученные изображения выводятся в шлем виртуальной реальности с помощью Oculus SDK и OpenGL.

Выбор пакета для кодирования видео обусловлен необходимостью эффективной работы на портативном компьютере заданной конфигурации. В процессе разработки были протестированы реализации кодирования видео с помощью средств OpenCV и libavcodec (ffmpeg). На каждом этапе тестирования из-за недостаточной эффективности были приняты решения использовать более низкий уровень взаимодействия с аппаратурой.

Выбор кодека производился из соображений популярности (больше документации и выше вероятность аппаратной поддержки) и эффективности (лучшее качество при меньших затратах на сжатие и распаковку).

Микроконтроллер и портативный компьютер связаны по USB через виртуальный последовательный порт. Компьютер передает микроконтроллеру команды управления некоторых типов, которые затем преобразуются в сигналы контроля для сервомоторов.

Для связи компьютера с камерами используется высокоскоростной канал связи по USB 3.0. С помощью специального API, предоставляемого разработчиками камер, обеспечивается управление параметрами съемки, а также прием кадров. Размер кадра – 1280 x 1024 пикселя, максимальная частота кадров – 60 Гц.

Разработанный программный комплекс обеспечивает модульность, выражающуюся в возможности

замены отдельных компонентов с целью поддержки другой аппаратуры либо использования симулятора, не требующего наличия материального робота, при сохранении общей структуры системы.

Программный комплекс представляет собой решение для Microsoft Visual Studio 2015, написанное на языке C++ с использованием зависимых библиотек Intel IPP, Intel Media SDK (с использованием DirectX для работы с видеопамятью), Oculus SDK (с использованием OpenGL), Ximea API и boost. Кроме того, имеется проект для среды Arduino, реализующий функции управляющего микроконтроллера.

3. Минимизация задержки

Как было сказано ранее, для комфортной работы пользователя в системе удаленного присутствия должны быть приняты меры по минимизации общей задержки. Для этого общая задержка разделена на составные этапы, каждый из которых измеряется, либо оценивается отдельно:

Задержка передачи команд управления: получение команд от пользователя, передача команд по сети удаленному компьютеру, передача команд управляющему микроконтроллеру, передача команд сервомоторам.

Задержка реакции на команды управления: движение сервомоторов, выдержка кадра.

Задержка передачи видеопотока: получение изображений с камер, дебайеринг, конвертация цветового пространства, сжатие изображений, передача сжатого потока на локальный компьютер, распаковка изображений, обратная конвертация цветового пространства, отрисовка виртуального мира.

Для измерения вклада этапов обработки в величину общей задержки код системы дополнен вызовами функций замера времени и соответствующими расчетами. Не все этапы удается измерить, поэтому далее перечислены величины, которые оцениваются сверху аналитически.

Задержка, вызванная движением сервомоторов, зависит от соотношения скорости движения головы пользователя и максимальной скорости движения сервомотора. С помощью контрольных измерений установлено, что скорость движения используемых сервомоторов примерно константна и равна 450 градусов/сек. При этом измеренная средняя комфортная скорость движения головы составляет 50 градусов/сек. Данная скорость была замерена при использовании системы. Исходя из указанных величин можно заключить, что комфортная скорость достаточно медленная по сравнению со скоростью сервомоторов, и что задержка, вызванная движением сервомоторов может быть оценена сверху величиной 2 мс.

Задержка получения изображений с камер оценивается константой, рассчитанной на основе пропускной способности USB-канала и битрейта камеры. Результат – 2.7 мс.

Задержка получения команд управления от пользователя (поворот головы или нажатие клавиши) оценивается равной 2 мс согласно спецификации используемого шлема [3].

Задержка передачи команд управления микроконтроллеру оценивается сверху 1 мс, так как объем передаваемых команд в байтах невелик (в среднем 30 байт).

Задержка передачи команд управления сервомоторам (предшествующая их движению, оценка задержки которого была приведена выше) оценивается сверху 10 мс, так как такая частота задана для широтно-импульсной модуляции.

Задержка отрисовки кадра оценивается исходя из частоты кадров. В шлеме виртуальной реальности выводится 75 кадров в секунду, поэтому задержка отрисовки составляет 13 мс.

В таблице 1 приведены оцененные и измеренные величины задержки.

Таблица 1: Результаты оценки и измерений компонентов общей задержки

Величина	Среднее
Передача команд управления	22 мс
Получение от пользователя	2 мс
Передача по сети	6 мс
Передача микроконтроллеру	1 мс
Передача сервомоторам	10 мс
Задержка реакции	16 мс
Движение сервомоторов	2 мс
Выдержка экспозиции	16 мс
Передача видеопотока	64 мс
Обработка на сервере	29 мс
Получение изображений с камер	2.7 мс
Дебайеринг	4.5 мс
Конвертация	6.4 мс
Сжатие изображений	15 мс
Передача по сети	10 мс
Обработка на клиенте	10 мс
Распаковка изображений	8.1 мс
Обратная конвертация	1.5 мс
Отрисовка виртуального мира	13 мс

Для минимизации задержки применяются прежде всего оптимизации программного кода. Используются режим сжатия видео с низкой латентностью, не требующий хранения буфера нескольких кадров. Кроме того, используется конвейеризация обработки с целью эксплуатации многоядерности со-

временных процессоров. Потоки, выполняющие отдельные операции, синхронизируются с помощью специальных двойных буферов, принцип действия которых показан на рис. 3.

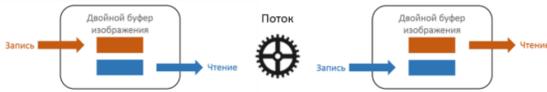


Рис. 3: Принцип работы синхронизационного буфера.

4. Эффект присутствия

Согласно распространенному мнению, хороший интерфейс – это незаметный интерфейс, способствующий выполнению задачи. Эффект присутствия позволяет создать наиболее естественный интерфейс управления удаленным роботом – с помощью движения тела пользователя. В данном случае только головы. Для создания эффекта присутствия помимо достаточного качества изображения и высокой частоты кадров необходимо обеспечение минимальной задержки реакции на действия пользователя. Согласно внутренним исследованиям компании Oculus, максимальная величина задержки, не вызывающая явного дискомфорта у пользователя при использовании шлема виртуальной реальности, составляет 20 мс.

В разрабатываемой системе при использовании большого количества компонент уменьшить общую задержку до комфортной величины с учетом наличия нестабильной неконтролируемой задержки передачи по сети не представляется возможным.

Поэтому применяется техника компенсации, заключающаяся в разделении первичной и вторичной реакции системы. Изображение в шлеме выводится на виртуальном экране, расположенном на некотором отдалении перед виртуальной камерой (см. рис. 4, виртуальная камера расположена в точке O). Экран привязан к глобальной системе координат, поэтому движение экрана и головы не синхронны.

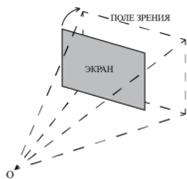


Рис. 4: Схема виртуального пространства.

Первичная реакция состоит в движении виртуальной камеры согласно движению головы пользователя. Это происходит в рамках одного компьютера за время, меньшее времени отрисовки одного кадра, поэтому укладывается в комфортный предел.

Вторичная реакция состоит в смещении экрана согласно оценке направления съемки конкретного кадра. Оценка может быть получена как явно с помощью измерения угла поворота сервомоторов при съемке (при наличии необходимой аппаратуры), так и неявно путем примерного расчета общей задержки для конкретного кадра и хранения траектории головы. В разработанной системе используется неявный способ. Виртуальный экран запаздывает относительно движения головы на рассчитанную величину задержки для каждого кадра.

Размер виртуального экрана имеет значение, поскольку от него зависит воспринимаемый размер объектов. Для реалистичности он должен быть согласован с углом зрения камер. Угол зрения камер можно оценить с помощью аналитического расчета на основе спецификации камеры, либо измерить с помощью контрольной съемки. При разработке системы использованы оба метода и для имеющихся камер получена величина горизонтального угла, равная 47° . Если виртуальный экран плоский, то необходимо, чтобы при съемке камера сохраняла перспективную проекцию, так как при отрисовке виртуального пространства она будет применена к экрану. В общем случае это не так, и требуется коррекция искажений камеры с целью восстановления перспективной проекции. Для этого камеры откалиброваны с помощью средства GML Camera Calibration Toolbox. Затем полученные коэффициенты используются во фрагментном шейдере для обратной деформации изображения по модели Брауна [2].

Кроме того, экран не должен быть объектом виртуального мира, поскольку тогда при отрисовке будет вноситься дополнительное смещение, вызванное смещением виртуальной камеры для получения левого и правого ракурса. Смещение ракурсов уже заложено в изображение, выводимое на экран, поэтому дополнительное смещение корректируется с помощью сдвига экрана при отрисовке для каждого ракурса в противоположную сторону (рис. 5).

Наконец, требуется соответствие углов поворота головы и трехосевой подвижки. Для этого сервомоторы необходимо откалибровать, чтобы получить соответствие длины ШИМ сигнала и получаемого угла поворота. Для используемых в системе сервомоторов получена линейная модель соответствия, которая используется для контроля сервомоторов в микроконтроллере.

Следует отметить, что расстояние между камерами влияет на ощущение размера воспринимаемых объектов. В соответствии с [1] была выбрана величина в 65 мм, являющаяся средней величиной для межзрачковое расстояние у взрослых мужчин. Таким образом обеспечивается соответствие воспринимаемых размеров реальным, что, однако, не является строгой необходимостью.

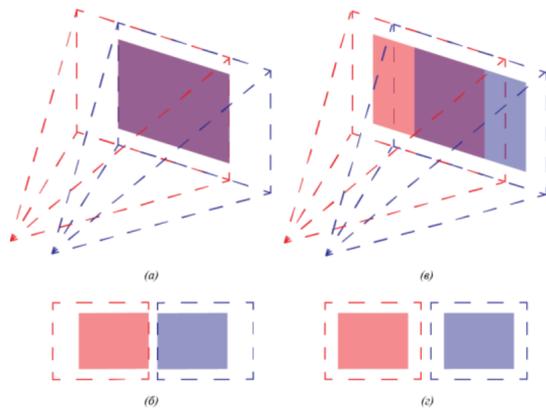


Рис. 5: (а) схема виртуального мира без корректирующего сдвига экрана, (б) вид из камер без сдвига, (в) схема виртуального мира с корректирующим сдвигом, (г) вид из камер со сдвигом.

5. Заключение

В статье описана разработанная система удаленного присутствия с использованием шлема виртуальной реальности, обеспечивающая естественное взаимодействие пользователя с удаленным роботом с помощью поворотов головы.

Описаны возникающие проблемы, возникающие на пути создания эффекта присутствия, и приведены методики их решения. В результате система предоставляет пользователю возможность комфортного управления. Разработанная система использует общедоступные аппаратные компоненты и поэтому может быть тиражирована при относительно небольших трудозатратах. Модульность системы и распространённость используемой системы программирования позволяет привлекать к развитию системы других разработчиков и студентов, обеспечивая невысокий порог вхождения.

В дальнейшем планируется дальнейшее развитие системы с помощью добавления новых способов управления и использования дополненной реальности для более глубокого взаимодействия с удаленным пространством.

Литература

- [1] Dodgson N.A. Variation and extrema of human interpupillary distance // *Electronic Imaging 2004*. International Society for Optics and Photonics. 2004. С. 36-46
- [2] Hugemann W. Correcting lens distortions in digital photographs // *Ingenieurburo Morawsky + Hugemann*. Leverkusen, Germany. 2010.
- [3] LaValle S.M. et al. Head tracking for the Oculus Rift // *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2014. С. 187-194.
- [4] Mestre D.R. Immersion and Presence // *Le traité de la réalité virtuelle*. Paris: Ecole des Mines de Paris. 2006. С. 309-38.
- [5] Voronov A., Vatolin D. et al. Methodology for stereoscopic motion-picture quality assessment // *Stereoscopic Displays and Applications*, vol. 8648. 2013. С. 864810-1-864810-14