

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное
учреждение высшего образования
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ

РАЗРАБОТКА ВИРТУАЛЬНОЙ
И ДОПОЛНЕННОЙ РЕАЛЬНОСТИ

Учебное пособие



УДК 004.946
ББК 32.973.26-018.2
Р17

Рецензенты:

доктор технических наук, профессор, главный научный сотрудник
факультета информационных технологий и программирования
Национального исследовательского университета ИТМО *А. А. Шалыто*;
Кандидат технических наук, доцент кафедры информационных технологий
Санкт-Петербургского государственного
архитектурно-строительного университета *Т. Н. Костюнина*

Авторы: Д. А. Булгаков, А. В. Никитин, Н. Н. Решетникова,
И. А. Ситников

Р17 Разработка виртуальной и дополненной реальности: учеб.
пособие / под общ. ред. д-ра техн. наук, проф. М. Б. Сергеева.
– СПб.: ГУАП, 2021. – 77 с.
ISBN 978-5-8088-

Рассмотрены теоретические вопросы и актуальные аппаратно-программные средства систем виртуальной и дополненной реальности.

Предназначено для студентов направления 09.03.01 «Информатика и вычислительная техника», изучающих дисциплину «Разработка виртуальной и дополненной реальности», реализуемую в рамках проекта «Университет FutureSkills».

Пособие может быть полезно специалистам в области информационных и коммуникационных технологий, а также студентам и аспирантам по соответствующим направлениям подготовки.

УДК 004.946
ББК 32.973.26-018.2

ISBN

© Санкт-Петербургский государственный
университет аэрокосмического
приборостроения, 2021

СПИСОК СОКРАЩЕНИЙ

ВР, VR – виртуальная реальность (англ. Virtual Reality).

ВС – виртуальная среда.

ДВ, AV – дополненная виртуальность (англ. Augmented
Virtuality).

ДР, AR – дополненная реальность (англ. Augmented Reality).

ВДР – виртуальная и дополненная реальность.

РВДР – разработка виртуальной и дополненной реальности.

СДР – смешанная и дополненная реальность.

СИИ – сенсомоторная иммерсивность и интерактивность.

ЦР – цифровые реальности.

UI – англ. User Interface (пользовательский интерфейс).

ПК – персональный компьютер.

ПО – программное обеспечение.

API – англ. Application Programming Interface (программный
интерфейс приложения).

SDK – англ. Software Development Kit (средства для разработки
ПО).

ВВЕДЕНИЕ

В настоящее время Министерство науки и высшего образования Российской Федерации, Автономная некоммерческая организация «Агентство развития профессионального мастерства (Ворлдскиллс Россия)», федеральный центр развития компетенций FutureSkills в лице ФГАОУ ВО «Санкт-Петербургский государственный университет аэрокосмического приборостроения» реализуют проект «Федеральный Университет FutureSkills».

Проект FutureSkills опирается на направление WorldSkills, которое ориентируется на приобретение практических навыков обучающимися и их проверку в форме соревнований.

Основной целью проекта FutureSkills является трансформация кадровой подготовки в системе высшего образования за счет внедрения новых, актуальных для цифровой экономики учебных дисциплин и разработку комплекса инновационного методического сопровождения интеграции FutureSkills в образовательную деятельность в ведущих вузах страны для обеспечения роста экономики и культуры Российской Федерации.

Предполагается, что ключевые тренды в экономике влияют на рабочую деятельность, на основе анализа которой прогнозируются требования к будущим навыкам работников и, соответственно, к моделям подготовки [1, 2].

В связи с этим рассматривается отмирание существующих профессий и появление новых, что нашло отражение в постоянно обновляемом «Атласе новых профессий» [3].

В рамках проекта FutureSkills используются следующие термины [1]:

Навык – это способность осуществлять определенную деятельность, сформированная путем повторения и доведения до автоматизма.

Компетенция – это комплексная характеристика готовности человека применять полученные знания, умения и личностные качества в профессиональной деятельности.

Английский термин «skill» означает способность выполнить задачу с предопределенным результатом. Это определение шире, чем привычное нам значение слова «навык», и в некоторых случаях приближается к значению слова «компетенция».

Компетенция «Разработка виртуальной и дополненной реальности» относится к так называемым сквозным технологиям, которые формируют ключевые научно-технические направления, оказы-

вающие наиболее существенное влияние на развитие различных рынков. Она включена в государственные программы: образование, цифровая экономика, культурная политика, национальная технологическая инициатива [4].

Разработана «Дорожная карта развития «сквозной» цифровой технологии «Технологии виртуальной и дополненной реальности» [5], в которой определены основные субтехнологии и их уровень развития в мире и России, рыночные тенденции и эффекты от развития и применения, связь с другими сквозными технологиями, риски и ограничения развития, текущее состояние и целевые показатели развития, этапы и инструменты реализации.

Однако она не содержит строгих определений терминов, необходимых для правильного понимания систем виртуальной и дополненной реальности, а также не включает дополненную виртуальность и мультимодальный интерфейс, которые взаимосвязаны с двумя другими и по важности не уступают им в связи с развитием технологий передачи данных от реальных объектов в их цифровые модели (цифровые двойники).

В дальнейшем авторы наряду с названием компетенции «Разработка виртуальной и дополненной реальности» будут использовать термин «цифровые реальности» как обобщающий для технологий виртуальной и смешанной реальности (дополненная реальность и дополненная виртуальность), виртуальных миров, мультимодального интерфейса, которые поддерживают работу с сенсомоторными характеристиками человека.

Существующая компетенция «Разработка виртуальной и дополненной реальности» в рамках WorldSkills на сегодняшний день:

- не определяет основных понятий и методов, не включает актуальные цифровые реальности в данной области (дополненную виртуальность, смешанную реальность, виртуальные миры, мультимодальный интерфейс);

- не соответствует уровням подготовки специалистов;

- не отражает процессы жизненного цикла создания приложений на основе данных методов и технологий, принятые и прогнозируемые в промышленности и других областях с учетом международных и отечественных стандартов, что необходимо учитывать для создания систем на основе цифровых реальностей, соответствия их реальным научно-техническим, экономическим и социальным процессам в обществе, прогнозам будущих навыков, а также для объективизации результатов соревнований.

Учебное пособие подготовлено специалистами кафедры вычислительных систем и сетей ГУАП кандидатами технических наук, доцентами А. В. Никитиным и Н. Н. Решетниковой (глава 1) а также старшим преподавателем Д. А. Булгаковым и программистом И. А. Ситниковым (глава 2). Общая редакция выполнена доктором технических наук, профессором М. Б. Сергеевым.

1. МЕТОДЫ И ТЕХНОЛОГИИ ЦИФРОВЫХ РЕАЛЬНОСТЕЙ

1.1. Основы человеко-машинного взаимодействия

1.1.1. Основные термины человеко-машинного взаимодействия

Рассмотрим некоторые основные термины в области человеко-машинного взаимодействия [6, 7, 8, 9, 10].

Интерфейс (англ. Interface – взаимодействие, сопряжение) – совокупность возможностей, способов и методов взаимодействия двух систем.

Человеко-машинный интерфейс (ЧМИ) – методы и средства обеспечения непосредственного взаимодействия между оператором и технической системой, предоставляющие возможности оператору управлять этой системой и контролировать ее работу.

Интерфейс пользователя, пользовательский интерфейс (UI – англ. user interface) – представляет собой совокупность средств и методов, при помощи которых пользователь взаимодействует с компонентами интерактивной системы (программное обеспечение и\или аппаратное обеспечение), которые предоставляют пользователю информацию и являются инструментами управления для выполнения определенных задач.

Интерактивная система (interactive system) – система компонентов аппаратного и программного обеспечения, которая получает информацию, вводимую пользователем, и передает ему свой ответ, помогая в работе или выполнении задачи.

3D взаимодействие – взаимодействие человека с компьютером, при котором задачи пользователя выполняются непосредственно в реальном или виртуальном трехмерном пространственном контексте.

3D-интерфейс пользователя (3D UI) – пользовательский интерфейс, предполагающий трехмерное взаимодействие.

3D интерактивные методы и техники – методы выполнения задачи через интерфейс, включая аппаратные компоненты, программные и управления, метафоры или концепции, а также универсальные задачи: выбор и манипулирование, навигация и перемещение, управление системой.

1.1.2. Термин «виртуальный»

В гуманитарных науках под виртуальностью чаще всего понимают «объект или состояние, которые реально не существуют, но могут возникнуть при определенных условиях» [11, 12, 13].

В [14] уточняется: «Мы должны помнить, что «виртуальный» не является противоположным «реальному», как мы склонны считать, а скорее, противоположным «существующему», «фактическому» в его исходном значении «существующий в данное время». Яйцо является виртуальным цыпленком; оно запрограммировано стать цыпленком и должно им стать, если ничего не случится в процессе его развития».

В информатике и вычислительной технике используется другое понимание. Например, в [13] упоминается: «Другой подход к виртуальности сформировался под влиянием развития компьютерных и информационных технологий. С помощью современных технических средств можно погрузиться в виртуальную реальность, в которой субъект не будет различать вещи и события действительного и виртуального мира: мир дан ему непосредственно в его ощущениях, а они оказываются на этом уровне неразличимыми».

Схожее понимание термина приведено в [12]: «реализованный программно, симулированный, имитированный с помощью компьютера» и «то же, что эффективный, заменяющий реальный параметр в упрощенной модели», в [15]: «Имитируемый компьютером, связанный с виртуальной реальностью», а также в ГОСТ 15971-90 [16]: «Виртуальный (Virtual) – определение, характеризующее процесс или устройство в системе обработки информации кажущихся реально существующими, поскольку все их функции реализуются какими-либо другими средствами».

1.1.3. Понятие «реально-виртуальный континуум»

Реально-виртуальный континуум (Reality-Virtuality Continuum) – понятие, используемое для описания концепции существования непрерывной шкалы в диапазоне от полностью виртуальной до полностью реальной среды. Она охватывает все возможные варианты и композиции из реальных и виртуальных объектов. Эта концепция была впервые представлена Полом Мильгрэмом (Paul Milgram) и иллюстрируется в табл. 1.1 [17].

Таблица 1.1

Реально-виртуальный континуум

Реальная среда	Смешанная реальность		Виртуальная среда
	Дополненная реальность	Дополненная виртуальность	

В табл. 1.1 представлены:

– Реальная среда (Real Environment) – определяет любую окружающую среду, состоящую исключительно из реальных объектов, и включает то, что мы могли бы наблюдать/ощущать (фрагмент реального мира) непосредственно лично или через некоторое устройство (очки, камеру мобильного телефона, бинокль и т. п.).

– Виртуальная среда (Virtual Environment) – определяет среду, состоящую исключительно из виртуальных (цифровых) объектов, например, компьютерные игры или отображаемые на мониторе результаты графического моделирования.

– Смешанная реальность (Mixed Reality) – физические и цифровые объекты сосуществуют и взаимодействуют в режиме реального времени. Смешанная реальность включает в себя:

– дополненную реальность (Augmented Reality) – когда виртуальность дополняет реальное;

– дополненную виртуальность (Augmented Virtuality) – когда реальное дополняет виртуальное.

Понимание термина «виртуальный» как подмены реальных ощущений имитированными при помощи компьютерных средств обуславливает необходимость определения основных понятий психофизиологии человека как для описания его сенсомоторных связей с моделируемой средой, так и с точки зрения понимания возможностей существующих и перспективных устройств ввода и вывода систем ЦР.

1.1.4. Поэтапная модель обработки информации

Рассмотрим поэтапную модель обработки информации человеком, сопоставленную с основными его характеристиками: ощущениями, восприятием, познанием и эргономикой, которые рассмотрим ниже для понимания РВДР (рис. 1.1) [18].

В этой модели стимулы или события получают осмысленную интерпретацию, основанную на воспоминаниях о прошлом опыте. В ответ на восприятие, действия могут быть выбраны и выполнены, либо информация может сохраняться в кратковременной рабочей памяти. Рабочая память имеет ограниченные возможности и сильно зависит от ресурсов внимания (чтобы сохранить что-то в рабочей памяти, нужно уделять этому внимание). Напротив, объем долговременной памяти человека огромен, она хранит информацию о нашем мире, концепциях и процедурах, не подвергаясь прямому влиянию внимания. В результате выполненных действий пользова-

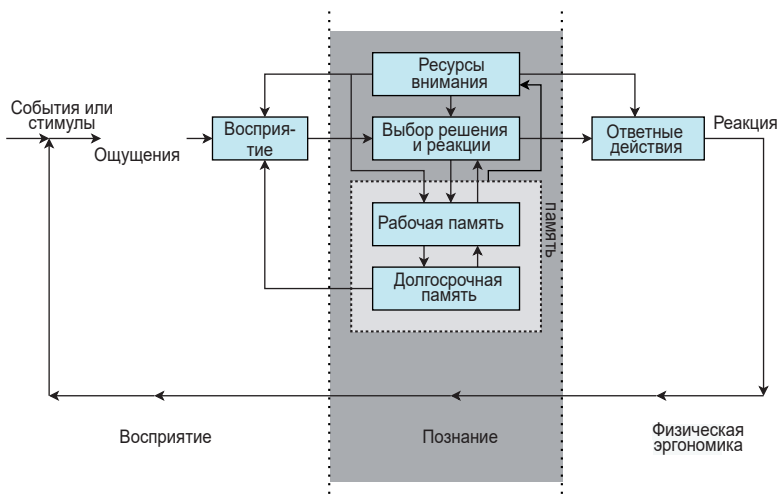


Рис. 1.1. Обработка информации и связанные с ней вопросы человеческого фактора [18]

тели могут получать обратную связь, которая возвращается в цикл обработки информации.

Далее рассмотрим основные понятия, представленные на рис. 1.1.

Ощущение – «это отражение в сознании человека отдельных свойств и качеств предметов и явлений, непосредственно воздействующих на его органы чувств» [19]. Органы чувств – это те механизмы, с помощью которых информация об окружающей нас среде поступает в кору головного мозга. С помощью ощущений отражаются основные внешние признаки предметов и явлений (цвет, форма, величина, особенности поверхности предметов, звук, вкус и др.) и состояние внутренних органов (мышечные ощущения, боль и др.).

Приведем *классификацию ощущений* по их расположению [19]:

1. «Экстерорецептивные ощущения отражают свойства предметов и явлений внешней среды». К ним относятся зрительные, слуховые, обонятельные, вкусовые, температурные и тактильные ощущения.

2. «Интерорецептивные ощущения отражают состояние внутренних органов». Это болевые ощущения, ощущения равновесия и ускорения.

3. «Проприоцептивные ощущения – это ощущения, отражающие движение нашего тела. С помощью мышечно-двигательных ощущение-

ний человек получает информацию: о положении тела в пространстве, о взаимном расположении всех его частей, о движении тела и его частей, о сокращении, растяжении и расслаблении мышц и т. п.».

Также приведенная ниже классификация интересна для понимания существующих и будущих устройств ввода-вывода информации. Согласно [20], «в зависимости от физической природы воспринимаемых стимулов сенсорные рецепторы подразделяются на:

- хеморецепторы, чувствительные к действию химических веществ;
- фоторецепторы, чувствительные к световым сигналам;
- механорецепторы, чувствительные к давлению, вибрации, перемещению, степени растяжения;
- терморецепторы, чувствительные к изменениям температуры;
- осморецепторы, чувствительные к изменению осмотического давления крови».

Одним из подходов для описания взаимодействия человека с виртуальной средой является определение перцептивных и моторных (мышечных) систем человека по Дж. Гибсону (табл. 1.2 и 1.3) [34] – автору концепции экологической психологии, идея которой состоит в прямом восприятии потока информации в рецепторах органов чувств без опосредования когнитивными процессами в условиях реальной среды, а не обедненного эксперимента. Данное понятие во многом определяет исследования и разработки в областях дизайна, эргономики и систем ВР.

Восприятие – «это отражение в сознании человека непосредственно воздействующих на его органы чувств предметов и явлений в целом, а не отдельных их свойств, как это происходит при ощущении. Восприятие – не сумма ощущений, получаемых от того или иного предмета, а качественно новая ступень чувственного познания с присущими ей особенностями» [19, 20]. Существуют различные *виды восприятий* в зависимости от форм отражения (пространства, времени), степени организации и цели, а также различные свойства восприятия и физические характеристики.

С точки зрения РВДР нас в первую очередь интересуют восприятия:

- пространства – формы объемности и величины предметов, глубины и удаленности предметов, линейная и воздушная перспектива, а также зрительные иллюзии;
- времени – последовательности и длительности явлений, темпа и ритма;

Таблица 1.2

Моторные (мышечные) системы человека

Система	Цель	Применение	Другие системы
равновесие	ориентация за счет сил гравитации и ускорения	поддержание равновесия тела	вестибулярные органы
ориентированно-исследовательская	движение частей тела для получения внешних стимулов	информация от сенсоров или исследование	все другие органы чувств
локомоторная	движение тела или его частей в среде	перемещение из одной точки в другую	ориентированно-исследовательская и позирования
приема	движение частей тела для приема внутрь или освобождения	прием внутрь или освобождение	вкусовые, приема и другие функции тела
совершения действий	движения, нужные индивидууму	совершение действий, за-щита себя	локомоторные и другие
экспрессивная	движение для выражения себя; проявление эмоций или идентификация себя	совершение позиционных движений, движение лица и голоса	голосовые, слуховые, мускулы лица
семантическая	движения для выражения сигнальных действий, состояния или экспрессии	голосовые выражения	любые другие системы, связанные с передачей сигналов

Таблица 1.3

Перцептивные системы человека

Система	Модель активности	Рецептивные элементы	Анатомия органа	Активность органа	Стимулы	Внешняя информация
равновесия	Позирование и ориентация	механические и гравитационные рецепторы	вестибулярные органы	равновесие тела и балансирование	силы гравитации и ускорения	направление гравитации и ускорения
звуковая	слушание	механические рецепторы	улитки уха	ориентация по звуку	вибрации воздуха	природа и локация вибраторных событий
осязания	прикосновение	механические, термальные, кинематический	кожа, суставы, мускулы, жилия	исследование различного рода на основе придатков кожи, языка	деформация ткани, конфигурация суставов, растяжение волокон	контакт с формой и поверхностью объекта, состояние материала, вязкость, тепло, холод
вкуса	проба на вкус	химические и механические рецепторы	рот	смакование	химия введенных объектов	питательные вещества, биохимические элементы
запах	восприятие запаха	химические рецепторы	нос	активное вдыхание	химия паров	природа запахов
зрения	смотрение	фоторецепторы	окулярный механизм, включающий глаза и движения целого глаза	аккомодация, настройка, фиксация, конвергенция, сканирование	свет	размер, форма, состояние, цвет, текстура, движения

– движений, скорости, ускорения, продолжительности, характера, формы, амплитуды.

Внимание – это избирательная направленность сознания человека на определенные предметы и явления [10]. Один из важных процессов работы с 3D-интерфейсами.

В системе обработки информации внимание влияет на ряд подпроцессов, например, на то, как пользователи выполняют поиск в аналитических приложениях, какую информацию собирают из динамических пространственно-временных сред. Это влияние может быть использовано при проектировании интерфейсов для привлечения внимания пользователя к конкретной (срочной) информации в большом информационном пространстве. Хотя внимание до сих пор полностью не изучено, данные свидетельствуют о том, что оно *состоит из трех компонентов*:

- ориентации на сенсорные события;
- обнаружения сигналов для целенаправленной обработки;
- поддержания состояния бдительности или готовности.

Внимание является основным компонентом обработки визуальных стимулов и поискового поведения, т. е. включает в себя отбор некоторой информации для дальнейшей обработки и запрет на дальнейшую обработку другой информации.

Выделяют следующие *виды внимания*:

– Избирательное внимание (Selective attention) – это выбор того, какие события или стимулы следует обрабатывать.

– Сосредоточенное внимание (Focused attention) – это усилие по поддержанию обработки этих элементов, избегая при этом отвлечения внимания на другие стимулы или события.

– Разделенное внимание (Divided attention) – это способность обрабатывать более одного события или стимула в данный момент времени.

Таким образом, внимание концептуально можно рассматривать как процесс отбора.

Внимание включает в себя три *мыслительные операции*:

- 1) отвлечение внимания от текущего местоположения,
- 2) перемещение внимания в новое местоположение,
- 3) повторное привлечение внимания в новом месте для поддержки обработки в этом месте.

Внимание в одном сенсорном канале может вызвать внимание в другом канале. Например, на многолюдном вокзале друг может махать вам рукой. Но как только вы сосредоточите на нем свое вни-

мание, то сможете заметить, что он также зовет вас по имени. Интересно, что между различными сенсорными модальностями существует множество связей. Каким бы полезным ни было внимание в нашей повседневной жизни, его процессы, к сожалению, подвержены ошибкам, которые могут быть усилены определенными узкими местами. Примерами таких узких мест являются ограничения в наших сенсорных системах (например, низкая острота нашего периферического зрения) или разделение внимания на различные задачи.

Принятие решений, поведение и навыки

Как видно на рис. 1.1, процессы принятия решений находятся в центре обработки информации и зависят от сбора, организации и комбинирования информации из различных источников.

Принятие решения – это процесс выбора между двумя или более альтернативами, приводящий к реальным или мнимым последствиям. Сам процесс принятия решения основан на идентификации текущей ситуации, потенциальных действиях и их последствий. Типичным методом визуального определения структуры решений является так называемое дерево решений. В процессе принятия решений есть две проблемы, на которые стоит обратить внимание: *поведение и навыки*.

В контексте принятия решений под *поведением* понимается использование определенных стратегий для принятия решений, которые могут быть сформированы на основе опыта и могут зависеть от культуры, убеждений, привычек, социальных норм или взглядов человека. Поведение человека может регулироваться его эмоциональным состоянием.

В контексте принятия решений более глубокое изучение поведения пользователей может потенциально выявить такие проблемы, как различия в принятии решений или общее отношение к риску, влияющие на работу с 3D-интерфейсами.

При проектировании 3D-интерфейсов поведение часто влияет на дизайн естественных пользовательских интерфейсов, например, тех, которые используют жесты.

На поведение пользователей также влияют *навыки*. Навыки позволяют пользователям выполнять более точные и быстрые действия и могут быть приобретены со временем, при многократном выполнении аналогичных действий. Как таковые, они связаны с процессами принятия решений, памятью и моторикой.

Выделяют три различных *этапа приобретения навыков*:

1. Когнитивный этап представляет собой декларативное знание и часто относится к сознательному набору инструкций.

2. Постепенно пользователи могут перейти в ассоциативную стадию, на которой поведение настраивается, а количество ошибок снижается.

3. Наконец, со временем пользователи входят в автономную стадию, на которой действия могут быть точно и быстро выполнены.

Считается, что навыки хранятся в нашей долговременной памяти как процедурные знания.

Приобретение навыков – одно из основных направлений развития 3D-интерфейсов. Определенные действия могут быть трудными для изучения и, следовательно, требуют приобретения навыков. Кроме того, передача навыков является важным фактором во многих приложениях обучения с использованием виртуальной реальности, и ожидается, что навыки, полученные в виртуальной среде, также могут быть применены в реальном мире.

Выбор и контроль действий

Выбор и контроль действий тесно связаны с принятием решений, поскольку они включают выбор и выполнение ответных действий на основе определенных стимулов. Выбор действий – это автоматический процесс, который включает только простое познание.

Познание

Воспринимаемые стимулы обрабатываются когнитивными процессами для того, чтобы инициировать выполнение задачи. С точки зрения пользователя, знания обычно рассматриваются как информация о мире, которая считается достоверной, может быть обоснована и является последовательной. **Как видно на рис. 1.1**, познание является ключом ко многим процессам. Оно не только управляет процессами в памяти и мышлении, но также напрямую связано с восприятием и вниманием человека. Знания хранятся в долговременной памяти, где также хранятся другие виды воспоминаний. Долговременная память состоит из декларативной (или явной) и недеklarативной (или неявной) памяти. Декларативные (описательные) знания в основном состоят из фактов и событий, в то время как недеklarативные (процедурные) знания содержат первичную, процедурную (например, навыки и привычки) и ассоциативную память. На запоминание влияют подсказки и контекст, память очень ассоциативна.

Рабочая память используется для кратковременного хранения и выполняет манипуляции или действия по преобразованию фраг-

ментов информации. Исследования показали, что объем рабочей памяти и, следовательно, объем информации, который в ней хранится, различается у разных людей и дает общее представление об интеллекте человека.

Рассмотрим, как познание влияет на наше взаимодействие в пространственной среде, начиная с *осознания ситуации*. Осведомленность о ситуации, которая широко изучалась в авиации, обычно характеризуется как модель текущего состояния среды пользователя, т. е. восприятие элементов окружающей среды в определенном периоде времени и объеме пространства, понимание их значения и прогнозирование их статуса в ближайшем будущем, что напрямую связано с принятием решений и планированием действий. В 3D пользовательском интерфейсе осведомленность о ситуации влияет на то, как пользователь взаимодействует с окружающими объектами, поскольку постоянно необходимо отслеживать пространственную обстановку, в которой происходит действие. Особой областью интереса является поиск путей, который будет рассмотрен в п. 1.3.3.

Эргономика

Наряду с восприятием и познанием при разработке 3D пользовательских интерфейсов, необходимо обратить внимание разработчика на физическую эргономику, которая является важнейшим фактором, позволяя разрабатывать системы ВДР для эффективного и комфортного использования.

Физическая эргономика зависит от анатомических возможностей различных частей человеческого тела, которые определяют, как и насколько хорошо мы можем выполнять конкретную задачу, и связана в первую очередь с опорно-двигательной системой человека, поэтому важно понимать основы анатомии и физиологии человека.

1.2. Погружение, присутствие, киберболезнь

В этом разделе будут рассмотрены ключевые понятия разработчи- ки ВДР.

1.2.1. Погружение

Погружение определяется как:

– степень, в которой технология реализации интерфейса передает стимулы органам чувств [21];

– свойство моделируемой среды, вовлекающее человека через его сенсомоторные средства в поведение среды и снабжающее его непрерывным потоком стимулов, сопоставимым с естественным [24].

Погружение достигается посредством стимуляции органов чувств пользователя, при которой генерируются ощущения, позволяющие воспринимать виртуальную среду.

Погружение имеет отношение к техническим средствам создания виртуальной среды и связано с тем, в какой степени пользователь «окружен» искусственным сенсорным потоком и изолирован от сенсорного потока, исходящего от реальной среды (например, прилегание очков и т. п.).

Характеристики погружения [24]:

– Степень – количество используемых одновременно перцептивных ресурсов (зрение, слух, касания и т. п.) человека.

– Формы (позиции восприятия) – непосредственный участник среды (первая форма), видит себя в среде с точки зрения другого человека (вторая форма), видит среду и себя с позиции стороннего наблюдателя (третья форма).

В [21, 22] к основным *факторам, вызывающим погружение*, относятся:

– Восприятие нахождения себя в цифровой модели среды и естественное взаимодействие с ее объектами (ходьба с кинестетическими ощущениями, манипулирование предметами, изменение положения частей тела и т. п.).

– Изоляции от реальной среды с целью сосредоточения только на генерируемых компьютером стимулах. Выполнение этого требования определяется правильно подобранным оборудованием.

1.2.2. Понятие вовлеченности

Некоторые авторы [21, 22] помимо погружения вводят понятие **вовлеченность** – «психологическое состояние, переживаемое как следствие сосредоточения внимания на последовательном наборе стимулов или обоснованно связанных действий и событий».

Степень вовлеченности больше, если человек сосредоточен на стимулах среды, и снижается при его отвлечениях от среды, в том числе за счет плохого оборудования. Кроме того, вовлеченность зависит от объектов и их поведения в среде, которые привлекают и удерживают внимание.

Считается что погружение, зависящее от создающих виртуальную реальность технических средств, и вовлеченность, зави-

сящая от содержания рабочей среды человека, определяют присутствие.

1.2.3. Присутствие

Присутствие является одним из ключевых понятий систем ВДР. Однако единого его понимания не существует [21, 22]. Ранние работы сводят к сенсорно-перцептивной сфере.

Рассмотрим некоторые *современные взгляды* [21, 22, 23]:

– человек, испытывающий данное чувство, переживает виртуальную среду, происходящие с ним там события и свое нахождение в ней как нечто реальное, а не созданное с помощью технических средств;

– человек воспринимает себя в компьютерной виртуальной среде, а не в месте своего фактического пребывания;

– это ощущение нахождения «там» (внутри цифровой модели) в противоположность наблюдению со стороны;

– это сложный психосоциальный феномен, наблюдаемый при взаимодействии человека с некоей реальностью, отличной от непосредственно наблюдаемой им (обычной) реальности [24].

Факторы, вызывающие присутствие [21, 22]:

1) естественность взаимодействия;

2) контроль процессов среды пользователем;

3) реалистичность изображения;

4) длительность пребывания в виртуальной среде, ограниченная возможностью киберболезни;

5) наличие других людей в виртуальной среде;

6) аппаратное и программное обеспечение и сценарии предоставления информации с возможностью взаимодействовать с объектами и участниками виртуальной среды.

Выделяют следующие *типы присутствия* [21, 22]:

– Средовое присутствие – это степень реакции виртуальной среды на действия пользователя.

– Социальное присутствие – в виртуальной среде может находиться несколько человек, которые могут видеть друг друга и взаимодействовать между собой (см. раздел «Виртуальные миры»).

– Личное присутствие – это степень, в которой человек чувствует себя находящимся в виртуальной среде, устанавливается изменениями.

Существуют следующие *методы измерения присутствия* [21, 22]:

- субъективный;
- физиологический;
- поведенческий.

Субъективные методы (наиболее удобны):

– Рейтинговые шкалы (например, «по балльной шкале оцените естественность вашего взаимодействия с виртуальной средой»).

– Субъективные сообщения на открытые вопросы (например, «я действительно чувствовал нахождение в другом месте»).

– Метод парных сравнений виртуальных сред по уровню реализма и взаимодействия, задержек и т. п. (например, «в какой из двух виртуальных сред вы в большей степени ощущали присутствие?»).

– Оценка относительно какой-то величины, например, 100 («оцените естественность этой виртуальной среды по шкале от единицы до ста»).

– Метод кросс-модального подбора (например, «сделайте какой-то параметр восприятия таким же, насколько вы испытали сильное ощущение его в виртуальной среде»).

Физиологические методы – фиксация психофизиологических параметров человека: частоты сердечных сокращений, кожно-гальванической реакции и других параметров. Они используются в первую очередь для измерения реакции пользователя на изображение опасной среды (высота, огонь, взрыв и т. п.).

Поведенческий метод основан на эквивалентности действий человека в виртуальной среде его действиям в реальной среде, что предполагает восприятие виртуальной среды как реальной.

1.2.4. Киберболезнь

Одним из факторов, мешающих переживанию присутствия, является **киберболезнь** – недомогание, возникающие при взаимодействии человека с виртуальной реальностью, предположительно связанное с конфликтом между зрительной информацией и ощущениями, получаемыми другими сенсорными системами [21, 22].

Некоторые авторы указывают, что это не болезнь, а естественный физиологический ответ организма на помещение в виртуальную среду, связанный с предъявляемым организму необычным стимулом.

Основные симптомы сходны с симптомами морской болезни – это тошнота, рвота, головокружение, головная боль, утомление и др.

На настоящий момент предлагаются три *теории, объясняющие причины возникновения киберболезни*:

- теория сенсорного конфликта,
- токсиновая теория (ядов),
- теория постуральной неустойчивости.

1. Теория сенсорных конфликтов

Гипотеза – симуляторные расстройства возникают в результате рассогласования между сигналами сенсорных систем различных модальностей, принимающих участие в формировании представления об ориентации и положении тела в пространстве, к которым относятся вестибулярная, проприоцептивная и зрительная сенсорные системы.

Неадекватные сигналы одного из каналов могут привести к возникновению симуляторного расстройства. Например, в виртуальном пространстве перемещения объектов относительно неподвижного наблюдателя могут приводить к сенсорному конфликту его зрительной системы, которая получает сигналы о перемещении тела, тогда как вестибулярная и проприоцептивная системы сигнализируют о его статичном положении.

Критика – не объясняет различную устойчивость людей к этому фактору механизм формирования к недомоганию.

2. Токсिनговая теория

Гипотеза – эквивалентность ощущений отравления человека токсинами (ориентация в пространстве, головокружение и т. п.) симуляторным расстройствам, что позволяет предположить схожесть их формирования.

Критика – не объясняет различие разного проявления этого фактора у людей и не имеет прогностической силы.

3. Теория постуральной неустойчивости

Гипотеза 1 – недостаточность адекватных сенсорных сигналов приводит к иллюзии движения тела (векция) и нарушения равновесия. Например, человек видит свое движение в виртуальной среде на основе зрительных стимулов. Он старается сохранить равновесие, но привычные механизмы не работают, так как он неподвижен и это приводит к потере равновесия.

Гипотеза 2 – нарушения в поддержании позы всегда возникают раньше, чем иллюзия движения собственного тела. Сначала человек пытается адаптироваться к вращающейся среде и найти оптимальное положение тела, а потерпев в этом неудачу, испытывает симуляторное расстройство.

Критика – не объясняет индивидуальные различия в проявлениях и ничего не предсказывает.

Гипотеза 3 – один из механизмов адаптации заключается в выработке компенсаторных поведенческих актов, уменьшающих выраженность негативных симптомов. Например, если наблюдатель замечает, что при уменьшении движения головы уменьшается головокружение, то он, вероятно, и дальше будет использовать этот компенсаторный прием. На основе данных об эффекте привыкания была предложена модель воздействия ВС на человека, которая включала как психологические, так и физиологические адаптивные изменения конкретного организма. Согласно предложенной модели, при воздействии ВС происходят адаптационные изменения организма, которые откладываются в памяти как *виртуальный опыт*, изменяющий динамические индивидуальные характеристики взаимодействия наблюдателя с ВС. Эти данные следует учитывать при планировании любого эксперимента с применением систем ВС, особенно, если участники несколько раз проходят тестирование в ходе исследования [25].

Рассмотрим основные факторы возникновения киберболезни [21, 22]:

- технические,
- индивидуальные.

Технические:

- 1) ошибки в отслеживании положения головы (рассогласование изображения с ощущением неподвижности собственного тела);
- 2) задержка реакции среды на действия пользователя;
- 3) мерцание (низкое мерцание вызывает усталость глаз).

Индивидуальные изучены меньше, однако можно говорить о следующих:

- 1) ширина индивидуального поля зрения (периферия глаз более чувствительна);
- 2) пол (по некоторым данным, женщины более подвержены киберболезни);
- 3) плохое самочувствие;
- 4) положение, которое человек занимает в тренажере (согласно теории постуральной неустойчивости лучше сидеть, чем стоять);
- 5) отсутствие контроля над ситуацией (активное участие в событиях среды, лучше, чем пассивное наблюдение).

Преодоление киберболезни (некоторые советы):

- адаптация к виртуальной среде за счет регулярной работы с ней;
- пауза между сеансами от двух до пяти дней с целью восстановления организма, но не полной потери навыков переживания непривычных условий;

- активное взаимодействие со средой, вместо пассивного ее созерцания;
- корректная реакция и отсутствие взаимной задержки «пользователь-среда».

1.3. Интерактивность

1.3.1. Понятие интерактивности

Интерактивность (взаимодействие) – свойство моделируемой среды, предоставляющее возможность человеку в реальном времени осуществлять навигацию в среде и взаимодействие с ее объектами с ощущением их ответных реакций, а также изменять среду.

Рабочий процесс взаимодействия представлен на рис. 1.2.

Универсальные задачи трехмерного взаимодействия (см. рис. 1.4) [26–30]:

- Навигация:
 - перемещение: моторная составляющая;
 - поиск пути: когнитивный компонент.
- Селекция/выборка – определяет один или более объектов из набора;



Рис. 1.2. Рабочий процесс взаимодействия «пользователь – система» [26]

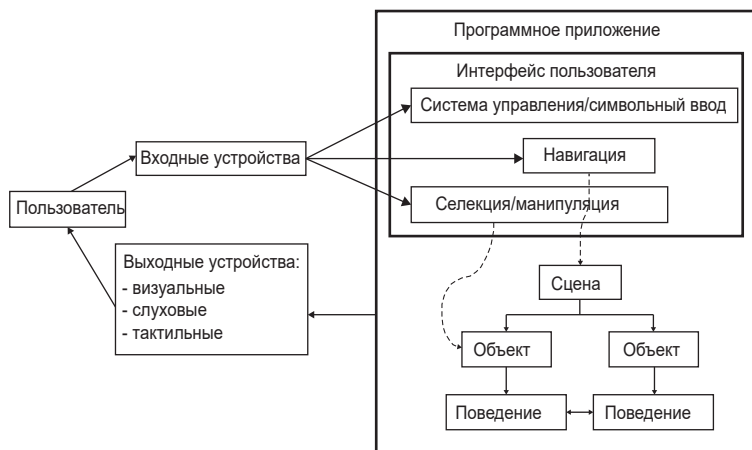


Рис. 1.3. Схема взаимосвязи понятий

– Манипуляции:

- определение положения и ориентации объекта;
- указание масштаба, формы, цвета и других атрибутов.
- Управление системой:
 - изменение состояния системы или режима взаимодействия;
 - может состоять из других задач.
- Символьный ввод.

Схема взаимодействия пользователя с системой приведена на рис. 1.3.

1.3.2. Задачи 3D-манипуляций

Предварительно отметим, предполагается, что все взаимодействия человека с виртуальной средой состоят из одних и тех же основных задач, из которых формируются более сложные сценарии.

Следовательно, если мы можем разделить трехмерные манипуляции на ряд таких основных задач, то вместо того, чтобы исследовать все пространство задач трехмерных манипуляций, мы сможем разработать и оценить методы взаимодействия только для этого небольшого подмножества. Затем результаты можно экстраполировать на все пространство трехмерных манипуляций.

Виртуальные 3D-манипуляции как имитируют движения в реальном мире, так и позволяют делать то, что невозможно в реальном мире, например, увеличивать или уменьшать объект.

Выделяют *следующие задачи* [26–29]:

– Выбор (или захват) – получение или идентификация определенного объекта или подмножества объектов из всего набора доступных объектов. Например, взять один или несколько объектов рукой или указать на них с помощью речи.

– Позиционирование – изменение трехмерного положения объекта (аналог – перемещение объекта из начального положения в целевое).

– Вращение – изменение ориентации объекта (аналог – поворот объекта из начальной ориентации в целевую).

– Масштабирование – изменение размера объекта (нет прямого аналога в реальном мире, но это обычная виртуальная манипуляция как для 2D, так и для 3D пользовательского интерфейса).

Каждая задача определяет *пространство задач*, которое включает несколько вариантов одной и той же задачи, определяемой параметрами задачи – переменными, которые влияют на производительность пользователя при выполнении этой задачи (табл. 1.4).

Классификация техник манипуляции [26, 27, 28]:

I. Изоморфизм в манипуляционных техниках.

Изоморфное представление предполагает строгое геометрическое однозначное соответствие между движениями рук в физическом и виртуальном мирах на том основании, что оно наиболее естественно и, следовательно, лучше для пользователей.

Однако оно также имеет важные недостатки:

1. Эти сопоставления часто непрактичны из-за ограничений в устройствах ввода. Например, диапазон слежения может быть ограничен.

2. Часто неэффективно из-за ограниченности человека. Например, длина нашей руки естественным образом ограничивает расстояние, на которое мы можем дотянуться.

Таблица 1.4

Параметры пространства задач

Задача	Параметры
Селекция	Направление и дистанция до цели, размеры цели, скученность объектов вокруг цели, количество селектируемых объектов, заслоненность цели
Позиционирование	Направление и расстояние до исходного положения, направление и расстояние до целевой позиции, дальность перемещения, требуемая точность позиционирования
Вращение	Дистанция до цели, исходная ориентация, конечная ориентация, количество вращений, требуемая точность вращения

3. Трехмерные пользовательские интерфейсы могут быть более эффективными, интуитивно понятными и богатыми, если вместо того, чтобы пытаться имитировать физическую реальность, мы создаем сопоставления и методы взаимодействия, специально адаптированные для трехмерной среды, обеспечивая в некотором смысле «лучшую» реальность.

Неизоморфный подход значительно отличается от строгого реализма, предоставляя пользователям «волшебные» виртуальные инструменты, такие как лазерные лучи, резиновые руки и другие. Эти отображения и методы могут позволить пользователям манипулировать объектами совершенно иначе, чем в физическом мире, сохраняя при этом удобство использования и производительность.

Какой метод использовать – зависит от строгого реализма задачи. Фактически, большинство методов прямого 3D-манипулирования сегодня неизоморфны.

II. Классификация по декомпозиции задач.

Все техники манипуляции состоят из одних и тех же основных компонентов, которые служат схожим целям. Например, в задаче выбора метод взаимодействия должен предоставлять пользователю средства для указания объекта для выбора и подтверждения выбора, а также обеспечения визуальной, тактильной или звуковой обратной связи при выполнении задачи (рис. 1.4). Аналогичным образом можно разложить задачи манипулирования и ротации.



Рис. 1.4. Классификация техник селекции [28]

Мы можем рассматривать методы 3D-манипуляции как построенные из нескольких компонентов, каждый из которых позволяет выполнить одну подоперацию. Весь «конструктор» может включать множество различных компонентов для каждой подоперации. Преимущество этого подхода состоит в том, что мы можем структурировать пространство проектирования методов взаимодействия так, чтобы новые методы можно было построить из базовых элементов, просто выбрав соответствующие компоненты и собрав их вместе.

III. Классификация по метафорам.

Методы манипулирования 3D UI – основа метафоры взаимодействия или комбинации метафор. Каждая формирует фундаментальную ментальную модель техники – перцептивное проявление того, что пользователи могут делать с помощью техник и того, чего они не могут.

Основанная на метафорах таксономия для классификации методов трехмерного манипулирования основана на шести общих метафорах, используемых для трехмерных манипуляций: захват, указание, взаимодействие с поверхностью, косвенное манипулирование объектами, бимануальные взаимодействия и комбинирование.

1. Метафора захвата. Самый естественный способ взаимодействия в трехмерном мире – хватать объекты и манипулировать ими руками.

Два различных подхода к обеспечению способности захватывать виртуальные объекты – методы, основанные на руках и методы на основе пальцев.

Техники захвата *на основе рук* используют упрощенный подход к реализации захватывающих взаимодействий, представляя виртуальную руку пользователя как единый эффектор (т. е. все вычисления взаимодействия основаны на единственном преобразовании, которое обычно соответствует положению с 6-ю степенями свободы физической руки пользователя).

Техника захвата *на основе пальцев* использует более реалистичный и гораздо более сложный подход к моделированию виртуальной руки пользователя в виде иерархии сегментов пальцев, которые обычно напрямую контролируются положением физических пальцев пользователя.

2. Метафора указания. Техника указания или наведения состоит в том, чтобы позволить пользователю легко выбирать объекты, расположенные за пределами досягаемости, и манипулировать ими, просто указывая на них. Когда вектор, определяемый направлением

наведения, пересекает виртуальный объект, пользователь может выбрать его, выдав триггерное событие, подтверждающее выбор.

Лучше, чем методы, основанные на хватании, поскольку для наведения требуется значительно меньше физических движений руки от пользователя. Недостаток – выразительные манипуляции с 6-ю степенями свободы с помощью базовых приемов наведения невозможны.

Однако указание, как правило, является очень плохой техникой позиционирования: манипулирование объектами может быть эффективно выполнено только радиальными движениями вокруг пользователя (перпендикулярно направлению указания), когда задача не требует изменения расстояния между пользователем и объектами. Вращения можно эффективно выполнять только вокруг одной оси – определяемой указательным вектором.

3. Поверхностные метафоры. Мультисенсорные поверхности – методы поверхностного 2D-взаимодействия и поверхностными 3D-методами взаимодействия.

4. Косвенные метафоры. Методы взаимодействия, которые позволяют пользователю манипулировать виртуальными объектами, не взаимодействуя с ними напрямую.

Есть несколько возможных причин для использования косвенных метафор:

- могут использоваться для взаимодействия с удаленными объектами без предварительного перемещения, аналогично использованию методов наведения;

- помогают избежать проблем с окклюзией, которые возникают из-за того, что части физического тела пользователя визуально блокируют взаимодействие;

- могут целенаправленно ограничивать манипуляции за счет уменьшения степени свободы манипуляционных задач для повышения эффективности и точности.

Можно разделить косвенные метафоры на три различных подхода: *управляемые пространства, прокси и виджеты.*

- Методы *пространства управления* отделяют управление от дисплея, позволяя пользователю взаимодействовать в физическом пространстве, отличном от видимого местоположения виртуальной среды, и отображать эти взаимодействия в среде.

- Методы *прокси* позволяют пользователю напрямую взаимодействовать с прокси (т. е. репрезентативными копиями объектов в виртуальной среде), а затем сопоставлять эти манипуляции с исходными объектами.

– Методы *виджетов* размещают в виртуальной среде виджеты, которыми пользователь может напрямую манипулировать. Эти виджеты, в свою очередь, влияют на свой контекст или связанные виртуальные объекты.

5. *Бимануальные метафоры*. Некоторые техники требуют обеих рук для взаимодействия, такие как «выбор воздушного шара», «мир в миниатюре» и др.

Выделяют четыре различных класса бимануальных взаимодействий:

- 1) симметрично-синхронные,
- 2) симметрично-асинхронные,
- 3) асимметрично-синхронные,
- 4) асимметрично-асинхронные.

Симметрично-синхронные взаимодействия предполагают, что каждая рука выполняет идентичные движения в одно и то же время, тогда как симметрично-асинхронные взаимодействия включают идентичные движения в разное время.

1.3.3. Навигация

Первоначально задача навигации определялась только как процесс перемещения через виртуальную среду. В дальнейшем в это определение включили процесс нахождения пути перемещения. Наконец, его расширили еще средствами и указателями для успешной навигации [26–28].

Навигация – процесс, при котором люди управляют своим движением, используя окружающие указатели и искусственные средства типа карт так, чтобы они могли достигать своих целей без возможности потеряться.

Перемещение является двигательным компонентом навигации – это задача перемещения из текущего местоположения в новое целевое местоположение или движение в желаемом направлении (при помощи ходьбы или транспортных средств).

В 3D-интерфейсах возможны перемещения в ограниченном объеме пространства с ограниченной скоростью, а также с ограничениями при движении на транспортном средстве из-за зрительно-вестибулярного несоответствия (возможна киберболезнь).

Важность техник взаимодействия при перемещении:

– Перемещения – это самая распространенная и универсальная задача взаимодействия в 3D-интерфейсах (иногда точка обзора пользователя остается неподвижной или движение автоматизировано).

– Перемещения часто поддерживают другую задачу, а не являются самоцелью, например, достижение в 3D-играх мест сражений и др. Это увеличивает потребность в удобстве движения для возможности сосредоточения пользователя на задаче, а не на управлении.

Поиск пути является когнитивным компонентом навигации. Это процесс определения маршрута между исходной точкой и пунктом назначения и следования по нему. Он включает в себя пространственное понимание и задачи планирования, такие как определение текущего местоположения в среде, определение пути от текущего местоположения к целевому местоположению и построение ментальной карты окружающей среды.

Трудности навигации в виртуальных средах [24]

1. Проблемы, связанные с определением пути в ВС, из-за того, что ВС вообще представляют менее сенсорные (визуальный, слуховой, локомоторный) подробности, чем среды реального мира (пространственных указателей и др.)

2. Сложности с определением пути в ВС и тем, что ВС имеют тенденцию изменяться быстрее, чем среды реального мира, что затрудняет узнавать такие среды.

3. Концепция навигации в ВС, которая может иметь полностью различную физику, чем навигация в физическом мире: движения и манипуляции могут не иметь никакого дубликата в реальном мире. Например, использование клавиатуры и мыши для перемещения и вращения, изменения местоположения и ракурса. Как следствие, посетители ВС будут сталкиваться с ситуациями, где они не будут способны положиться на умения навигации, приобретенные в реальном мире, поскольку знакомые виды сигналов неэффективны или несоответствующие.

4. Проблемы, связанные с дезориентацией, краткой потерей обзора, трудностями возврата к местоположению, посещенному ранее, или повторным нахождением объекта, найденный прежде, что может привести к неудовлетворенности, расстройству и, в конечном счете, к прерванным использованиям среды.

Типы задач 3D-перемещения [26–28]:

1. *Исследование или просмотр.* У пользователя нет явной цели для своего движения. Скорее, он просматривает окружающую среду, получает информацию об объектах и местах в мире, и накапливает знания о пространстве.

2. *Поиск*. Включают в себя перемещение пользователя к определенной цели или целевому местоположению в окружающей среде, которое он знает (хотя может и не знать).

3. *Маневрирование*. Маневрирования выполняются на определенной территории и предполагают небольшие точные движения. Чаще всего маневрирование используется для более точного позиционирования точки обзора в пределах ограниченной локальной области для выполнения конкретной задачи.

Дополнительные характеристики задач перемещения:

- расстояние, которое необходимо преодолеть;
- величина кривизны или количество поворотов на пути;
- видимость цели из начальной точки;
- количество степеней свободы, необходимое для движения;
- требуемая точность движения;
- другие основные задачи, выполняемые во время движения. (Часто движение – это второстепенная задача, выполняемая во время другой, более важной задачи, например, развешивание картин в галерее, что требует ненавязчивости, понятности и легкой управляемости).

Классификация 3D-перемещений [28]

- активные или пассивные;
- физические или виртуальные;
- на основе декомпозиции задачи;
- на основе метафор.

1. *Активные и пассивные техники*. Различают активные методы передвижения, в которых пользователь напрямую управляет движением точки обзора, и пассивные методы передвижения, при которых движение точки обзора контролируется системой.

2. *Физические методы против виртуальных*. Различают физическое перемещение, в котором тело пользователя физически перемещается или вращается, чтобы перемещать или вращать точку обзора, и виртуальное перемещение, в котором тело пользователя в основном остается неподвижным, даже если виртуальная точка обзора движется.

3. *Классификации с использованием декомпозиции задач*. Задачу перемещения разбивают на три подзадачи: выбор направления или цели, выбор скорости/ускорения и условия ввода (рис. 1.5).

Каждая подзадача может быть выполнена с использованием различных технических компонентов.

– Выбор направления или цели относится к основной подзадаче, в которой пользователь указывает, как двигаться или куда двигаться.

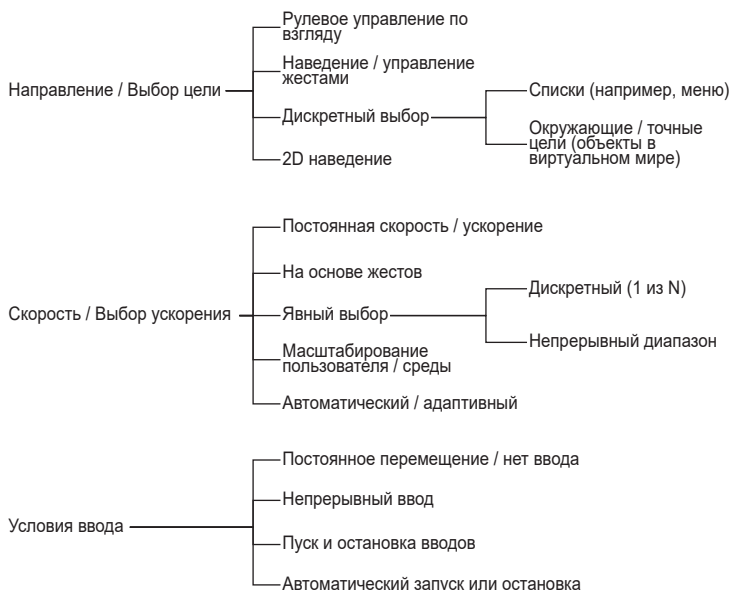


Рис. 1.5. Таксономия техник перемещения по задачам [28]

– Выбор скорости/ускорения описывает, как пользователи контролируют свою скорость.

– Условия ввода относятся к тому, как путешествие начинается, продолжается и прекращается.

Выбирая компонент техники для каждой из трех подзадач, мы можем определить полную технику перемещения.

Вторая декомпозиция задачи подразделяет задачу перемещения другим, более хронологическим образом (рис. 1.6). Чтобы выполнить задачу перемещения, пользователь сначала начинает движение, затем указывает положение и ориентацию, а затем прекращает движение.

Как показано на рис. 1.6, таксономия дополнительно разделяет подзадачу определения положения на подзадачи положения (координаты xuz), скорости и ускорения. Наконец, здесь перечисляются три возможные метафоры для подзадачи определения позиции: дискретная спецификация цели, одноразовая спецификация маршрута и непрерывная спецификация позиции. Эти метафоры различаются степенью контроля, который они дают пользователю над точным путем (различие между активным и пассивным).



Рис. 1.6. Таксономия техник перемещения по уровню пользовательского контроля [28]

4. *Классификация по метафоре.* Выделяют техники перемещения по четырем общим метафорам:

- ходьба,
- руление,
- выбор цели/маршрута,
- путем манипулирования.

Метафоры ходьбы

Самый естественный способ перемещаться в трехмерном мире – это физически ходить по нему. Но из-за технологических и пространственных ограничений реальная ходьба не всегда практична или осуществима.

Для преодоления разработали три категории метафор, основанных на походке человека:

- техника «полной походки», поскольку эти метафоры включают в себя всю биомеханику полного цикла походки;
- техника «частичной походки» – имитирует лишь некоторые биомеханические аспекты походки человека;
- методы «отрицания походки» предназначены для того, чтобы пользователь продолжал ходить в определенном пространстве, препятствуя его передвижению вперед.

Управляющие метафоры (вождения)

Одной из наиболее распространенных метафор является рулевое управление, которое означает непрерывный контроль направления движения пользователем.

Среди метафор вождения различают методы пространственного управления и управление при помощи муляжей рулей:

– Методы пространственного управления. Позволяют пользователю направлять или контролировать движение, манипулируя ориентацией устройства слежения. Просты для понимания и обеспечивают высочайший уровень контроля. Виды: рулевое управление, направленное взглядом, с ручным управлением (наведение), по направлению к туловищу, наклонное.

– Муляжи рулей управления. Специализированные устройства, предназначенные для выполнения задач рулевого управления. Полезны, когда моделируется определенный тип транспортного средства, когда интерфейс должен быть доступен любому человеку без какой-либо подготовки или когда рулевое управление является важной частью общего пользовательского опыта. Например, автомобильные рули и джойстики самолетов. Недостаток – возможные нереалистичные ожидания.

Метафоры перемещения, основанные на отборе

Такое перемещение зависит от того, выбирает ли пользователь цель, к которой он будет перемещаться, или путь, по которому он будет перемещаться.

Метафоры отбора упрощают перемещения, поскольку не требуют от пользователя постоянно думать о деталях. Вместо этого пользователь сначала указывает желаемые параметры перемещения, а затем позволяет технике перемещения позаботиться о фактическом перемещении. Хотя эти методы не самые естественные, они, как правило, чрезвычайно просты для понимания и использования.

Метафоры перемещения, основанные на манипуляциях

Эти методы используют метафоры ручного управления объектами для управления точкой обзора или всем миром. Следует использовать в ситуациях, когда задачи перемещения и манипулирования объектами выполняются часто и перемежаются. Например, задача размещения мебели, картин и т. п. в комнате (сочетание манипулирования и перемещения).

Поиск пути в трехмерной среде

Поиск пути (когнитивный аспект навигации) – это процесс принятия решений (рис. 1.7), в котором пользователь принимает решения («где я нахожусь?», «в каком направлении мне следует двигаться?»), мысленно обрабатывая «входные данные» (информацию, полученную из окружающей среды) и производя «выходные данные» (движение по траектории).

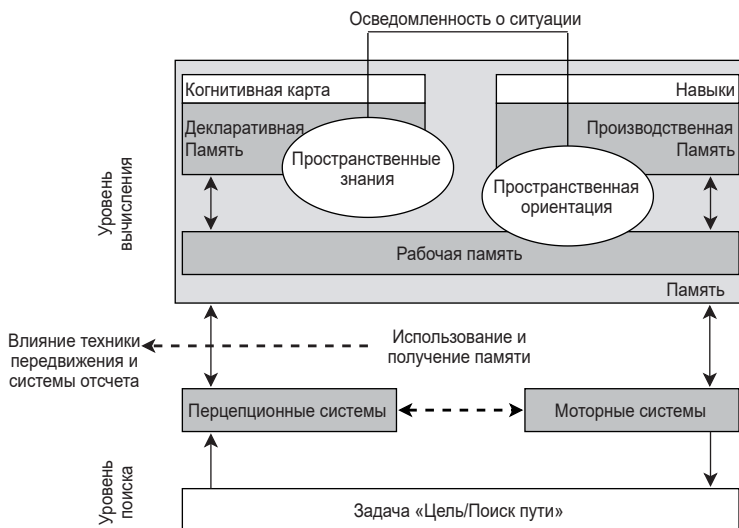


Рис. 1.7. Представление поиска пути (навигации) как процесса принятия решений [28]

Рассмотрим подробнее основные когнитивные проблемы, влияющие на поиск пути, включая когнитивное картографирование и различные виды пространственных знаний, влияющие на разработку ВДР [28].

Когнитивное картирование

Навигация в виртуальном 3D-пространстве включает в себя обработку множества источников сенсорной информации, которую мы получаем из окружающей среды, и использование этой информации для выполнения подходящей траектории движения. Информация об окружающей среде хранится в нашей долговременной памяти и обычно называется когнитивной картой, совокупностью пространственных знаний, которые мы получаем из окружающей среды.

Если быть более точным, когнитивная карта представляет собой ментальную иерархическую структуру информации, которая представляет пространственные знания. Когда пользователь выполняет задачу поиска пути, то он использует существующие пространственные знания, приобретает новые пространственные знания или использует комбинацию того и другого. Процесс доступа, использо-

вания и построения древовидных структур на когнитивной карте также называется когнитивным отображением. Навигация основана на обратной связи, которая непрерывно определяет взаимосвязь между информацией, которую воспринимает пользователь, и когнитивной картой окружающей среды, что позволяет понять местоположение и ориентацию. Знание местоположения и направления обзора называется пространственной ориентацией, в то время как сочетание пространственной ориентации и пространственных знаний (когнитивная карта) способствует осознанию ситуации.

Типы пространственных знаний

Стратегии поиска и параметры передвижения в виртуальном 3D-пространстве влияют на эффективность получения пространственных знаний. Эти факторы влияют не только на эффективность построения когнитивной карты, но и на качественно различные виды пространственных знаний, которые приобретаются. Во время поиска пути пользователь получает по крайней мере три различных вида пространственного знания.

Знание ориентиров состоит из визуальных характеристик окружающей среды. Визуально заметные объекты (ориентиры) являются частью этой информации, но другие визуальные особенности, такие как форма, размер и текстура, также играют определенную роль.

Процедурные знания (или знания о маршруте) описывают последовательность действий, необходимых для следования по определенному пути. Например, турист в Санкт-Петербурге быстро запомнит маршрут между своим отелем и ближайшей станцией метро.

Геодезические знания можно описать как топологические знания окружающей среды, состоящие из местоположений объектов, расстояний между объектами и ориентации объектов. Этот вид знаний подобен карте и, следовательно, может быть получен с карты. Из трех видов пространственных знаний геодезические знания представляют (качественно) самый высокий уровень знаний и, как правило, требует больше всего времени на мысленное построение.

Эффективность навигации зависит от количества и качества подсказок или вспомогательных средств, предоставляемых пользователям.

Подсказки разделяют на ориентированные на пользователя и ориентированные на среду подсказки.

Ориентированные на пользователя подсказки

Используют характеристики человеческого восприятия и могут опираться на несколько человеческих органов чувств.

Могут отрицательно влиять на навигацию из-за ограниченных возможностей человека.

Для уменьшения этих негативных эффектов используют различные ориентированные на пользователя подсказки для поиска пути:

- поле зрения,
- сигналы движения,
- мультисенсорный вывод,
- присутствие,
- стратегии поиска.

Ориентированные на окружающую среду подсказки

Относятся к сознательному дизайну виртуальной среды для поддержки поиска пути.

Большинство средств навигации для виртуальных сред могут быть напрямую связаны со средствами реального мира.

Они варьируются от естественных сигналов окружающей среды, таких как высокая гора, до искусственных сигналов, таких как карта.

Например, несколько ориентированных на окружение подсказок:

- понятность среды,
- ориентиры,
- карты,
- компасы,
- знаки,
- тропы,
- эталонные (справочные) объекты.

1.3.4. Управление системой 3D пользовательского интерфейса

Рассмотрим основные проблемы формирования управления системой 3D пользовательского интерфейса [26, 28].

Человеческий фактор

- Проблемы с восприятием:
- видимость, такая как окклюзия и разборчивость;
- переключение фокуса;
- выбор способа обратной связи.



Рис. 1.8. Классификация техник системы управления [28]

– Когнитивные проблемы:

– функциональная широта и глубина системы, структурирование задач для снижения когнитивных препятствий (когнитивный вызов вариантов доступа и мест их нахождения).

– Эргономические проблемы:

– размещение, а также поза, захват и типы движений, с которыми используется конкретное устройство;

– форма, размер и расположение элементов управления могут сильно повлиять на производительность управления системой.

Системные факторы

– При разработке интерфейсов управления системой важно учитывать высокоуровневые характеристики системы:

– характеристики системы могут диктовать конкретные варианты управления системой.

– Основные вопросы:

– устройства визуального отображения будут налагать определенные границы восприятия, такие как разрешение, размер и яркость;

– устройства ввода определяют возможности для управляющих отображений;

– факторы окружающей среды (например, шум), ограничения устройства или диапазон движения могут ограничивать выбор.

Классификация методов управления системой приведена на рис. 1.8.

1.4. Виртуальная реальность

При описании и создании ВР возникают *три фундаментальных вопроса*:

1. Проблема анализа и моделирования человеческой деятельности в реальной и виртуальной среде.

2. Проблема анализа, моделирования и создания интерфейса субъекта для его погружения и взаимодействия в виртуальной среде.

3. Проблема моделирования и создания виртуальной среды.

1.4.1. Подходы к определению виртуальной реальности

Рассмотрим некоторые подходы к определению ВР с точки зрения лучшего понимания ее свойств.

В [31] в общем случае термин *виртуальная реальность* «относится к любой ситуации, когда искусственно создается ощущение пребывания человека в определенной среде».

Нас, в первую очередь, интересуют ситуации, которые достигаются при помощи компьютерных средств. Для формирования ВР вводится понятие «генератор изображений» – прибор, способный создавать у пользователя точно определенные ощущения, а также репертуар генератора – набор сред, ощущение нахождения пользователя в которых может создать генератор.

Любой генератор ВР должен иметь три составляющие:

– набор сенсоров (которыми могут быть детекторы нервных импульсов), чтобы узнать о действиях пользователя;

– набор генераторов изображений (в роли которых могут выступать приборы стимуляции нервов);

– управляющий компьютер.

Предполагается, что в «конце концов, станет возможным обойти все органы чувств и оказать непосредственное воздействие на нервы, связывающие их с мозгом».

Кроме того, «среда может состоять из собственного тела пользователя, поскольку тело находится вне разума. Описание среды виртуальной реальности вполне может включать требование, о

том, чтобы тело пользователя казалось замещенным новым телом с определенными свойствами».

Отмечается, что ВР – это не просто технология моделирования поведения физических сред с помощью компьютеров. Возможность существования ВР – важная черта структуры реальности. Это основа не только вычислений, но и человеческого воображения, внешних ощущений, науки и математики, искусства и вымысла.

В [32] дается 52 *определения ВР*. Приведем несколько из них.

Четвертое определение VR: замена системы взаимодействия человека и физической среды системой взаимодействия человека со средой моделируемой.

Девятое определение VR: исследование сенсомоторного цикла, соединяющего людей с их миром, и способы его настройки инженерными методами. Это исследование бесконечно, потому что люди меняются.

Десятое определение VR: реальность с когнитивной точки зрения – это ожидание мозгом следующего момента. В виртуальной реальности мозг на какое-то время убеждает ожидать происходящего в иллюзорном мире, а не в реальном.

В [33] приводится следующее определение ВР: вызывание (индцирование) целевого поведения в организме с помощью искусственной сенсорной стимуляции, в то время как организм практически не осознает вмешательство.

В определении присутствуют четыре ключевых компонента:

1. *Целевое поведение*: организм получает «опыт», который был разработан проектировщиком. Например, полет, ходьба, исследование, просмотр фильма и общение с другими организмами.

2. *Организм*: это можете быть вы, кто-то другой или даже другая форма жизни. Например, ученые использовали подобные технологии на мухах, тараканах, рыбах, обезьянах и др.

3. *Искусственная сенсорная стимуляция*: с помощью инженерной мысли одно или несколько органов чувств организма перехватываются, а их обычные входные сигналы заменяются искусственной стимуляцией.

4. *Осведомленность*: во время переживания организм не осознает вмешательства, тем самым заставляя чувствовать себя присутствующим в виртуальном мире.

В [34] ВР определяется через сенсомоторику человека.

Основная задача систем виртуальной реальности – создать у пользователя ощущение реальности попадания в смоделированный на

компьютере мир, который должен быть не только реалистичен, но и позволять взаимодействовать человеку с его объектами так же, как с объектами, окружающими человека в обычной жизни – естественным и привычным путем, не требующим специального обучения. Для этого необходимо описать взаимодействие человека с виртуальной средой посредством сенсомоторных характеристик человека, основываясь на которых определяют виртуальную реальность как «*компьютерный интерфейс с перцептивными и моторными системами человека*». В настоящее время большинство систем виртуальной реальности основаны на стимуляции различных органов чувств. Однако для человека зрение является доминантной сенсорной системой. На биологическом уровне это подтверждается тем, что в обработке зрительной информации участвует почти половина коры головного мозга [24], а на поведенческом уровне доминирующая роль зрения подтверждается многочисленными экспериментами на сенсорный конфликт, в которых зрительная информация противоречит информации от какой-либо другой сенсорной системы.

1.4.2. Подход к разработке VR

Более подробно рассмотрим *подход к разработке VR*, описанный в [35], в котором приводятся определения VR на различных уровнях.

Цель виртуальной реальности – сделать возможной сенсомоторную и когнитивную деятельность человека (или людей) в созданном цифровым образом искусственном мире, который может быть воображаемым, символическим или имитацией определенных аспектов реального мира.

– Моделирование определенных аспектов реального мира: эти аспекты должны быть определены во время разработки приложения.

– Символический мир: мы также можем использовать символические представления, чтобы улучшить понимание симулируемого мира.

– Воображаемый мир: виртуальность используется для создания нереального мира, плода воображения художника или писателя-фантаста.

Функциональное определение

Виртуальная реальность поможет пользователю выйти из физической реальности, чтобы виртуально изменить время, место и/

или тип взаимодействия: взаимодействие со средой, имитирующей реальность, или взаимодействие с воображаемым или символическим миром.

Техническое определение

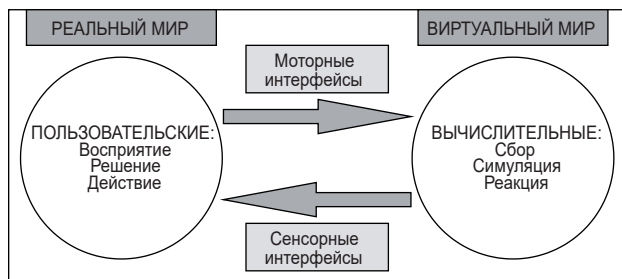
Виртуальная реальность – это научная и техническая область, которая использует информатику и поведенческие интерфейсы для моделирования в виртуальном мире поведения трехмерных объектов, которые взаимодействуют в реальном времени друг с другом и с одним или более пользователей в псевдо-естественном погружении через сенсомоторные каналы.

Термин «*поведенческие интерфейсы*» отражает использование (эксплуатацию) человеческого поведения и включает сенсорные (пользователь получает информацию о развитии виртуального мира через свои органы чувств), моторные (информируют компьютер о моторных действиях человека в виртуальном мире) и сенсомоторные интерфейсы (работают в обоих направлениях).

Пользователь должен находиться в максимально эффективном «псевдо-естественном погружении» в виртуальном мире. Погружение не может быть естественным, потому что мы научились действовать естественно в реальном мире, а не в виртуальном (создаются сенсомоторные смещения, отсюда и термин «псевдо»).

Из этого анализа мы можем вывести фундаментальный принцип виртуальной реальности. Этот принцип представлен в цикле, показанном на рис. 1.9.

Пользователь действует в виртуальной среде с помощью моторных интерфейсов, которые фиксируют его действия (жесты, движения, голос и т. д.). Эти действия передаются в вычислитель, который



*Рис. 1.9. Цикл «восприятие, решение, действие»
через виртуальный мир [35]*

интерпретирует их как запрос на изменение среды. В соответствии с этим запросом на модификацию, вычислитель оценивает изменения, которые необходимо внести в виртуальную среду, и сенсорные реакции (изображения, звук, эффекты и т. д.), которые будут переданы на сенсорные интерфейсы. Этот цикл в интерактивной виртуальной среде – всего лишь перенос цикла «восприятие, решение, действие» поведения человека в реальном мире.

Однако данный цикл нарушают *два основных ограничения*: задержка (латентность) и сенсомоторные несоответствия.

Задержка – это временной интервал между воздействием пользователя на моторные интерфейсы и восприятием последствий этого действия для виртуальной среды через сенсорные интерфейсы. Наличие задержки в цикле влияет на качество любого приложения виртуальной реальности. Эта задержка – артефакт, присущий интерактивным виртуальным средам.

Сенсомоторные несоответствия – еще один артефакт виртуальной реальности. Независимо от того, сколько сенсорных каналов используется в приложении и сколько взаимодействий предоставляется субъекту, сенсомоторные расхождения в отношении сенсомоторного поведения объекта в реальном мире почти всегда существуют.

После представления цели и технического определения ВР предлагается метод проектирования и оценки систем ВР, ориентированных на помощь человеку выполнять сенсомоторную и когнитивную деятельность в искусственном мире.

Предполагается, что перед переходом в виртуальный мир необходимо сначала определить поведение человека в реальном мире – стимул, орган чувств, нервный центр, исполнительные органы, движения.

Когнитивное изучение погружения и взаимодействия в виртуальную среду основано на подходе к деятельности субъекта. В основе системы лежит человек, поскольку виртуальное приложение предназначено для него. Это антропоцентрическое видение пользователя приложения. В абсолютном смысле можно схематически представить человека (рис. 1.10) полностью погруженным в искусственный мир, каким он должен воспринимать его.

Антропоцентрической диаграммы для разработки ВР недостаточно, так как ее необходимо разделить, и проанализировать взаимодействие с учетом используемых или проектируемых устройств. Такое изучение поведенческого взаимодействия осуществляется с помощью техноцентрической диаграммы.

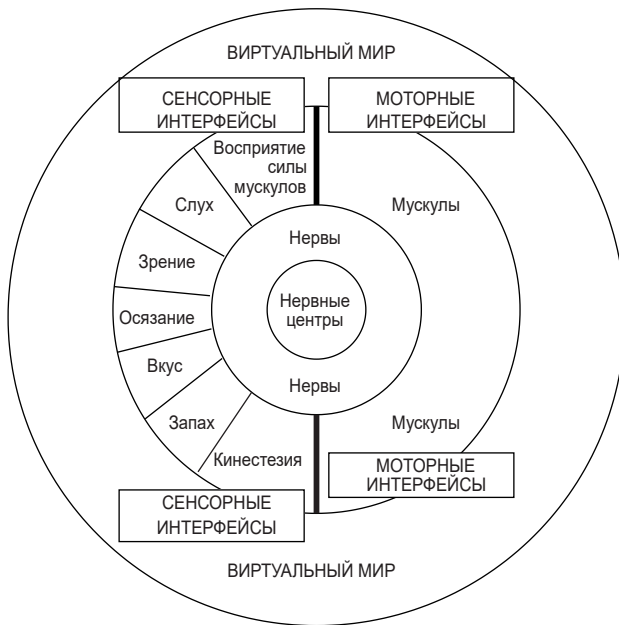


Рис. 1.10. Антропоцентрическая диаграмма видения виртуального мира человеком [35]

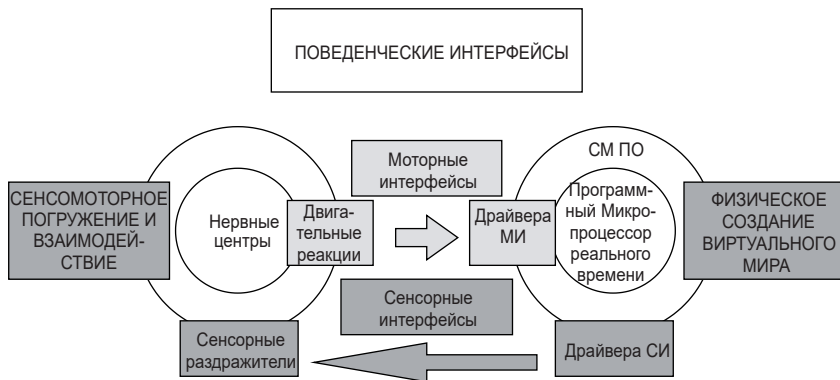


Рис. 1.11. Техноцентрическая диаграмма сенсомоторной иммерсии и интерактивности [35]

Далее можно представить *трехуровневую модель системы ВР*.

1. Первый уровень. На первом уровне – физическом уровне сенсомоторного погружения и взаимодействия используется техноцентрическая диаграмма (сенсомоторная петля между субъектом и компьютером, рис. 1.11).

Недостаток – на сенсомоторном уровне мы нарушаем цикл «восприятие, познание, действие» пользователя, добавляя артефакты (интерфейсы, их драйверы и один или несколько компьютеров, см. п. 1.1).

Эта диаграмма носит ограничительный характер, поскольку дает лишь частичное представление о проблемах и решениях, которые должен изучить каждый проектировщик.

2. Второй уровень. Поведенческий интерфейс создает проблему взаимодействия, аналогичную проблеме взаимодействия оператора со своим станком или инструментом.

В этом случае недостаточно думать только о физическом интерфейсе (кнопки управления и обратная связь по действиям). Не менее важно понимать ментальные модели, в соответствии с которыми человек будет думать и действовать. Эти интерфейсы зависят от артефактов (или инструментов), и пользователь управляет ими, используя свои когнитивные процессы.

Вопрос – какой познавательный (или нет) процесс использовал бы погруженный человек в этой ситуации?

В реальном мире все делается естественно. То же самое хотелось бы иметь в случае виртуальной среды.

Один из подходов к применению поведенческих интерфейсов, предлагающих псевдо-естественное погружение и интерактивность, основан на концепции психолога Пиаже [35]. Согласно ей, *схема (фрейм)* – это структурированный набор характеристик действия, которые можно обобщить, что помогает повторить действие или применить его к новому содержанию.

Таким образом, поведенческий интерфейс представляет собой смешанный объект, включающий как артефакт (его аппаратное устройство), так и схему, которую называют «*Импортированная поведенческая схема*» (ИПС). Эта схема импортируется из реальной среды для переноса и адаптации к виртуальной среде.

Отсюда вытекает *фундаментальный постулат виртуальной реальности*: в интерактивной виртуальной среде человек использует тот же подход, что и в реальном мире, для организации виртуального мира в соответствии с набором пространственно-временных и причинных правил.

В случае технических, экономических или теоретических трудностей, препятствующих использованию импортированной поведенческой схемы, можно обойти эти трудности, используя «*метафору*».

Вместо использования сенсомоторного поведения и знаний человека мы, в большинстве случаев, предлагаем ему визуальное символическое изображение действия или желаемого восприятия.

Например, в виртуальном магазине мы можем дать потребителю возможность подтвердить покупку продукта, просто щелкнув сначала на его изображение, а затем на значок, представляющий кассу. Это действие становится символическим и больше не представляет сенсомоторное действие в реальном магазине. То же самое касается метафоры «телепортации» из одного места в другое в виртуальной среде, оставаясь при этом неподвижным в реальной среде.

Использование метафоры может потребовать дополнительных когнитивных усилий, если метафорические символы не известны пользователям. Им нужно приложить усилия, чтобы понять и усвоить символ, чтобы он постепенно превратился в схему использования. Это же касается ИПС, так как она должна быть адаптирована к виртуальному миру с определенным артефактом и с учетом сенсомоторных несоответствий.

Мы используем либо метафору, либо ИПС – в зависимости от психологических, технических и экономических трудностей и планируемого применения.

На практике мы можем сочетать когнитивное погружение и взаимодействие (КПВ), и метафоры в зависимости от типа требуемых действий, что определяет второй уровень модели системы ВР.

3. Третий уровень. На третьем уровне цель состоит в том, чтобы попытаться погрузить человека в заданную функциональность (или задачу).

В данном случае мы говорим о функциональном погружении и взаимодействии. Это разделение помогает нам лучше прояснить различные проблемы, с которыми сталкиваются при погружении и взаимодействии предмета. Они тесно связаны, а не противоположны.

Рассмотрим пример. Допустим, мы держимся за предмет. Мы можем использовать схему захвата, используя трекер с шестью степенями свободы и экран, отображающий объект (сенсомоторная иммерсивность и интерактивность – СИИ). Но если СИИ плохого качества (большое время отклика между действием с трекером и ощущением движения на экране), схему захвата использовать нельзя.

С другой стороны, если интерфейс не использует схему захвата, когнитивная иммерсивность и интерактивность выйдет из строя даже при точном функционировании интерфейса.

Таким образом, в основе подхода лежит *иерархическая трехуровневая модель системы ВР* с разделением между человеком и виртуальным миром: параллельно с различными уровнями сенсомоторной и когнитивной иммерсивности и интерактивности для человека у нас есть два уровня функционирования программного обеспечения для виртуального мира.

Компьютер должен управлять программной частью (реальное время и драйверы для аппаратных интерфейсов) симметрично СИИ, чтобы физически создавать виртуальный мир.

На уровне функциональной иммерсивности и интерактивности по отношению к приложению и его целям необходимо анализировать действия, которые должен выполнять пользователь.

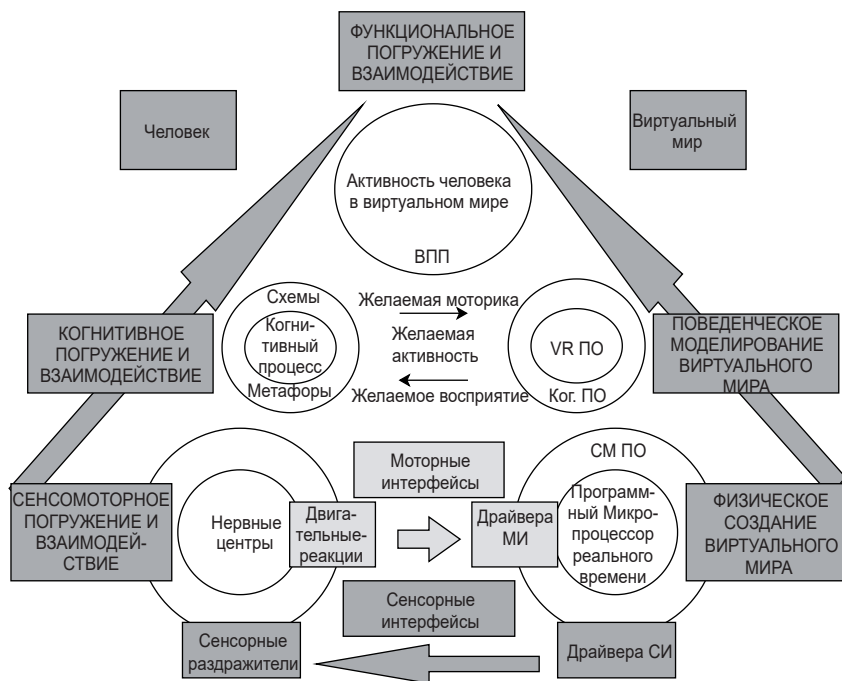


Рис. 1.12. Техноцентрическая эталонная трехуровневая модель системы ВР [35]

В принципе, во всех приложениях ВР действия человека всегда можно разделить на некоторые базовые модели поведения, которые можно назвать «*виртуальными поведенческими примитивами*».

Независимо от области применения их можно разделить на четыре категории:

- 1) наблюдение за виртуальным миром;
- 2) перемещение в виртуальном мире;
- 3) действие в виртуальном мире;
- 4) общение с другими людьми или с приложением.

Эталонная диаграмма системы виртуальной реальности приведена на рис. 1.12. Ее можно использовать в качестве основы для проектирования.

1.4.3. Присутствие, погружение, интерактивность

Последним рассмотрим определение ВР на основе понятий «*присутствие*», «*погружение*» и «*интерактивность*» [36].

Виртуальная реальность – разновидность человеко-компьютерного интерфейса, обеспечивающего присутствие человека в моделируемой среде посредством поддержки его погружения и взаимодействия с объектами этой среды в реальном времени.

Пример структуры системы ВР приведен на рис. 1.13 [24].

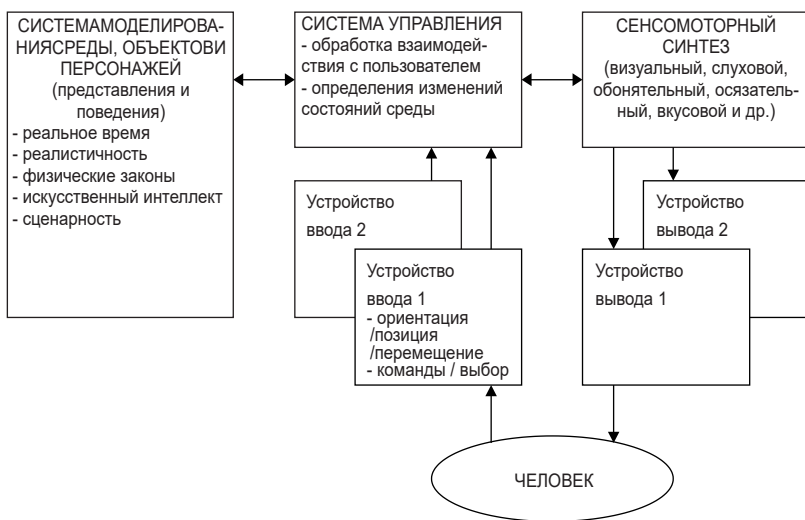


Рис. 1.13. Структура системы формирования и поддержки ВР [24]

Укрупненно, система ВР состоит из следующих основных подсистем:

– *Интерфейс пользователя* – обеспечивает отображение состояний среды; регистрацию действий человека, производимых с помощью моторики (эффекторов), и воздействие на модальности человека в соответствии с состоянием среды.

– *Подсистема управления* – включает анализ выполненных субъектом действий; формирование реакции на действия субъекта; определение изменений состояний среды во времени и пространстве.

– *Подсистема моделирования среды, объектов и персонажей* – формирует их реалистичное представление (геометрию и иерархию, физические свойства, свойства материалов) и имитирует их поведение во времени и пространстве в реальном времени на базе физических моделей или моделей искусственного интеллекта.

1.4.4. Основные классификации ВР

Рассмотрим основные классификации ВР [24]:

1. *По виду интерактивности* – полетные и реактивные, соответственно позволяют передвигаться в трехмерной среде с шестью степенями свободы без ограничений или предоставляют возможность взаимодействовать с объектами с реагированием.

2. *По уровням погружения* – экран стандартного монитора, стереоскопический монитор или проектор в комплекте с очками, полное погружение.

3. *По особенностям организации в пространстве* – локальные и распределенные, когда, соответственно, все ресурсы реализующей среды сосредоточены практически в одном месте, из-за чего передача данных по линиям связи не используется, или имеется совокупность сред, физически распределенная по взаимосвязанным ресурсам реализующей среды и доступная для совместного использования в различных приложениях.

4. *По способу реализации* – настольные, проекционные, носимые, тренажеры.

– *Настольные (Desktop VR)* – монитор служит как «окно» в виртуальный мир; для взаимодействия могут использоваться стереочки и 3D-устройства ввода.

– *Проекционные (Projection VR)* – модели проецируются на различные поверхности, используя, например, такие технологии как Walls, CAVE и др.

– Носимые (Wearable VR) – используется прямое взаимодействие частей тела пользователя с виртуальной моделью или средой через устройства типа шлемов, перчаток, костюмов, систем тактильной обратной связи, платформ движения.

– Тренажеры (Simulator VR) – используются физические макеты транспортных средств с реальным управлением (руль, педали и т. д.), при помощи которых можно перемещаться в виртуальной среде.

5. По режиму обслуживания – однопользовательские и многопользовательские. Соответственно, мир предоставляется полностью в распоряжение пользователя, по крайней мере на время решения его задачи, или возможен одновременный доступ нескольких независимых географически распределенных пользователей к одному миру.

1.5. Дополненная реальность и дополненная виртуальность

1.5.1. Определения дополненной реальности

Рассмотрим с разных точек зрения несколько определений ДР.

Система дополненной реальности (Augmented Reality) дополняет реальный мир виртуальными (компьютерно-генерируемыми) объектами, которые сосуществуют и взаимодействуют в одном и том же пространстве, как в реальном мире [37, 38, 39].

В [40] даны следующие определения:

– *Дополненная реальность* – тип системы смешанной реальности, в которой данные виртуального мира встроены и/или зарегистрированы (совмещены) с представлением данных физического мира.

– *Система дополненной виртуальности* – тип системы смешанной реальности, в которой данные физического мира внедрены и/или зарегистрированы с представлением данных виртуального мира.

– *Дополнение* – данные виртуального объекта (созданные компьютером, синтетические), добавленные или связанные с данными целевого физического объекта (видео в реальном времени, изображение реального мира) в сцене смешанной и дополненной реальности (Mixed and Augmented Reality – MAR).

– *Сцена СДР* – наблюдаемая пространственно-временная организация физических и виртуальных объектов, которая является результатом представления сцены СДР, интерпретируемого меха-

низмом выполнения СДР и имеющего по крайней мере один физический и один виртуальный объект.

Также в [36] дается следующее определение: дополненная реальность (ДР) дополняет различные ощущения (зрение, слух, осязание и др.) реального мира цифровыми (виртуальными) объектами, которые сосуществуют и взаимодействуют в одном и том же пространстве, как в реальном мире.

Данная технология дополняет, например, поле зрения пользователя при помощи компьютерных устройств соответствующей информацией, что позволяет ему получать со своей точки зрения и в соответствии с индивидуальными настройками необходимые для работы сведения о наблюдаемых объектах и помощь в решении поставленных задач.

Некоторые приложения дополненной реальности устраняют реальные объекты из воспринимаемой среды. Например, при визуализации строящегося здания, которое стоит на определенном месте, может понадобиться удалить строительные леса, краны и т. п., которые существуют в настоящий момент. Некоторые исследователи называют подобную задачу устранения реальных объектов *уменьшенной реальностью*.

В настоящее время большинство систем расширенной реальности реализуется с использованием «живого» цифрового видео, которое обрабатывается и дополняется компьютерно-генерируемыми изображениями.

Однако данные определения дополненной реальности не ограничиваются только визуальным каналом восприятия человека, но могут включать различные другие его *сенсомоторные характеристики*: слух, обоняние, осязание, тактильно-силовые взаимодействия и др.

Свойства систем дополненной реальности

Системы дополненной реальности имеют следующие свойства [38]:

- комбинируют реальные и виртуальные объекты в реальной среде;
- работают в интерактивном режиме в реальном времени;
- реалистично совмещают (регистрируют) реальные и виртуальные объекты друг с другом.

Существует *три* основные технологии, на которых строится ДР [41]:

1. *Отслеживание (Tracking)* – определение параметров местоположения и ориентации, система должна знать точку зрения поль-

зователя для поиска и представления связанного с ней цифрового контента.

2. *Регистрация (Registration)* – совмещение реальной и цифровой информации, которая предоставляется пользователю. Обычно проводится с пиксельной точностью в кадре для представления реальных и виртуальных объектов в одной и той же области.

3. *Отображение (Display)* – наблюдение пользователем реального мира с наложенной цифровой информацией на устройстве с учетом отслеживаемости объектов в каждом кадре.

Технологии распознавания дополненной реальности

Существуют следующие технологии распознавания:

1. Маркерная (коды или изображения):

– идентификация выбранных для представления объектов происходит при помощи меток, размещенных на них или рядом с ними, или на основе распознавания естественных особенностей объекта;

– при наведении устройства на метку происходит ее распознавание, после чего пользователь может ознакомиться с привязанной к этой метке информацией или просмотреть симуляцию события поверх вида объекта.

2. *Безмаркерная* – пространственное картирование обеспечивает представление реальных поверхностей окружающей среды видимыми на них полигональными сетками с возможностью привязки к произвольным элементам сетки цифровых объектов, которые наблюдаются через камеру выбранного для реализации дополненной реальности устройства [42]. Пользователь может привязать любую информацию к элементам сетки.

3. Геолокационная:

– при наведении устройства на определенный сектор пространства на его экране появятся метки, привязанные к выбранным для представления и попадающим в этот сектор объектам города или природной среды, с которыми связана некоторая информация или смоделированная ситуация;

– активировав метку, пользователь сможет ознакомиться с этой информацией или просмотреть симуляцию события поверх вида объекта.

Наибольшее распространение на сегодняшний день получила *визуальная дополненная реальность* – это технология наложения в режиме реального времени цифровых объектов (текст, графика, аудио и др.) на изображение реальной среды, полученное с камеры какого-либо устройства и наблюдаемое на его экране.

Основная *цель систем визуальной дополненной реальности* – реалистичное включение в реальное изображение, поступающее из видеокамер, виртуальных объектов в реальном времени, что требует решения следующих типовых задач [43]:

- обработка и распознавание изображений объектов среды в реальном времени;

- синтез и визуализация виртуальных объектов. В зависимости от сценария приложения необходима различная информация, например, для пространственной навигации необходима информация о попадающих в поле зрения объектах, для градостроительства – возможный вид здания при его ремонте, для военных операций – характеристики вооружения противника и т. п.;

- синхронизация между объектами реального мира и виртуальными объектами. Перед тем, как объединить синтезированное изображение с объектами реального мира, трехмерные параметры реальной и синтезированной камеры должны быть синхронизированы. Для этого реальная камера должна быть калибрована, и ее местоположение и ориентация должны быть отслеживаемы при помощи специальных сенсоров;

- определение и визуализация трехмерных геометрических и других зависимостей между физическими и виртуальными объектами, в частности, определение места виртуального объекта в сцене и вычисление в каждом кадре видимых с данной позиции камеры точек виртуальных и реальных объектов (пересечения виртуальных и реальных объектов). Реалистичное слияние реального мира и виртуальных объектов достигается только в том случае, если виртуальные объекты ведут себя подобно объектам реального мира: они могут заслонять объекты реального мира или сами заслоняться ими, не могут проходить сквозь реальные объекты, должны отбрасывать тени от реальных источников света и изменять освещенность своих элементов при попадании их в тень реальных объектов;

- взаимодействие с пользователем. Необходимы новые методы и средства взаимодействия на основе жестов, голоса, тактильного управления и т. п.

В [44] рассматриваются *системы отображения пространственной дополненной реальности* – «это формирующие изображения системы, которые применяют ряд оптических, электронных и механических компонентов для создания изображений на оптическом пути между глазом наблюдателя и физическим объектом, который в результате этого будет расширен (дополнен)».

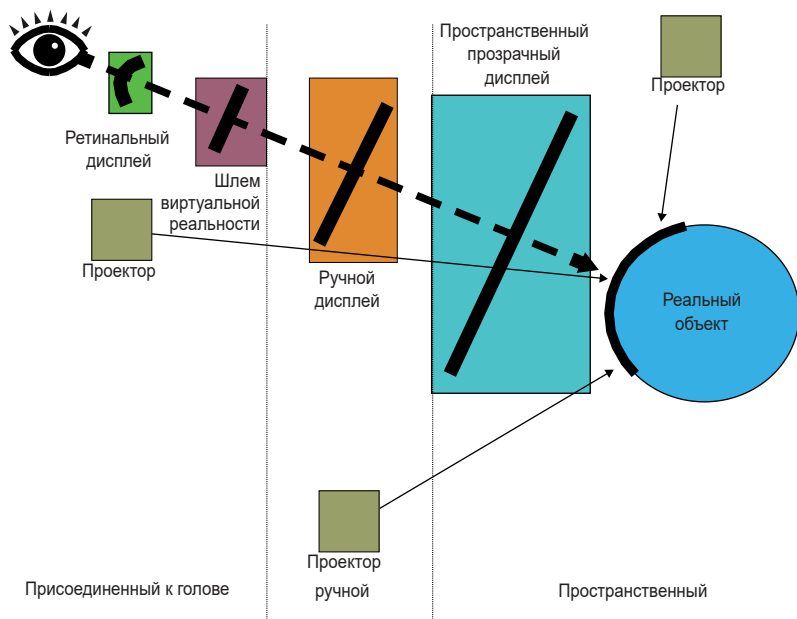


Рис. 1.14. Схема генерации изображений для ДР

На рис. 1.14 приведены различные варианты создания изображений с учетом расположения наблюдателя и реального объекта.

1.5.2. Классификация устройств визуального отображения

Приведем классификацию устройств визуального отображения [38, 44].

1. Дисплеи, закрепленные на голове. В зависимости от технологии генерации изображений существует два основных вида подобных систем:

Ретинальные дисплеи (Retinal displays) – применяют лазеры малой мощности для проецирования изображений непосредственно на сетчатку глаза.

Дисплеи, закрепленные на голове (Head-Mounted Displays), и очки, которые используют миниатюрные дисплеи перед глазами.

Существуют две разные HMD-технологии, используемые для наложения графики на пользовательское видение реального мира.

Видеопрозрачность (video see-through) – использует видеосмешивание и отображение слитного изображения внутри HMD (рис. 1.15).

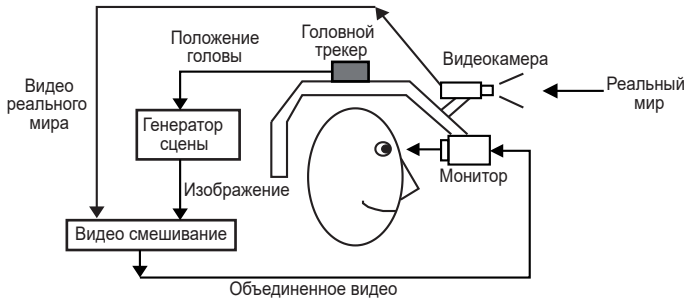


Рис. 1.15. Схема устройства видеопрозрачного шлема

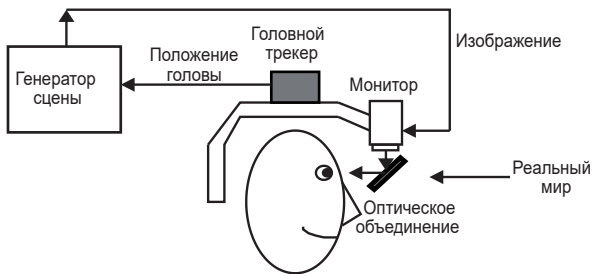


Рис. 1.16. Схема устройства оптически прозрачного шлема

Оптическая прозрачность (optical see-through) – использует комбинации оптических наложений (в основном прозрачные зеркала или ЖК-дисплеи) (рис. 1.16).

2. *Ручные дисплеи.* Примеры «ручных» отображений, позволяющих генерировать изображения «на руку» – планшеты и смартфоны. Перед отображением в видеокамере интегрируется захваченный живой видеопоток от среды с наложенной дополнительной графикой (подход на основе видеопрозрачности).

3. *Пространственные дисплеи.* В отличие от дисплеев, связанных с телом (головные или ручные), пространственные дисплеи отсоединяют большую часть техники от пользователей и интегрируют ее в среду. Существуют три различных подхода, которые в основном отличаются способом увеличения среды – с помощью видеопрозрачности, оптической прозрачности или путем прямого расширения:

а) видеопрозрачные отображения, базирующиеся на экранах (мониторах). Используют видеосмешивание (видеопрозрачность) и отображают объединенные изображения на монитор (рис. 1.17);

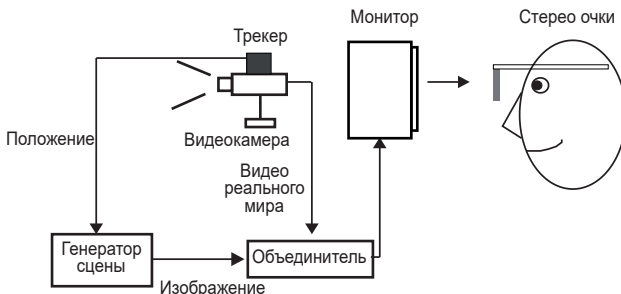


Рис. 1.17. Схема работы экранной системы

б) пространственные оптические прозрачные отображения. В отличие от видеопрозрачных отображений (прикрепленных к голове или ручных), пространственные оптические прозрачные отображения создают образы, которые совмещены в физической среде. Пространственные оптические конфигурации как, например, плоские или изогнутые зеркала, прозрачные экраны или оптические голограммы, являются важнейшими компонентами таких отображений;

с) пространственные отображения, основанные на проекциях. Используют фронтпроекции для бесшовного проецирования изображений непосредственно на поверхности физических объектов вместо отображения их на плоскость изображения (или поверхность) в пределах поля зрения пользователя. Единый стационарный большой экран и несколько проекторов применяются для увеличения потенциальной области отображения.

1.5.3. Критерии анализа ДР

В [45] приведены несколько критериев анализа ДР, а именно: предприятие, вычисления, информация. Определение каждой точки обзора приведено в табл. 1.5.

Таблица 1.5

Точки обзора ДР

Точка зрения	Определение	Темы, освещаемые MAR- RM
Пред- при- ятия	Формулирует «бизнес-модель», которая должна быть понятна всем заинтересованным сторонам. Основное внимание уделяется цели, сфере охвата и	– Актеры и их роли, – Возможные бизнес- модели для каждого участника,

Точка зрения	Определение	Темы, освещаемые MAR- RM
	политике, а также целям различных участников, вовлеченных в эту область	– Желательные характеристики для участников на обоих концах цепочки (создатели и пользователи)
Вычислительная	Определяет функциональные возможности компонентов системы и интерфейсы между ними, службы и протоколы, которые каждый компонент предоставляет среде	– Услуги, предоставляемые каждым основным компонентом AR, – Описание интерфейсов для вариантов использования
Информационная	Обеспечивает семантику информации в различных компонентах цепочки, общую структуру и абстрактный тип контента, а также источники информации. Также описывает, как информация обрабатывается внутри каждого компонента. Не обеспечивает полную семантику и синтаксис данных, а лишь минимальный набор функциональных элементов, и ее следует использовать в качестве руководства для разработчика приложения или разработчика стандарта при создании их собственных информационных структур	– Контекстная информация, такая как пространственная регистрация, захваченное видео и аудио. – Информация о контенте, такая как виртуальные объекты, поведение приложения, управление взаимодействием с пользователем. – Служебная информация, такая как удаленная обработка контекстных данных

1.5.4. Пример архитектуры системы ДР

Пример архитектуры системы ДР приведен на рис. 1.18 [46].

Рассмотрим назначение компонентов системы:

Устройства (датчики) – аппаратное «устройство», которое измеряет физическую величину и преобразует ее в сигнал, пригодный для прочтения другим модулем.

Модуль распознавания – программный модуль, который принимает необработанные данные сенсорного устройства и создает «события», соответствующие описанию, данному в спецификации контента, с тем же идентификатором.

Модуль отслеживания – программно-аппаратный модуль, который принимает необработанные данные сенсорного устройства и

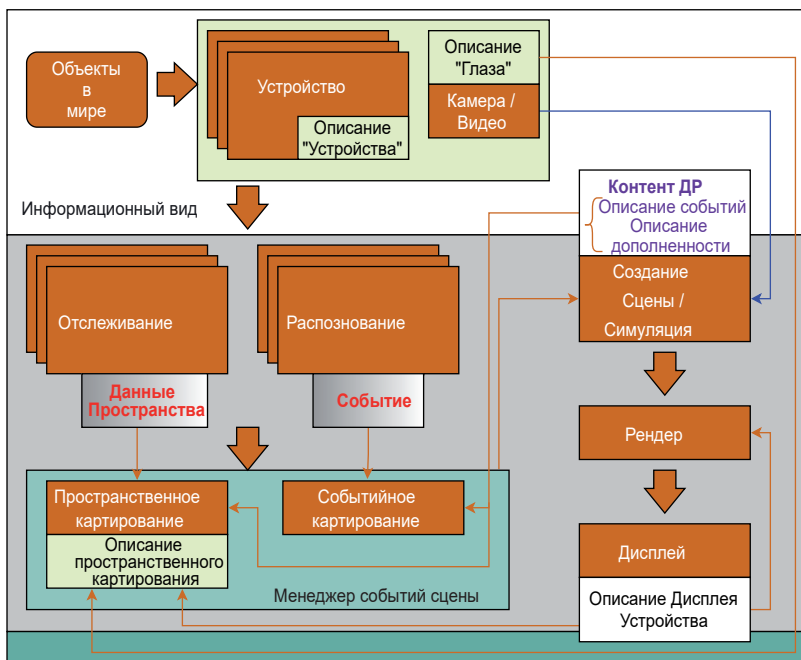


Рис. 1.18. Архитектура ДР [46]

определяет положение и ориентацию целевого физического объекта или объекта, который определяется описанием события из спецификации контента.

Менеджер событий сцены – программный модуль, который принимает внешние события и моделирует поведение сцены и соответственно динамически обновляет описание сцены AR (поведение динамической AR-сцены указано в описании контента; составлен с помощью Event Mapper и Spatial Mapper).

Картирование событий – программный модуль, который передает событие, созданное модулем «Распознавание», в «Менеджер событий сцены AR». Он также анализирует описание события и позволяет модулю распознавания понять, какое событие следует распознать для указанного содержимого (события определены в данной спецификации контента).

Пространственное картирование – программный модуль, который передает данные отслеживания, созданные модулем «Отслеживание», в «Менеджер событий сцены AR». Он также анализи-

рует описание события и позволяет модулю отслеживания понять, какое событие следует распознать и какой объект следует отслеживать для указанного содержимого. Он также берет описание внешней камеры/видео и отображает его характеристики в виртуальную камеру в сцене.

MAR сцена контента/исполнительная платформа – динамическая иерархическая структура данных, описывающая виртуальную сцену. Для целей **AR** сцена контента представляет собой традиционный граф сцены для виртуального мира с добавлениями: событий **AR**, сенсорного устройства **AR**, камеры **AR**, возможностей отображения **AR**.

Модуль камеры/глаза – захватывающее устройство реального мира – это аппаратное/программное обеспечение, создающее видеопоток (и другой «визуальный» формат данных) для встраивания в сцену дополненной реальности.

Рендерер (визуализатор) – берет граф сцены и создает мультимедальный сигнал рендеринга (визуальный, слуховой и тактильный).

Отображение – аппаратное устройство, которое отображает сцену в различных модальностях (визуальных, слуховых и тактильных). Связано с описанием, в котором указывается его тип, важные параметры и значения, касающиеся его возможностей.

1.5.5. Хранение информации о точках интереса

Одной из важнейших проблем для систем ДР является хранение информации о *точках интереса*. Существует *три основных типа архитектур*, на которых строятся системы ДР: сетевая (web), автономная (standalone) и шлюзовая (gateway) [47].

Сетевая модель подразумевает, что у системы есть неограниченный доступ в интернет и вся информация хранится в глобальной сети. Соответственно, есть возможность доступа к неограниченному объему контента, что накладывает на разработчика дополнительные заботы по контролю мировой модели – информации, связанной с точками интереса. Например, необходимо тщательно отфильтровать права доступа к ресурсам, чтобы пользователь получал только достоверную информацию. Также хранение данных в сети значительно расширяет набор устройств, на которых можно реализовать подобную систему.

Использование **автономной модели** основано на применении сетевых технологий. Вся система должна функционировать без доступа в глобальную сеть, что и является главным преимуществом

данной архитектуры. Среди отрицательных сторон можно выделить необходимость постоянно производить выпуск новых версий приложения с обновленной информацией.

В **шлюзовой модели** информация о статусе пользователя и окружении переносится на платформу, т. е. платформа берет на себя все основные настройки внешнего вида приложения, фильтры точек интереса и т. д. В свою очередь поставщик контента настраивает приложение под свои нужды и размещает на своих серверах контент, на который ссылается через платформу, которая служит шлюзом. Преимуществом данной модели является то, что поставщику контента нет необходимости заботиться о создании собственного пользовательского интерфейса с нуля, его необходимо просто настроить. Таким образом, более широкий круг поставщиков может присоединиться к развитию платформы, что положительным образом скажется на количестве и качестве контента.

Каждая архитектура имеет свои положительные и отрицательные стороны. Необходимо исходить из конкретных условий применения приложения для выявления наиболее подходящей архитектуры. Сетевая архитектура подойдет для использования в приложениях с разнообразным и постоянно меняющимся контентом. Автономная архитектура должна использоваться, когда ограничен или отсутствует доступ к сети. Шлюзовая архитектура представляет собой компромиссный вариант.

1.5.6. Дополненная виртуальность

Вторым компонентом смешанной реальности является *дополненная виртуальность* (Augmented Virtuality) [48], которая означает слияние объектов реального мира с виртуальным (цифровым) миром. Например, физические объекты или люди динамически интегрируются в виртуальный мир и взаимодействуют с ним в режиме реального времени.

Такая интеграция осуществляется с использованием различных методов, например, потоковое видео из физического пространства, или с использованием трехмерной оцифровки физических объектов, или драпировки виртуальных объектов, или использование информации реальных датчиков, например, гироскопов, когда внешние входные данные обеспечивают контекст для виртуального представления.

В качестве примера можно также привести устройство HTC VIVE Tracker [49], которое будучи закрепленным на реальном объ-

екте может управлять поведением его цифровой модели в виртуальной среде. Подробнее оно рассмотрено в разделе 2.2.

1.6. Виртуальные и смешанные миры

1.6.1. Виртуальный или смешанный мир

Виртуальный или смешанный мир (ВСМ) – это компьютерная среда онлайн-сообщества, разработанная и совместно используемая отдельными людьми и предоставляющая им возможность взаимодействовать в специально созданном, смоделированном мире [50]. Можно сказать, что ВСМ – это сетевая виртуальная или смешанная реальность.

Пользователи взаимодействуют друг с другом в этом симулированном мире, используя текстовые, двухмерные или трехмерные графические модели, называемые аватарами.

Люди управляют своими аватарами с помощью устройств ввода, таких как клавиатура, мышь, а также более сложных систем – шлемов с контроллерами, сенсоров захвата рук, смартфонов и т. п. (например, HTC Vive, Oculus Rift, Microsoft Kinect, Leap Motion и др.). Посредством аватаров люди могут одновременно и независимо исследовать виртуальные миры, создавать новые, выполнять какую-либо работу и общаться между собой.

Также в настоящее время для виртуальных миров используется термин «*социальная виртуальная реальность*» [51].

1.6.2. Анализ многопользовательских ВСМ

В общем случае анализ многопользовательских ВСМ производится на основе следующих трех основных критериев:

1. Пространственная метафора. Для значимой визуализации и взаимодействий могут быть использованы различные метафоры, связанные с реальным или вымышленным миром. Однако, вероятно, что при использовании метафор реального мира познавательная нагрузка пользователя снижается, увеличивая степень сосредоточения и осознания пользователем задачи изучения. Например, при дистанционном обучении используются метафоры классов, комнат для частных встреч и лекционных аудиторий.

2. Представление пользователя. Определяет присутствие и тождество пользователей в ВСМ в любое время. Пользователи ВСМ должны быть видимы себе и другим пользователям. Для представления пользователя необходимо решить следующие основные проблемы:

– присутствие – первичная цель аватара состоит в том, чтобы передать присутствие пользователя в виртуальном мире, его местоположение и ориентацию;

– тождество – идентификация пользователя по аватару, представляющего его;

– точки зрения – передача разных позиций восприятия пользователя при поддержке взаимодействия;

– готовность – может возникнуть необходимость передать готовность к взаимодействию (присутствие аватара предполагает готовность пользователя, хотя это может не всегда иметь место);

– знак – поддержка знаков в коммуникации подразумевает необходимость наличия у аватаров выражающих их средств – руки, ноги, лицо и др.;

– мультимодальное представление – пользовательское представление должно рассматриваться не только как визуальное, но и давать возможность использовать другие средства коммуникации, например, голос и текст;

– эффективность – аватары должны быть настолько простыми, насколько это возможно;

– правдивость – насколько аватар должен соответствовать реальному человеку.

3. Коммуникация между пользователями. ВСМ должны предоставить возможность подражать реальному миру в поддержке визуальной, голосовой, текстовой и невербальной (жесты, телодвижения, мимика лица) коммуникации в режиме реального времени.

В качестве примеров социальных виртуальных миров можно привести SecondLife, AltSpaceVR, vTime, VRChat, Engage и др. [52].

1.6.3. Проекты Microsoft и Facebook

Отдельно стоит отметить проекты компаний Microsoft и Facebook.

Microsoft Mesh – сеть, позволяющая совместно использовать пространство и совместно работать в смешанной реальности друг с другом в любой точке мира. За счет этого можно улучшить виртуальные встречи, сеансы разработки, обучение, удаленную помощь другим пользователям и др. [53].

Метавселенная Facebook – это обширная сеть устойчивых, визуализируемых в реальном времени трехмерных миров и симуляций, которые непрерывно поддерживают идентичность объектов, историй, платежей и прав, и могут быть испытаны синхронно практи-

чески неограниченным количеством пользователей, каждый с индивидуальным чувством присутствия» [54, 55].

Сегодня специалисты технологической индустрии говорят о метавселенной, удовлетворяющей следующие основные требованиям [55]:

- Наборы функций, которые частично совпадают со старыми веб-службами или реальными действиями.

- 3D-графика в реальном времени и персонализированные аватары.

- Разнообразные индивидуальные социальные взаимодействия, менее конкурентные и целенаправленные, чем стереотипные игры.

- Поддержка пользователей, создающих свои собственные виртуальные элементы и среды.

- Связь с внешними экономическими системами, чтобы люди могли получать прибыль от виртуальных товаров.

- Дизайн, который кажется хорошо подходящим для гарнитур виртуальной и дополненной реальности, даже если они обычно поддерживают и другое оборудование.

Однако, в большинстве современных дискуссий метавселенная, возможно, не является фиксированным набором атрибутов. Это желанный термин для обозначения будущего цифрового мира, который более ощутимо связан с нашей реальной жизнью и телом.

Руководители компаний Epic и Facebook говорят, что они просто строят одну часть более крупной взаимосвязанной метавселенной, похожей на отдельную социальную сеть в современном Интернете, т. е. метавселенная – это не единственный продукт, который компания может создать в одиночку. Как и в Интернете, метавселенная существует независимо от того, есть там Facebook или нет.

Но неформально термин «метавселенная» также используется для описания единой платформы, которая соответствует критериям, перечисленным выше. Second Life – виртуальный мир, который не является традиционной игрой, часто описывается как метавселенная. Пользовательский опыт Fortnite можно также описать как метавселенную, потому что это виртуальное трехмерное пространство, в котором смешаны игровые и неигровые элементы.

В настоящее время подобные миры ориентированы не только на общение и развлечение, но и на решение задач обучения, тренажа, презентации, проведения совещаний, конференций и др.

1.7. Мультимодальный интерфейс

1.7.1. Мультимодальное взаимодействие в виртуальной среде

Человек в реальном мире использует свои сенсомоторные системы для взаимодействия с окружающей средой и общения с другими людьми, применяя для этого различные вербальные (язык, речь) и невербальные (мимика, жесты, запахи, касания и др.) средства. В виртуальной среде для этого используется мультимодальный интерфейс.

В области человеко-машинных интерфейсов [56–59] выделяют следующие определения:

– **Модальность** – тип канала коммуникации, используемого для взаимодействия.

– **Мультимодальная система (Multimodal System)** – поддерживает коммуникацию с пользователем через различные модальности, например, такие как голос, жест и печатание.

– **Мультимодальное взаимодействие (Multimodal Interaction)** – взаимодействие пользователя с приложением с использованием более чем одного сенсомоторного канала коммуникации.

Основное различие между мультимодальной системой и мультимедийной системой состоит в том, что «мультимедийная система позволяет приобретать, хранить и распределять данные, в то время как мультимодальная система способна к приобретению и интерпретации данных, так же как к хранению и распределению этих интерпретаций. Поэтому можно сказать, что мультимодальная система – это система с мультимедийными способностями, которая позволяет производить семантическую обработку данных».

Можно перечислить следующие преимущества мультимодального взаимодействия человека с машиной:

– более естественное и выразительное взаимодействие, позволяющее пользователю работать с компьютером аналогично тому, как люди взаимодействуют друг с другом естественным образом – с помощью речи, взгляда, мимики, жестов, поз, касаний и т. п.;

– повышение эффективности выполнения задач (время, усилия, обработка ошибок), в том числе повышение точности при решении пространственно-визуальных задач.

– поддержка наиболее предпочитаемого пользователем способа взаимодействия;

– снижение требований к интерфейсным знаниям пользователя.

Тем самым это обеспечивает для сложных человеко-машинных систем расширение не только доступности, привлечения более широкой аудитории, но и ситуаций использования.

1.7.2. Архитектура мультимодального интерфейса

Обобщенная архитектура мультимодального интерфейса представлена на рис. 1.19.

Основными ее подсистемами являются:

1. *Ввод* – реализация мультимодального взаимодействия с использованием нескольких режимов ввода, таких как речь, жесты, рукопись, текст, движения и др. Включает три компонента:

1) распознавание – воспринимает естественный ввод от пользователя и переводит его в форму, пригодную для дальнейшей обработки; например, переводит речь в текст, жесты в перемещение объекта в сцене и т. п.;

2) интерпретации – идентифицирует «значение» или «семантику», подразумеваемую пользователем; например, многие слова с одинаковой семантикой «да», «конечно», «согласен» могут быть представлены как «да»;

3) интеграции (мультимодальное слияние) – объединяет все входящие унимодальные события в единое представление намерения, выраженное пользователем. Некоторые или все возможности этого компонента могут быть обеспечены как часть компонента распознавания, интерпретации или взаимодействия. Например, аудиовизуальное распознавание речи может интегрировать распознавание

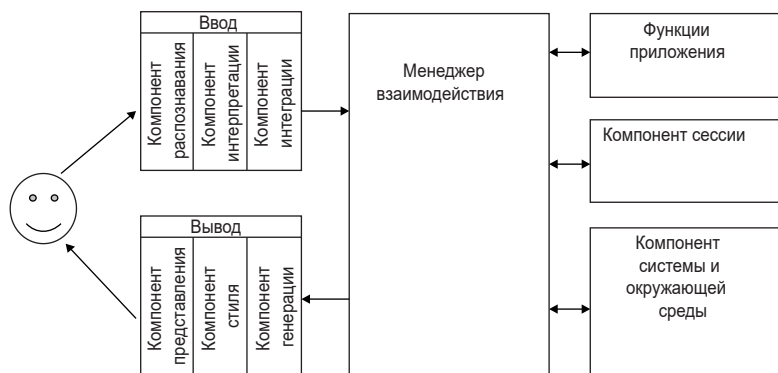


Рис. 1.19. Обобщенная архитектура мультимодального интерфейса

движения губ и распознавание речи как часть компонента чтения губ, как часть компонента распознавания речи или объединяться в отдельный компонент интеграции.

Существует *три основных подхода* для управления вводом в порядке увеличения сложности [60]:

а) последовательный мультимодальный ввод – соответствует входным данным, полученным от одной модальности, которая может меняться с течением времени. На данном уровне должно быть доступно описание, какая модальность или устройство будет использоваться для ввода и указания или осуществления переключения модальностей;

б) синхронный мультимодальный ввод – предполагается, что входные данные от нескольких модальностей интерпретируются одно за другим в порядке приема, вместо того чтобы комбинироваться перед интерпретацией;

в) смешанный мультимодальный ввод – соответствует входным данным, полученным от нескольких модальностей в одно и то же время и рассмотренных как единый, интегрированный комплекс введенных данных для последующей обработки;

2. *Вывод* – реализация мультимодального взаимодействия с использованием нескольких режимов вывода, таких как речь, текст, графика, аудио файлы и анимация. Также включает три компонента. Ими являются:

а) генерация (мультимодальное разделение) – определяет, какой способ вывода будет использоваться для представления информации пользователю. Анализ различий для активации наиболее подходящей модальности и построение сообщения через их комбинацию. Информация от менеджера взаимодействия может быть передана непосредственно на соответствующее устройство представления, не будучи закодированной на внутреннем языке представления. Например, записанное аудио посылают непосредственно в звуковую систему;

б) стиль – добавляет информацию о том, как информация должна представляться. Например, компонент стиля для дисплея описывает, как графические объекты располагаются на экране, в то время как компонент стиля для звука может вставлять паузы и интонации голоса в текст, который произносится синтезатором речи;

в) представление (рендеринг) – конвертирует информацию из компонента стиля в формат, который легко понять пользователю. Например, компонент графического представления отобразит век-

тор точек как кривую линию, а система синтеза речи конвертирует текст в синтезированную речь.

3. *Менеджер взаимодействия* – это логический компонент, который координирует данные и управляет потоками из различных входных и выходных модальностей. Он поддерживает состояние взаимодействия в контексте приложения, отвечает на входы от составляющих интерфейс объектов и изменения в системе и окружающей среде, управляет этими изменениями и координирует вход и выход через составляющие интерфейс объекты.

4. *Компонент сессии* – предоставляет интерфейс для менеджера взаимодействия, чтобы поддерживать управление состояниями, а также временными и постоянными сессиями для мультимодальных приложений. Особенно важен для распределенных приложений, включающих более одного устройства и/или пользователя. Он скрывает подробности используемых схем и протоколов и обеспечивает интерфейс высокого уровня для запроса или освобождения ресурсов внутри сессии.

5. *Компонент системы и окружающей среды* – позволяет менеджеру взаимодействия обнаруживать возможные изменения устройств, параметров пользователя, состояние окружающей среды и реагировать на них. Например, какой из доступных режимов предпочитает пользователь, разрешение экрана, возможности поддержки цвета и др.

1.7.3. Концептуальные модели мультимодального интерфейса

Существуют различные *концептуальные модели* мультимодального интерфейса, которые с разных точек зрения позволяют описывать, классифицировать и оценивать различные типы связей между взаимодействующими модальностями. Рассмотрим некоторые из них.

Подход **TYCOON Framework** (types of cooperation) позволяет описывать и оценивать различные типы связей между взаимодействующими модальностями [61]. Он основан на представлении типов и целей взаимосвязи между модальностями. В результате исследований в таких областях, как психология, искусственный интеллект, человеко-машинное взаимодействие, были выявлены пять типов взаимосвязей между модальностями:

1. Трансформация: устанавливает, какая часть информации, порожденная одной модальностью, будет использоваться другой модальностью. Трансформация может происходить как между дву-

мя входными или выходными модальностями, так и между входной и выходной модальностью.

Цели такого типа взаимосвязи:

– *перевод* – например, в гипермедиа интерфейсах клик мышью генерирует отображение рисунка или поиск информации в приложении, пользователь может выразить запрос в одной модальности (например, речь) и получить соответствующую информацию в другой (например, видео);

– *улучшение распознавания* – например, результат указания мышью может быть перенесен в модальность речи для облегчения распознавания предсказуемых слов (например, «здесь», «тот»);

– *возможность быстрого взаимодействия* – например, часть произнесенного высказывания была не распознана, она может быть отредактирована с помощью клавиатуры, при этом пользователю не нужно вводить/произносить все высказывание сначала.

2. Эквивалентность: две модальности считаются эквивалентными, если часть информации может быть обработана как альтернатива любой из модальностей.

Цели такого типа взаимосвязи:

– *улучшение распознавания команд* – например, когда механизм распознавания речи не работает эффективно (например, наличие шума в окружающей среде), пользователь может выбрать действие с помощью стилуса;

– *адаптация к характеристикам пользователя* – пользователь может выбрать предпочитаемую модальность.

3. Специализация: указывает на определенный тип информации, который всегда обрабатывается одной и той же модальностью.

Цели такого типа взаимосвязи:

– *интерпретация* – пользователь помогает интерпретировать события, порожденные системой;

– *улучшение распознавания* – облегчает процесс обработки и улучшает точность распознавания речи из-за уменьшения области поиска;

– *ускорение взаимодействия* – уменьшает продолжительность интеграции и процесса выбора модальности.

4. Избыточность: некоторые модальности взаимодействуют избыточно, когда они обрабатывают одну и ту же информацию (например, отображение подтверждающего диалога скрывается с помощью двух избыточных действий пользователя: вводом «прекратить» и произношением «прекратить», это способствует уско-

рению взаимодействия). Наблюдаются некоторые преимущества избыточности:

– *поддержка естественных для пользователя действий* – исследования показали, что иногда пользователи выбирают некоторые опции (например, город), одновременно используя речь и прикосновение к сенсорному экрану;

– *увеличение обучаемости* – избыточный мультимодальный вывод с одновременным отображением текста и его произношением позволяет ускорить процесс обучения по сравнению с графическим интерфейсом.

5. Комплементарность: рассматриваются несколько модальностей, каждая из которых обрабатывает различные части информации, которые впоследствии объединяются.

Цели такого типа взаимосвязи:

– *ускорение взаимодействия* – две модальности могут использоваться одновременно и передавать короткие сообщения, распознаваемые лучше, чем длинные;

– *улучшение интерпретации* – для продвинутых пользователей достаточно графического отображения, но для новичков необходимо также отображение текста.

2. АППАРАТНО-ПРОГРАММНЫЕ СРЕДСТВА ВИРТУАЛЬНОЙ И ДОПОЛНЕННОЙ РЕАЛЬНОСТИ

В данном разделе будут рассмотрены устройства и технологии, позволяющие пользователю работать с виртуальной, дополненной и смешанной реальностями. К ним относятся популярные комплекты виртуальной реальности VIVE Pro и Oculus Quest 2, треке-ры дополненной виртуальности VIVE Tracker 2.0 и интерфейс разработки ARCore, позволяющий создавать приложения дополненной реальности для мобильных устройств. В качестве программной платформы разработки интерактивных приложений с поддержкой перечисленных выше устройств выбран популярный игровой движок Unity. Для изучения дисциплины «Разработка виртуальной и дополненной реальности» обучающемуся необходимо иметь некоторый опыт работы с Unity. Сюда можно отнести такие навыки, как создание и редактирование объектов, компоновка сцен, применение и настройка материалов, установка освещения, работа с персонажем и контроллером персонажа, работа с элементами пользовательского интерфейса и компиляция (сборка) проекта в исполняемый файл для выбранной целевой платформы. Подробнее ознакомиться с базовыми инструментами и техниками Unity можно в учебно-методическом пособии «Основы разработки интерактивных 3D-приложений на движке Unity» [62].

2.1. Знакомство с комплектом VR-устройств VIVE Pro

Шлем виртуальной реальности VIVE Pro был представлен 8 января 2018 года и пришел на смену оригинальной модели HTC Vive. Официальные поставки в магазины начались 5 апреля того же года, а цена шлема составила \$799 [63]. Новая версия получила дисплеи с высоким разрешением (1440×1600 на каждый глаз), вторую фронтальную камеру, отключаемые наушники с сертификацией Hi-Res и микрофон с функцией шумоподавления. Также была улучшена эргономика шлема, добавлена поддержка 4 базовых станций и внесены многочисленные программные улучшения. Несколько позже, 2 июля 2018 года, разработчик анонсировал скорый выпуск комплекта виртуальной реальности VIVE Pro Full Kit, куда помимо шлема также вошли две обновленные базовые станции Lighthouse 2.0 и два контроллера HTC VIVE образца 2018 года. Цена предварительного заказа на комплект составила J1299.00 [64]. Фото шлема, контроллеров, базовых станций и хаба VIVE Pro представлено на рис. 2.1.



Рис. 2.1. Основные компоненты VR-системы VIVE Pro

Рассмотрим подробнее комплект VIVE Pro Full Kit, его комплектацию, технические характеристики, системные требования и принцип работы [65].

2.1.1. Комплект поставки и системные требования

В комплект системы виртуальной реальности VIVE Pro Full Kit входят:

- базовые станции SteamVR 2.0 Lighthouse – две штуки;
- адаптер питания базовых станций – две штуки;
- настенный монтажный комплект для базовых станций – две штуки;
- коммуникационный модуль Link Box (Хаб);
- кабель USB 3.0 – одна штука; кабель Micro-USB – две штуки;
- кабель DisplayPort;
- адаптер питания коммуникационного модуля;
- монтажная пластина коммуникационного модуля;
- шлем с присоединенным кабелем;
- чистящая салфетка;
- накладка для наушника – две штуки;
- контроллер (образца 2018 г.) с ремешком на руку – две штуки;
- адаптер питания для зарядки контроллера – две штуки.

Таблица 2.1

Минимальные системные требования VIVE Pro

Элемент	Требование
Центральный процессор	Intel Core i5-4590 / AMD FX 8350 (эквивалент или лучше)
Видеокарта	Nvidia GeForce GTX 970 / AMD Radeon R9 290 Наличие видеовыхода DisplayPort 1.2
Оперативная память	Не менее 4 ГБ DDR3
Интерфейсный порт	1 порт стандарта USB 3.0 или более новой модели
Операционная система	Microsoft Windows 7 (с последними обновлениями драйверов), Windows 8.1, Windows 10

Примечание: для максимального раскрытия возможностей VR производитель рекомендует использовать видеокарту не слабее Nvidia GeForce GTX 1070 или AMD Radeon Vega 56.

Минимальные системные требования к ПК для работы с комплектом VIVE Pro приведены в табл. 2.1.

2.1.2. Устройство и технические характеристики

Шлем – главный компонент системы VIVE Pro Full Kit, позволяющий пользователю погрузиться в виртуальную реальность. Благодаря массиву ИК-датчиков шлем умеет определять свое положение и ориентацию в пространстве относительно базовых станций, что обеспечивает точное позиционирование пользователя в виртуальной среде. Технические характеристики шлема VIVE Pro представлены в табл. 2.2.

Таблица 2.2

Спецификации шлема VIVE Pro

Элемент	Характеристики
Размер и тип экрана	AMOLED с диагональю 3.5» – две штуки
Разрешение экрана	1440×1600 пикселей на каждый глаз (суммарно 2880×1600 пикселей)
Частота обновления	90 Гц
Угол обзора	110 градусов
Выход звука	Съемные наушники, сертифицированные Hi-Res. Поддержка высокоомных наушников
Вход звука	Встроенный микрофон
Вход видео	Две фронтальные камеры

Элемент	Характеристики
Соединение и коммуникация	<ul style="list-style-type: none"> – USB-C 3.0 – DisplayPort 1.2 – Bluetooth
Датчики	Система отслеживания SteamVR Tracking <ul style="list-style-type: none"> – Акселерометр – Гироскоп – Датчик приближения – Сенсор IPD для подстройки межзрачкового расстояния
Эргономика	<ul style="list-style-type: none"> – Подстройка расстояния от глаз до линз – Подстройка межзрачкового расстояния – Подстройка наушников – Подстройка оголовья

Схематичное изображение шлема VIVE Pro с подписанными элементами конструкции представлено на рис. 2.2.

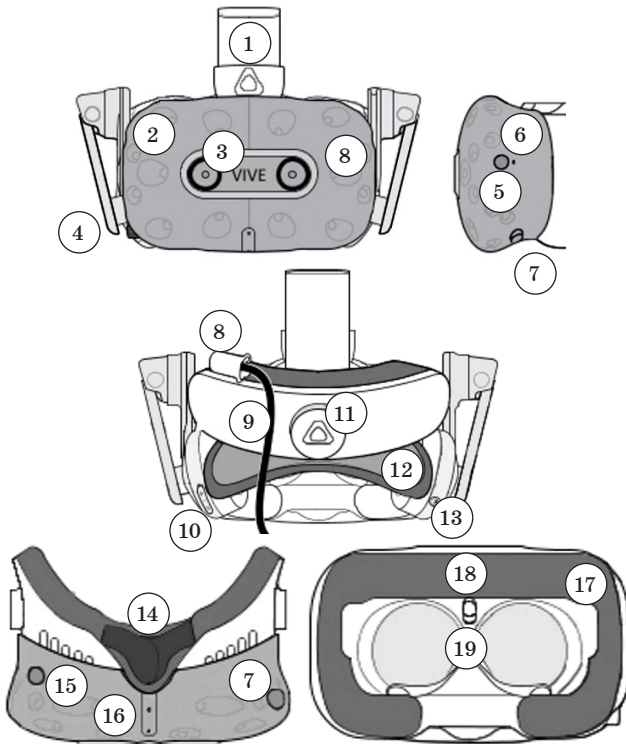


Рис. 2.2. Схематичное изображение шлема VIVE Pro

Цифрами на рисунке обозначены следующие элементы: 1 – ремешок шлема; 2 – датчики отслеживания; 3 – объективы фронтальных камер; 4 – наушники; 5 – кнопка шлема; 6 – индикатор состояния; 7 – ручка регулировки межлинзового расстояния; 8 – фиксатор кабеля на шлеме; 9 – кабель шлема; 10 – кнопка регулировки громкости; 11 – поворотный регулятор; 12 – подкладка для головы; 13 – кнопка отключения микрофона; 14 – накладка для носа; 15 – кнопка регулировки расстояния между лицом и окулярами; 16 – микрофон; 17 – лицевая накладка; 18 – датчик приближения; 19 – окуляры.

Наушники отстегиваются. При желании пользователь может заменить их на другие. Также к шлему можно подключить внешнее USB-устройство при помощи скрытого разъема USB-C.

Контроллеры VIVE 2.0 (2018)

Контроллеры используются для взаимодействия пользователя с виртуальными объектами, заменяя собой клавиатуру, мышь или геймпад, а также отслеживают положение и ориентацию рук в пространстве за счет встроенных трекеров. Технические характеристики контроллеров VIVE образца 2018 года представлены в табл. 2.3.

Схематичное изображение контроллера с подписанными органами управления и элементами конструкции представлено на рис. 2.3.

Цифрами на рисунке обозначены следующие элементы: 1 – кнопка «Меню»; 2 – сенсорная панель (TrackPad); 3 – кнопка «Система»; 4 – индикатор состояния; 5 – разъем micro-USB; 6 – датчики отслеживания положения; 7 – курок (Trigger); 8 – кнопка «Захват».

Таблица 2.3

Спецификации контроллера VIVE

Элемент	Характеристики
Технология трекинга	SteamVR 2.0
Элементы ввода	<ul style="list-style-type: none"> – Многофункциональный трекпад – Две кнопки захвата – Кнопка-триггер с двойным ходом – Системная кнопка – Кнопка вызова меню
Подключение к ПК и зарядка	Разъем micro-USB
Питание	Встроенный аккумулятор емкостью 960 мАч

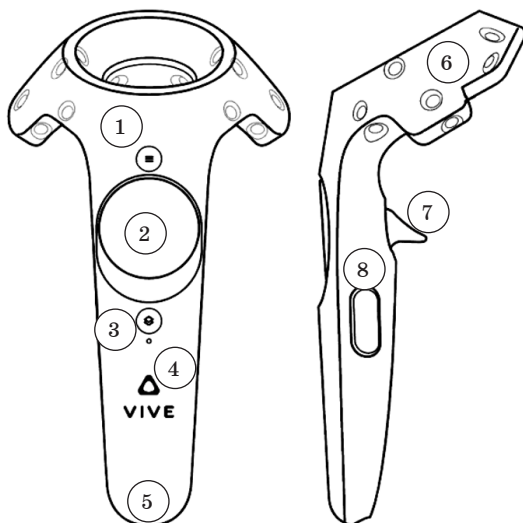


Рис. 2.3. Схематичное изображение контроллера VIVE версии 2.0

Базовые станции Lighthouse 2.0 (2018)

Базовые станции являются своего рода маяками (откуда и название Lighthouse – «маяк» по-английски), относительно которых шлем, контроллеры и трекары определяют свое положение и ориентацию в пространстве. Технические характеристики базовых станций Lighthouse 2.0 представлены в табл. 2.4.

Таблица 2.4

Спецификации базовой станции Lighthouse 2.0

Элемент	Характеристики
Технология трекинга	SteamVR 2.0 (многоосный лазерный излучатель с лазером класса 3B)
Стабилизация положения	Встроенный гиростабилизатор
Коммуникация	Bluetooth 2.4 ГГц с модуляцией GFSK
Зона охвата	150° по горизонтали 110° по вертикали
Частота импульсов	От 250 Гц до 1000 Гц
Крепление штатива	Отверстие с резьбой 1/4 дюйма
Интерфейсы	Разъем micro-USB для синхронизации – Разъем питания – Кнопка сброса – Индикатор состояния

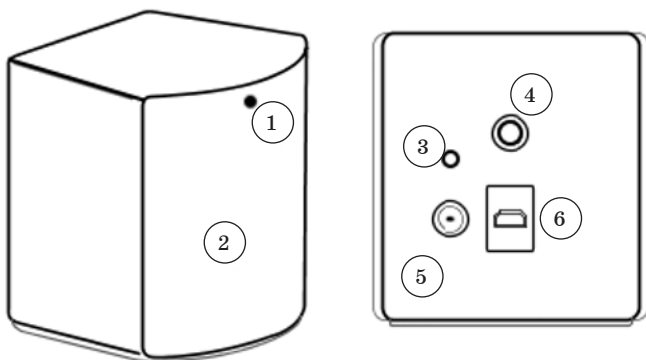


Рис. 2.4. Изображение базовой станции SteamVR Lighthouse 2.0

Схематичное изображение станции SteamVR Lighthouse версии 2.0 представлено на рис. 2.4.

Цифрами на рисунке обозначены следующие элементы: 1 – индикатор состояния; 2 – лицевая панель, за которой находятся ИК-датчик и гироскоп; 3 – утопленная в корпус кнопка сброса (Reset); 4 – монтажное отверстие с резьбой 1/2 дюйма; 5 – разъем для подключения адаптера питания; 6 – разъем Micro-USB для синхронизации.

Коммуникационный модуль Link Box

Коммуникационный модуль Link Box (также называемый хабом) используется для подключения шлема к персональному компьютеру через интерфейсы USB и DisplayPort. Схематичное изображение хаба с подписанными интерфейсами приведено на рис. 2.5.

Цифрами на рисунке обозначены следующие элементы: 1 – порт для кабеля шлема; 2 – кнопка питания; 3 – порт питания; 4 – порт DisplayPort 1.2; 5 – порт USB 3.0.

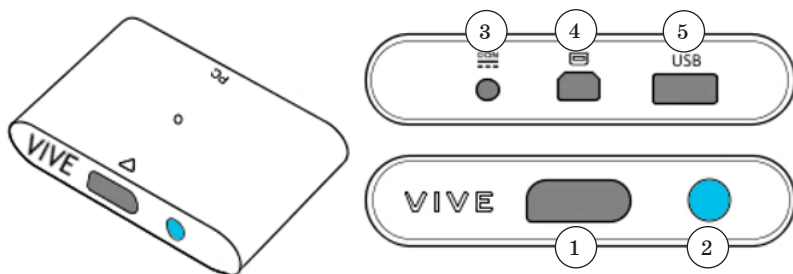


Рис. 2.5. Изображение коммуникационного модуля VIVE Pro

2.1.3. Устройство и способ подключение базовой станции

Базовая станция представляет собой ИК-излучатель с гироскопом для стабилизации положения в пространстве и механизмом синхронизации с соседними станциями и шлемом VIVE. «Общение» со шлемом и другими станциями происходит по протоколу Bluetooth.

Система виртуальной реальности VIVE Pro использует метод трекинга (отслеживания положения в пространстве) называемый Outside-In («снаружи-вовнутрь»). Его можно писать так: базовые станции с трекингом SteamVR отправляют многочисленные импульсы синхронизации и сканируют комнату лазерными лучами в радиусе до 5 м [66]. Система трекинга SteamVR тщательно отслеживает временные интервалы между импульсами и лазерными лучами и при помощи тригонометрии обнаруживает местоположение каждого датчика на отслеживаемом объекте с субмиллиметровой точностью. Датчики на объектах (шлеме, контроллерах или трекере) представляют собой легковесные, маломощные, экономичные сенсоры со специализированной интегральной микросхемой и встроенным гиростабилизатором, который позволяет системе SteamVR рассчитывать направление, скорость и угловое вращение отслеживаемых объектов с частотой от 250 Гц до 1 кГц. Объединяя данные с датчиков и станций, система точно определяет положение шлема, контроллеров и трекеров в поле видимости.

Для качественной работы шлема VIVE Pro и обеспечения трекинга с охватом в 360° требуется наличие как минимум двух базовых станций (они поставляются в комплекте), однако добавление дополнительных станций может улучшить качество трекинга и снизить вероятность попадания шлема в слепую зону. Всего в рамках одной системы виртуальной реальности можно использовать до четырех базовых станций в одной комнате на площади 10×10 м.

Поле зрения каждой станции составляет 150° по горизонтали и 110° по вертикали, поэтому две станции обычно размещают в комнате напротив друг друга по диагонали и направляют на шлем под углом от 25° до 35°. Такое расположение позволяет добиться максимальной ширины игровой зоны. Рекомендуется устанавливать станции на штативы с резьбой 1 дюйма (подойдут штативы от фототехники) или на комплектные настенные крепления с шаровым шарниром. Вариант настенного крепления станции показан на рис. 2.6.

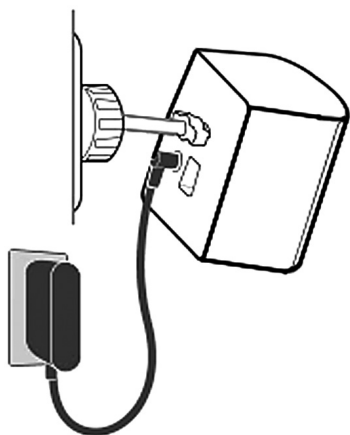


Рис. 2.6. Настенное крепление базовой станции

Наилучшая точность отслеживания достигается в диапазоне расстояний между станцией и шлемом или трекером от 0,5 до 5 м.

После подключения адаптера питания индикатор состояния замигает синим – начнется процесс определения станции и регулировки ее положения. Затем цвет индикатора сменится на зеленый или белый (в зависимости от версии станции), что укажет на готовность станции к работе. В процессе работы станция должна сохранять неподвижность, поскольку при любом смещении или колебании она вновь запустит процесс калибровки и регулировки положения.

Примечание: система виртуальной реальности VIVE Pro также обратно совместима со станциями Lighthouse 1.0 первого поколения, однако в рамках одного комплекта нельзя смешивать базовые станции разных поколений. Шлем HTC Vive первого поколения (черного цвета) не совместим с базовыми станциями второго поколения Lighthouse 2.0.

Внимание: не подключайте станцию к ПК через разъем синхронизации! Он используется только для соединения двух станций между собой в том случае, если связь между ними ненадежна. Сам кабель синхронизации microUSB-microUSB не входит в комплект поставки.

2.1.4. Подключение шлема к ПК

Шлем подключается к специальному хабу под названием Link Box, а тот, в свою очередь, к разъемам DisplayPort 1.2 и USB 3.0 на ПК (рис. 2.7). Порядок действий, следующий [65]:

1. Соедините хаб проводом USB 3.0 с соответствующим портом на задней панели компьютера.

2. Подключите провод DisplayPort к соответствующему порту на задней панели дискретной видеокарты компьютера. Для подключения хаба к ноутбуку можно воспользоваться кабелем miniDisplayPort 1.2 (не входит в комплект поставки и приобретается отдельно).

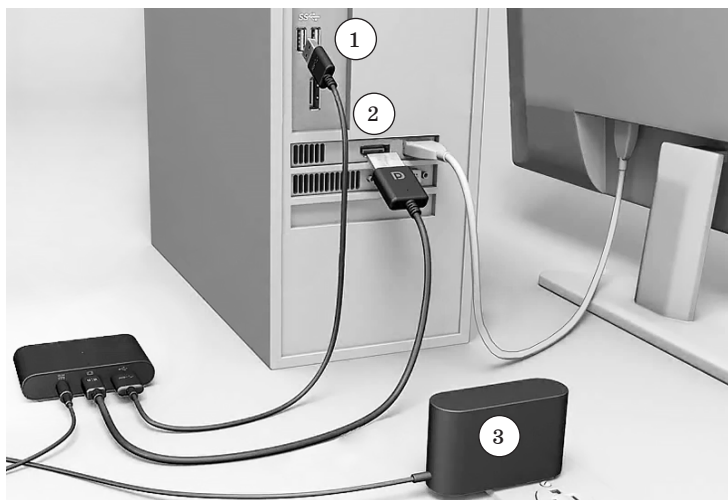


Рис. 2.7. Подключение хаба Link Box к компьютеру

3. Подключите провод питания к хабу, а блок питания – в розетку 220 В.

Внимание: при подключении хаба к ПК не используйте переходники с HDMI или других видеоинтерфейсов на DisplayPort. Переходник HDMI+USB на DisplayPort также не будет работать!

4. Подключите шлем к комбинированному порту на лицевой стороне хаба. Значок треугольника на коннекторе должен быть сверху (рис. 2.8).

5. Для включения шлема нажмите на синюю кнопку на лицевой стороне хаба. Зеленый светодиод сигнализирует о подаче питания.



Рис. 2.8. Подключение шлема VIVE Pro к хабу

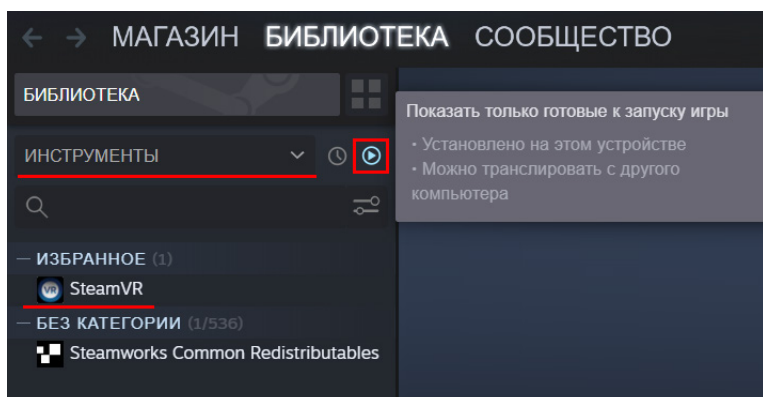


Рис. 2.9. Подключение шлема VIVE Pro к хабу

После подключения VR-комплекта VIVE Pro к компьютеру необходимо загрузить и установить платформу для игроков и разработчиков Steam (ссылка на официальный сайт для регистрации и скачивания: <https://store.steampowered.com/about/>), а внутри него – приложение SteamVR (ссылка на установку: <https://store.steampowered.com/app/250820/SteamVR/>).

После установки приложения SteamVR найти его можно будет во вкладке «Инструменты» выпадающего списка библиотеки пользователя Steam (рис. 2.9). Для простоты навигации по списку после установки SteamVR активируйте опцию «Показать только готовые к запуску игры».

2.1.5. Работа с приложением SteamVR

SteamVR – это среда выполнения для приложений виртуальной реальности [67]. Ее наличие необходимо для использования виртуальной реальности с различным оборудованием, включая шлемы Valve Index, VIVE Pro, Oculus Rift, Windows Mixed Reality и другие.

SteamVR позволяет пользователям создавать и конфигурировать игровую зону, привязывать совместимые VR-устройства и обновлять их прошивки, а также настраивать звук, режим отображения, сглаживание движений и другие параметры в играх.

После установки и запуска приложения SteamVR на экране отобразится оверлей с иконками обнаруженных устройств системы виртуальной реальности VIVE Pro (шлем, два контроллера и от 2 до 8 базовых станций, рис. 2.10).

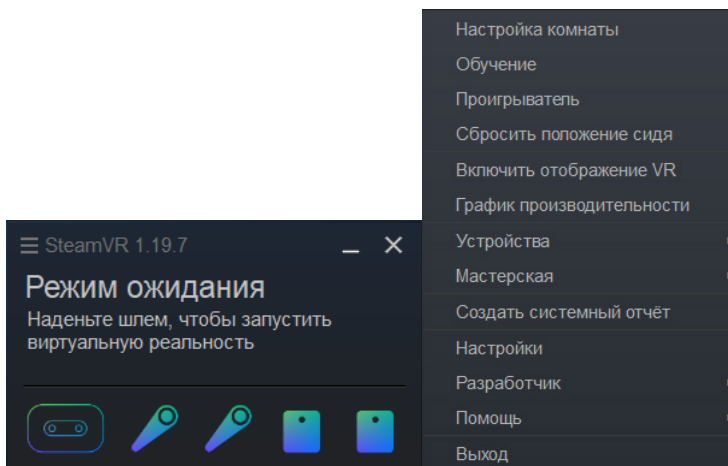


Рис. 2.10. Приложение SteamVR и его меню

Отправной точкой для пользователей устройств виртуальной реальности служит среда SteamVR Home. Это небольшое приложение представляет собой виртуальную комнату, куда автоматически попадает пользователь после подключения и настройки своего устройства виртуальной реальности. Внутри SteamVR Home игроки могут запускать приложения, общаться друг с другом, изучать различные игровые миры, выполнять задания, создавать и изменять аватары и окружающую обстановку, а также просматривать страницы сообщества и игры в магазине Steam. В самой комнате присутствуют интерактивные объекты, на которых можно потренироваться в обращении с контроллерами.

Настройка комнаты

Важным этапом перед началом использования шлема VR является первоначальная настройка игровой комнаты – помещения, где будет происходить работа с устройствами VR (рис. 2.11). Выберите размер комнаты и следуйте инструкциям на экране. Система попросит выполнить калибровку положения и высоты шлема и контроллеров, после чего для режима «большой комнаты» потребуется взять в руку контроллер и обрисовать им виртуальные границы так называемой «игровой зоны» – области пространства, внутри которой пользователь сможет свободно перемещаться. Границы этой зоны в целях безопасности будут проецироваться в виртуальное пространство, чтобы пользователь случайно не нат-

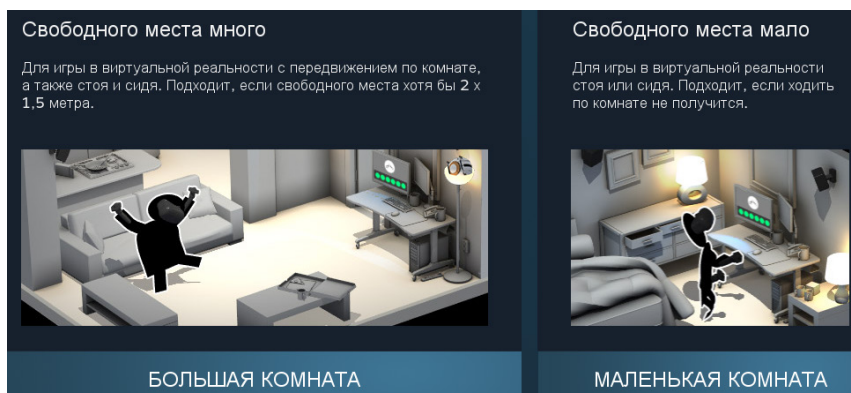


Рис. 2.11. Видимые устройства в приложении SteamVR

кнулся на стену или другие преграды, присутствующие в реальном помещении.

В режиме «маленькой комнаты» игровая зона не очерчивается, поскольку считается, что пользователь будет оставаться на месте. Хотя никакие ограничения на возможности передвижения пользователя по виртуальной среде режим «маленькой комнаты» не накладывает. Еще стоит отметить, что при калибровке уровня пола не обязательно класть шлем на пол вашей комнаты – достаточно положить его на сидение стула и задать высоту от пола до сидения в сантиметрах.

Привязка контроллера

Каждый комплект устройств виртуальной реальности VIVE Pro Full Kit может работать с двумя контроллерами: один для левой

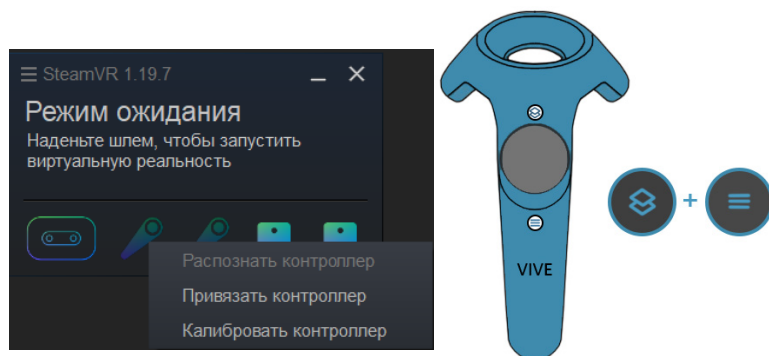


Рис. 2.12. Привязка контроллера к SteamVR

руки, второй для правой. Комплектные контроллеры привязываются к SteamVR автоматически после включения, но если вы хотите заменить один из контроллеров или привязать другую пару контроллеров, вам нужно кликнуть правой кнопкой мыши на значок контроллера в SteamVR и в открывшемся меню выбрать пункт «привязать контроллер». Чтобы перевести контроллер в режим привязки, одновременно зажмите на нем две кнопки: «система» и «меню» (рис. 2.12). Индикатор мигает синим цветом, после чего SteamVR обнаружит контроллер и добавит к активным устройствам.

2.1.6. Система ввода SteamVR для интерфейса OpenVR

OpenVR – это интерфейс программирования (API) и одновременно среда выполнения, обеспечивающие разработчику доступ к разнообразным устройствам виртуальной реальности [68]. При этом приложениям, разработанным на базе OpenVR, не нужно знать о том, какие именно устройства виртуальной реальности они используют. Движок Unity имеет встроенную поддержку OpenVR, начиная с версии 5.4 и далее.

Система ввода SteamVR – это API, с помощью которого разработчики могут добавлять в приложения виртуальной реальности поддержку разнообразных контроллеров и устройств ввода. Система ввода поставляется вместе с SDK OpenVR и позволяет пользователям создавать персональные раскладки управления для различных игр и контроллеров. Кроме того, созданными раскладками можно делиться с другими пользователями через мастерскую Steam.

При использовании системы ввода SteamVR разработчики задают высокоуровневые «действия» в приложении, которое контролирует вид раскладки управления в интерфейсе пользователей. Эта система функционирует в любом приложении, даже если разработчик еще не обновил его до новой версии системы.

Система скелетного ввода (Skeleton Input) позволяет производителям контроллеров захватывать положения рук пользователя с высокой точностью и передавать поток анимационных данных запущенному приложению. После чего этот поток данных может быть использован для анимации рук аватара пользователя.

Использование OpenVR в Unity

Для получения возможности взаимодействовать с разнообразными устройствами виртуальной реальности при разработке проек-

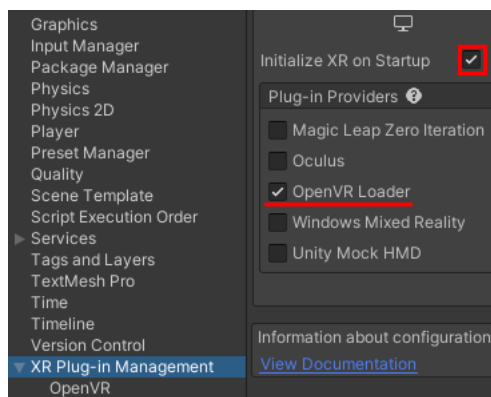


Рис. 2.13. Активация OpenVR в проекте Unity

та Unity необходимо активировать поддержку виртуальной реальности. Для этого откройте меню настроек плеера (**Player Settings**), выберите там раздел «**XR Plug-in Management**» (управление плагином XR) и поставьте галочку в пунктах «**Initialize XR on Startup**» (инициализировать при запуске) и «**OpenVR Loader**» (рис. 2.13).

Подсистема Unity VR определяет ввод данных с каждого контроллера как с отдельного джойстика. При этом доступ к параметрам отдельных осей и кнопок обеспечивается классом `UnityEngine.Input`. Благодаря абстрагированию от конкретных аппаратных платформ и моделей контроллеров, OpenVR обеспечивает поддержку широкого спектра устройств, например, контроллеров VIVE и Oculus Touch. Различия между их функциями ввода приведены в табл. 2.5.

Таблица 2.5

Различия в контроллерах VIVE и Oculus

Функция контроллера VIVE	Функция контроллера Touch	Тип взаимодействия	ID кнопки	D оси	Диапазон значений оси
Левый – кнопка Меню (1)	Button.Three	нажатие	2	–	–
Правый – кнопка Меню (1)	Button.One	нажатие	0	–	–
Левый – трекпад (2)	Button.PrimaryThumbstick	нажатие	8	–	–
Правый – трекпад (2)	Button.SecondaryThumbstick	нажатие	9	–	–
Левый – трекпад (2)	Button.PrimaryThumbstick	касание	16	–	–

Функция контроллера VIVE	Функция контроллера Touch	Тип взаимодействия	ID кнопки	D оси	Диапазон значений оси
Правый – трекпад (2)	Button. SecondaryThumbstick	касание	17	–	–
Левый – трекпад (2)	Axis2D. PrimaryThumbstick	горизонт. движение	–	1	от –1,0 до 1,0
Левый – трекпад (2)	Axis2D. PrimaryThumbstick	вертик. движение	–	2	от –1,0 до 1,0
Правый – трекпад (2)	Axis2D. SecondaryThumbstick	горизонт. движение	–	4	от –1,0 до 1,0
Правый – трекпад (2)	Axis2D. SecondaryThumbstick	вертик. движение	–	5	от –1,0 до 1,0
Левый – триггер (7)	Axis1D. PrimaryIndexTrigger	касание	14	–	–
Правый – триггер (7)	Axis1D. SecondaryIndexTrigger	касание	15	–	–
Левый – триггер (7)	Axis1D. PrimaryIndexTrigger	сжатие	–	9	от 0,0 до 1,0
Правый – триггер (7)	Axis1D. SecondaryIndexTrigger	сжатие	–	10	от 0,0 до 1,0
Левый – кнопка хвата (8)	Axis1D. PrimaryHandTrigger	сжатие	–	11	от 0,0 до 1,0
Правый – кнопка хвата (8)	Axis1D. SecondaryHandTrigger	сжатие	–	12	от 0,0 до 1,0

Когда вы подключите и правильно настроите совместимые с OpenVR контроллеры, они отобразятся в списке устройств как «OpenVR Controller – Left» и «OpenVR Controller – Right» и подсветятся зеленым цветом в окне состояния SteamVR. Проверка доступности контроллеров осуществляется специальным скриптом Unity с некоторой периодичностью. В том случае, если контроллеры разрядятся или будут отключены пользователем, их названия в системе изменятся на пустые текстовые поля. При включении контроллеров названия снова появятся.

2.1.7. Описание плагина XR Plug-in

Разработанный Unity плагин XR представляет собой унифицированный фреймворк для прямой интеграции поддержки устройств виртуальной и смешанной реальности в проекты для различных платформ [69]. Он включает в себя программный интерфейс (API), представляющий базовую функциональность по работе с цифровыми реальностями для всех поддерживаемых платформ.

Такая организация архитектуры предлагает следующие преимущества:

- Наличие инструментов разработки под различные целевые платформы (например, AR Foundation или XR Interaction Toolkit).
- Более быстрый процесс выпуска обновлений поддерживаемых плагинов компаниями-партнерами. Для доставки обновлений используется менеджер пакетов Unity.
- Большое количество платформ получают доступ к интерфейсу для эффективного использования оптимизаций визуализатора и инструментов разработки Unity XR.

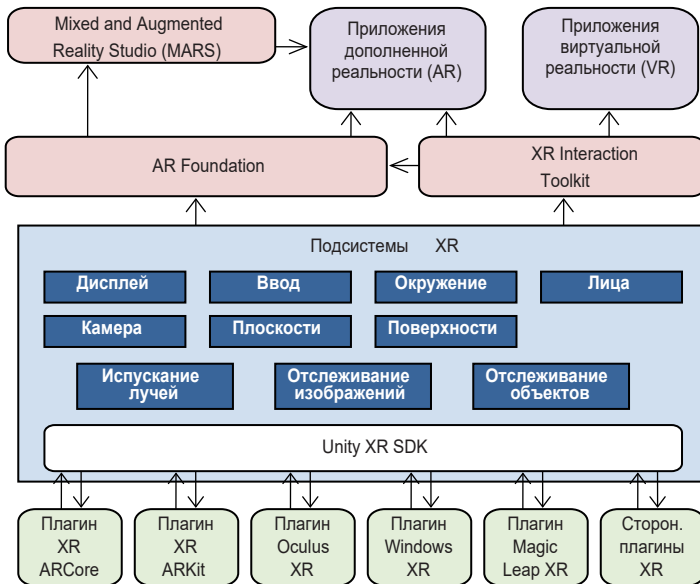


Рис. 2.14. Структура фреймворка XR Plug-in

Сам термин XR объединяет под одной оболочкой приложения виртуальной реальности (VR) и смешанной реальности (MR), куда входят дополненная реальность (AR) и дополненная виртуальность (AV). Схема на рис. 2.14 демонстрирует текущую структуру фреймворка XR.

Unity позволяет разрабатывать XR-приложения для следующих целевых платформ:

- ARKit;
- ARCore;
- Microsoft HoloLens;
- Windows Mixed Reality;
- Magic Leap;
- Oculus;
- OpenXR;
- PlayStation VR.

Важно: XR для WebGL не поддерживается в Unity!

2.1.8. Подготовка проекта Unity для шлема VIVE Pro

В данном разделе рассмотрим типовые действия, которые необходимо выполнить для создания нового проекта Unity с поддержкой шлема VIVE Pro и сопутствующих ему устройств.

Создание проекта Unity и импорт ассетов

На начальном этапе необходимо создать новый проект Unity с пресетом 3D. Это можно сделать через интерфейс менеджера проектов **Unity Hub**. Далее в пустой проект необходимо импортировать плагин SteamVR, включающий необходимые зависимости и набор

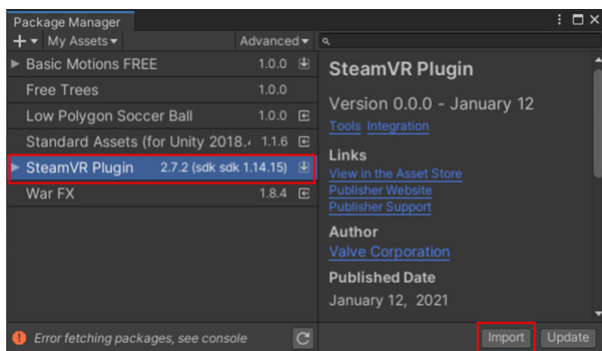


Рис. 2.15. Импорт плагина SteamVR в проект



Рис. 2.16. Применение рекомендуемых настроек проекта

ассетов для работы с виртуальной реальностью. Плагин SteamVR свободно доступен для скачивания в магазине ассетов Unity Asset Store (прямая ссылка: <https://assetstore.unity.com/packages/tools/integration/steamvr-plugin-32647>). Импорт плагина в текущий проект выполняется при помощи менеджера пакетов Unity (**Package Manager**, рис. 2.15).

После импорта плагина Unity сам скачает пакет «**OpenVR for Desktop**» и активирует его в настройках плеера (**Player Settings**). Также движок предложит пользователю принять рекомендуемые настройки плеера для виртуальной реальности (рис. 2.16).

Настройка органов управления

Следующий шаг – создание файла конфигурации контроллеров, в котором будет храниться схема управления (назначения клавиш). Для этого требуется открыть настройки системы ввода SteamVR («**Window → SteamVR Input**») и в открывшемся окне нажать «**Save and generate**», чтобы система автоматически сгенерировала классы для инициализации схемы управления (рис. 2.17). После этого в редакторе Unity станет возможна работа с контроллерами VIVE.

Проверить работоспособность шлема и контроллеров можно с помощью стандартной демонстрационной сцены «**Interaction System**». Она расположена в папке с ассетами проекта по адресу: «**Assets/SteamVR/InteractionSystem/Samples/Interaction_Example**».

Для изменения штатной схемы управления и переназначения клавиш используется интерфейс Binding UI. После нажатия на кнопку «**Open binding UI**» в окне «**SteamVR Input**» откроется окно с изображением контроллеров, где можно будет переназначить кла-

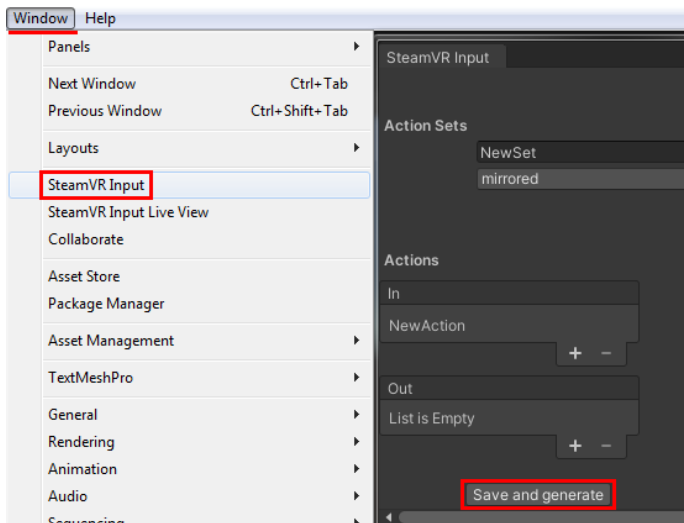


Рис. 2.17. Генерация файлов с настройками управления

виши или добавить новые действия (рис. 2.18). Чекбокс «**Mirror Mode**» (зеркалирование) дублирует функции органов управления с одного контроллера на другой. При подключении другого совместимого контроллера (не от VIVE Pro) картинка изменится, и клавиши можно будет переназначить заново.

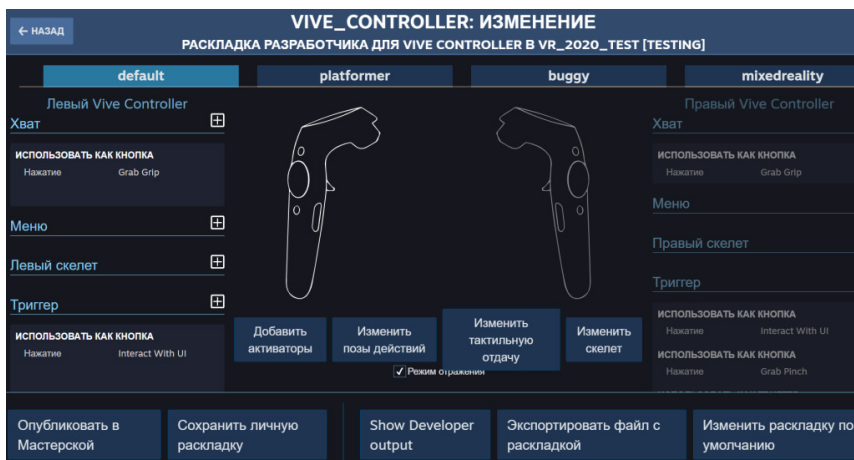


Рис. 2.18. Настройка органов управления контроллеров VIVE

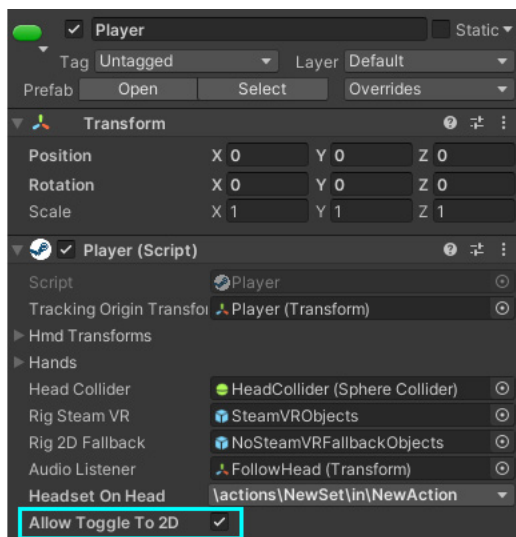


Рис. 2.19. Настройка игрока в Инспекторе

Добавление персонажа и функции обзора в сцене

В ассетах плагина SteamVR имеется готовый префаб игрока, для которого заранее запрограммированы функции обзора в сцене. Этот префаб можно использовать для аватара игрока в сцене.

Но перед размещением игрока необходимо задать в сцене плоскость земли, по которой будет передвигаться аватар. Для этого проще всего использовать стандартный примитив «Плоскость» («**GameObject** → **3D Object** → **Plane**»). Сам префаб игрока «**Player**» находится по адресу: «\Assets\SteamVR\InteractionSystem\Core\Prefabs».

В списке глобальных переменных скрипта префаба игрока имеется пункт «**Allow Toggle To 2D**», который позволяет имитировать наличие шлема виртуальной реальности при его фактическом отсутствии (рис. 2.19). Этот пункт может быть полезен при разработке и отладке проекта на компьютере, к которому в данный момент не подключен шлем VIVE Pro.

Добавление иконок-гизмо

Для удобства восприятия и навигации полезно будет реализовать отображение в сцене рук и ног виртуального игрока в виде иконок-гизмо. Они хранятся в папке «**Icons**» по адресу: «\Assets\SteamVR\InteractionSystem\Core». Чтобы гизмо появились в проекте, необхо-

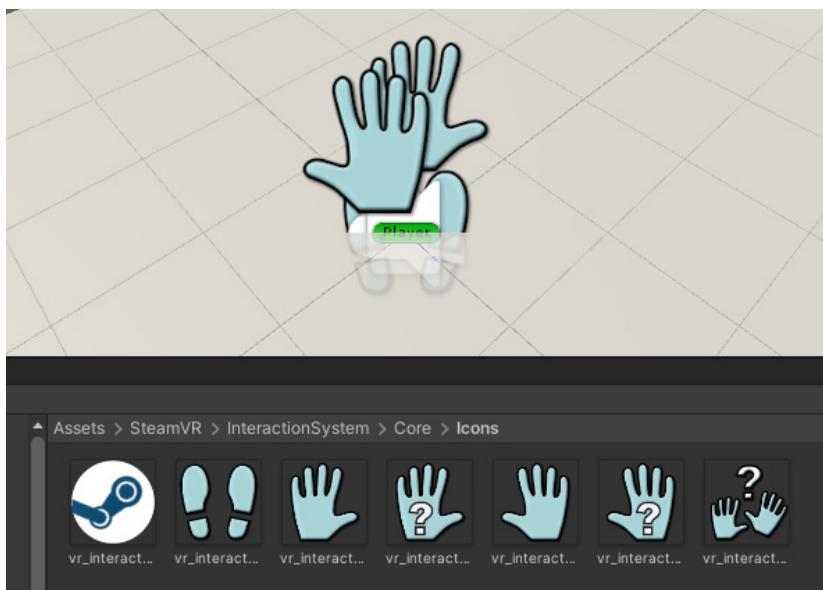


Рис. 2.20. Добавление иконок-гизмо в сцену

можно открыть место расположения иконок в проводнике Windows и скопировать их все в папку с ресурсами проекта «Assets/Gizmos» (рис. 2.20). В результате в окне редактора Unity отобразятся иконки рук и ног персонажа.

На этом базовая подготовка проекта Unity завершена. Дальнейшие действия разработчика будут связаны с добавлением в проект желаемого типа перемещения аватара (непрерывного или при помощи телепортации) и реализацией взаимодействия с объектами при помощи контроллеров или жестов.

2.2. Дополненная виртуальность на базе трекеров VIVE

Наряду со шлемом виртуальной реальности компания HTC также разработала и выпустила в продажу внешний трекер HTC VIVE Tracker. Впервые он был продемонстрирован компанией публике на выставке CES 2017. Это небольшое устройство с тремя «лепестками» и массивом датчиков системы SteamVR позволяет «перенести» информацию о перемещении и вращении любого реального предмета в виртуальную реальность. Трекер крепится к какому-либо предмету или части тела человека, после чего все его движения передаются в виртуальную среду. Так реализуется концепция допол-



Рис. 2.21. Трекер VIVE Tracker 2.0 и его комплектация

ненной виртуальности. Вдобавок трекер может выступать в роли контроллера, поскольку имеет встроенную кнопку и специальный интерфейс для подключения сторонних средств ввода/вывода.

Рассмотрим обновленную версию трекера с индексом 2.0.

HTC VIVE Tracker 2.0 был выпущен в 2018 году как замена оригинальной модели. Новая версия получила поддержку работы с базовыми станциями Lighthouse 2.0 (второго поколения), при этом внешний вид устройства в целом остался неизменным, за исключением цвета кнопки на фронтальной панели, которая стала голубой (рис. 2.21).

2.2.1. Комплект поставки и системные требования

В комплект HTC Vive Tracker 2.0 входят:

- HTC VIVE Tracker 2.0;
- USB-ретранслятор;
- Провод microUSB.

Минимальные системные требования к ПК для работы с HTC Vive Tracker 2.0 повторяют таковые для шлема VIVE Pro (см. табл. 2.1). В случае использования трекеров совместно с парой стандартных контроллеров VIVE 2.0 к компьютеру необходимо будет подключить USB-ретранслятор через кабель microUSB.

2.2.2. Устройство и технические характеристики

HTC VIVE Tracker 2.0 является контроллером, т. е. устройством, которое используется для взаимодействия пользователя с виртуальными объектами. Также, будучи закрепленным на каком-либо объекте, трекер позволяет отслеживать его положение и ориентацию в пространстве за счет 18 встроенных ИК датчиков [70]. Технические характеристики HTC VIVE Tracker 2.0 представлены в табл. 2.6.

Таблица 2.6

Спецификации HTC Vive Tracker 2.0

Элемент	Характеристики
Технология трекинга	SteamVR Tracking 2.0
Элементы ввода	Кнопка
Варианты подключения	Разъем microUSB Разъем GPIO Bluetooth
Питание	Встроенная батарея емкостью 700 мАч

Схематичное изображение VIVE Tracker 2.0 с подписанными органами управления и элементами конструкции представлено на рис. 2.22.

Цифрами на рисунке обозначены следующие элементы: 1 – ИК-датчики; 2 – кнопка; 3 – светодиод; 4 – стопор для крепления; 5 – винтовое дюймовое крепление; 6 – GPIO панель для подключения POGO-пинов, 7 – разъем microUSB.

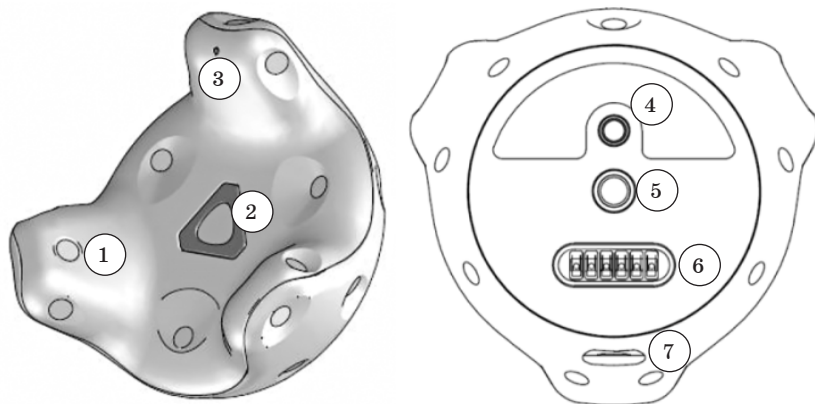


Рис. 2.22. Схематичное изображение HTC VIVE Tracker 2.0

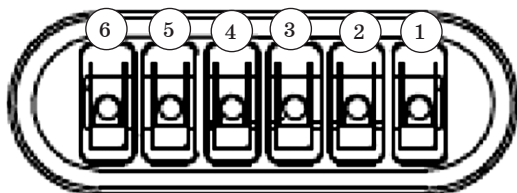


Рис. 2.23. Контактные площадки POGO-Pin

Таблица 2.7

Назначение контактов POGO-Pin

Номер контакта	Тип	Описание
1	Цифровой вывод	Контакт общего назначения для вывода данных
2	Земля (GND)	Земля
3	Цифровой ввод/вход питания	Контакт общего назначения для ввода данных. Имеет встроенный подтягивающий резистор, идущий к VDD. Сигнал с активным нижним уровнем (кнопка «Захват»). Контакт подачи питания
4	Цифровой ввод	Контакт общего назначения для ввода данных. Имеет встроенный подтягивающий резистор, идущий к VDD. Сигнал с активным нижним уровнем (кнопка «Триггер»)
5	Цифровой ввод	Контакт общего назначения для ввода данных. Имеет встроенный подтягивающий резистор, идущий к VDD. Сигнал с активным нижним уровнем (кнопка «Трекпад»)
6	Цифровой ввод	Контакт общего назначения для ввода данных. Имеет встроенный подтягивающий резистор, идущий к VDD. Сигнал с активным нижним уровнем (кнопка «Меню»)

Колодка с подвижными контактами POGO-Pin позволяет подключать к трекеру различные устройства ввода/вывода, что расширяет его функционал. POGO-контакты нумеруются справа-налево (рис. 2.23), и каждый из них имеет свою функцию (они приведены в табл. 2.7).

2.2.3. Включение и настройка VIVE Tracker 2.0

Для работы с трекером сперва необходимо подключить к компьютеру шлем VIVE Pro, включить базовые станции и, если требуется, контроллеры VIVE 2.0. Если трекер будет использоваться

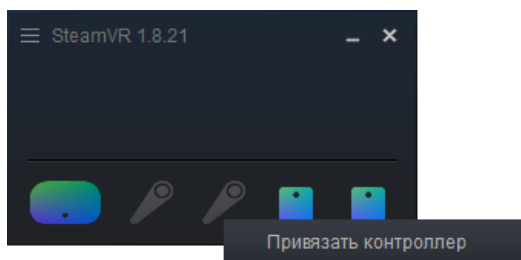


Рис. 2.24. Привязка нового контроллера

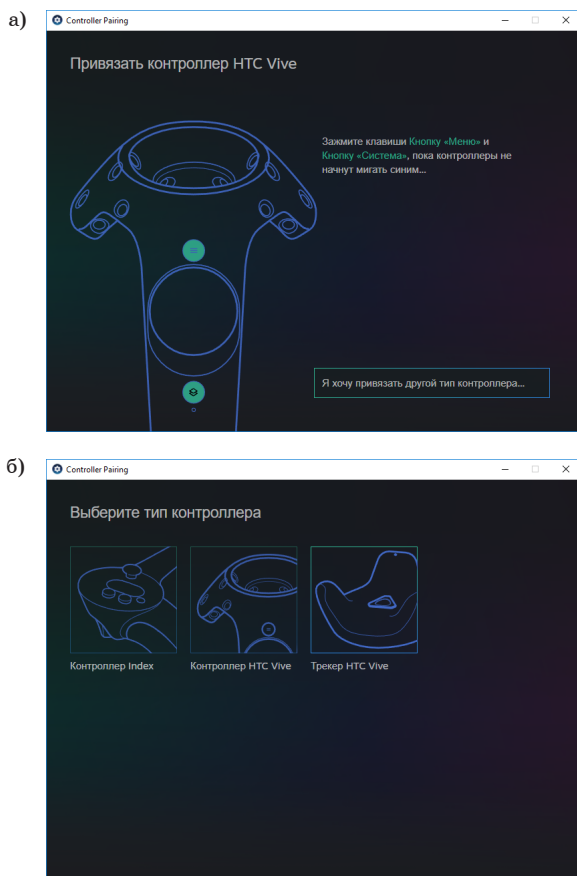


Рис. 2.25. Привязка трекера HTC Vive:
а – режим привязки контроллера; б – выбор типа контроллера

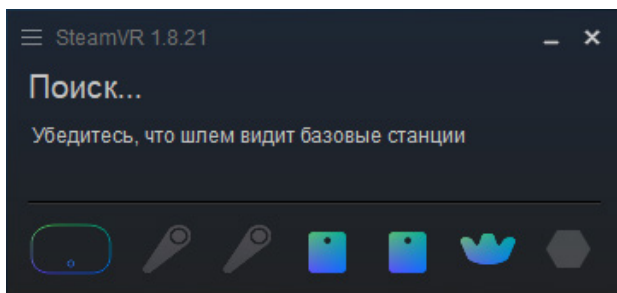


Рис. 2.26. Отображение трекеров на панели состояния SteamVR

совместно с двумя контроллерами, к ПК потребуется также подключить USB-ретранслятор при помощи кабеля microUSB. В противном случае ретранслятор можно не подключать. Добавить трекер к перечню активных устройств виртуальной реальности можно через контекстное меню приложения SteamVR. Для этого необходимо щелкнуть по окну состояния SteamVR правой кнопкой мыши и нажать кнопку «Привязать контроллер» (рис. 2.24).

Далее нужно выбрать и нажать на кнопку с надписью: «Я хочу привязать другой тип контроллера...» (рис. 2.25, а), после чего выбрать из перечня устройств «Трекер HTC Vive» (рис. 2.25, б).

Добавление трекера завершено. В последнем окне можно нажать «Готово» для завершения привязки устройства или добавить еще один трекер.

После успешного подключения трекера в окне состояния SteamVR появятся новые подключения. На рис. 2.26 шестая иконка на панели обозначает активный подключенный трекер, а седьмая – неактивный.

2.2.4. Работа с трекером в редакторе Unity

В п. 2.1.8 была описана подготовка проекта Unity для работы с виртуальной реальностью с использованием шлема VIVE Pro. Поскольку трекеры являются частью инфраструктуры VIVE, проект можно легко дополнить, чтобы активировать поддержку трекеров.

Если создать в сцене некоторый объект, например, примитив-куб («GameObject → 3D Object → Cube»), добавить к нему скрипт «TrackedObject.cs», расположенный по адресу: «\Assets\SteamVR\InteractionSystem\Core\Scripts», и в настройках этого скрипта выбрать правильный номер устройства (Device ID), то после запуска сцены куб закрепится за положением и ориентацией трекера

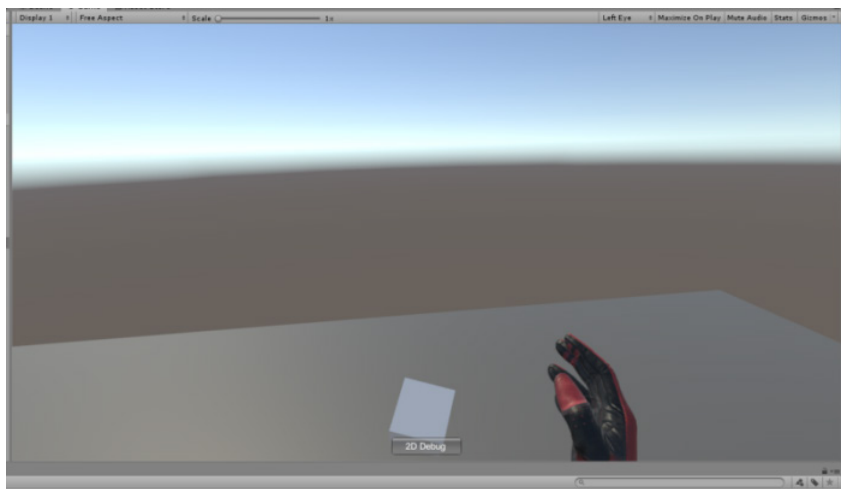


Рис. 2.27. Отображение куба, закрепленного за трекером

(рис. 2.27). Если теперь взять трекер в руку, куб будет следовать за перемещением кисти.

Ручной подбор верного значения параметра «Device ID» может быть утомительным, поскольку здесь немаловажную роль играет порядок подключения устройств к приложению SteamVR. Следовательно, нет никакой гарантии, что при следующем включении программы, например после перезагрузки, этот параметр сохранит свое прежнее значение.

Однако можно написать (и добавить к кубу) такой скрипт, который будет самостоятельно присваивать параметр «Device ID», обращаясь к внутреннему имени устройства. Код данного скрипта приведен ниже.

Код скрипта «DeviceID.cs»

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using Valve.VR;

public class DeviceID : MonoBehaviour
{
    GameObject go;
    void Start()
    {
```

```

uint index = 0;
var error = ETrackedPropertyError.TrackedProp_Success;
for (uint i = 0; i < 16; i++)
{
    var result = new System.Text.StringBuilder((int)64);
    OpenVR.System.GetStringTrackedDeviceProperty(i, ETrackedDeviceProperty.Prop_RenderModelName_String, result, 64, ref error);
    if (result.ToString().Contains("tracker"))
    {
        index = i;
        this.GetComponentInParent<SteamVR_TrackedObject>().index =
(SteamVR_TrackedObject.EIndex)index;
        Debug.Log("Трекер найден! Идентификатор устройства успешно
изменён на id_"+i);
        break;
    }
    else
    {
        Debug.Log("Трекер не найден!");
    }
}
}
}
}

```

Данный скрипт выполняет перебор всех записей свойств подключенных устройств (максимальное количество которых может достигать 16). Если в имени устройства был найден фрагмент «tracker», который указывает на наличие в системе зарегистрированного трекера, его порядковый номер записывается в скрипт «TrackedObject.cs», расположенный на том же объекте, на который был применен скрипт «DeviceID.cs».

Таким образом, если на объект сцены добавить скрипты «TrackedObject.cs» и «DeviceID.cs», порядковый номер устройства трекера будет автоматически синхронизироваться с параметром «Index» скрипта «TrackedObject.cs». Это значит, пользователю не придется настраивать параметр «Device ID» вручную.

2.2.5. ДВ с использованием двойной камеры шлема

Еще одна технология, реализующая дополненную виртуальность для устройств VIVE, – это сдвоенная камера шлема VIVE Pro. Она позволяет добавить реальный мир в виртуальную реальность. Некоторые приложения виртуальной реальности используют сдвоенную камеру для сканирования помещения, чтобы затем встроить предметы интерьера комнаты в интерфейс виртуальной реальности.

Для активации сдвоенной камеры необходимо запустить приложение SteamVR, перейти в меню приложения в раздел «**Настройки** → **Камера**» и там активировать камеру переключателем «**Включить/Отключить**». После этого необходимо перезапустить приложение SteamVR. Теперь в разделе настройках камеры появятся следующие пункты:

– *Показывать камеру на контроллере.* Рядом с контроллером появится экран, на котором будет постоянно отображаться реальное окружение.

– *Показывать камеру на краю комнаты.* Реальное окружение будет отображаться только при приближении к границе игровой зоны.

– *Включить вид комнаты.* При двойном нажатии системной кнопки на дисплеях шлема отобразится реальное окружение. Следующее двойное нажатие вернет пользователя обратно в виртуальную реальность.

2.3. Знакомство с VR-шлемом Oculus Quest 2

Шлем виртуальной реальности Oculus Quest 2 был представлен 13 октября 2020 года и пришел на смену предыдущей модели Oculus Quest. Новая версия получила единый дисплей с более высоким разрешением на каждый глаз (1832×1920), новую версию процессора, увеличение объема ОЗУ в полтора раза, поддержку более высокой частоты обновления экрана и обновленные контроллеры Oculus Touch. Материал шлема, как и его форма, остались прежними; общий вес устройства составил 571 грамм, что на 62 грам-



Рис. 2.28. Основные компоненты VR-системы Oculus Quest 2 [71]

ма меньше модели первого поколения. Фото шлема, контроллеров, зарядного устройства, рамки для очков и провода Oculus Quest 2 представлено на рис. 2.28.

2.3.1. Комплект поставки и системные требования

В комплект системы виртуальной реальности Oculus Quest 2 входят:

- контроллеры Oculus Touch (обновленные под Oculus Quest 2) – две штуки;
- элементы питания для контроллеров типа AA – две штуки;
- пластиковая рамка для использования шлема с очками;
- шлем Oculus Quest 2;
- кабель USB-C 3.0 (male-to-male);
- адаптер питания шлема USB-C.

Минимальные системные требования к ПК для работы с комплектом Oculus Quest 2 приведены в табл. 2.8 [72].

Таблица 2.8

Минимальные системные требования Oculus Quest 2

Элемент	Требование
Центральный процессор	Intel i5-4590 / AMD Ryzen 5 1500X или лучше
Видеокарта	Nvidia GeForce GTX 970 / AMD Radeon 400 серии или лучше (объем памяти не менее 3 ГБ)
Оперативная память	Не менее 8 ГБ
Интерфейс USB	1 порт стандарта USB-C 3.0
Предустановленное ПО	Oculus Client
Операционная система	Windows 10

Примечание: следует понимать, что Oculus Quest 2 является самостоятельным устройством, поэтому наличие компьютера для его использования не обязательно. При работе со шлемом в автономном режиме приложения скачиваются во внутреннюю память шлема из магазина Oculus или по кабелю USB (доступно для разработчиков). Шлем можно использовать как периферийное устройство для компьютера – в таком случае на нем активируется опция Oculus Link (проводное подключение) или Oculus Air Link (беспроводное подключение), а само приложение запускается на компьютере. Подключение Oculus Quest 2 к компьютеру по Wi-Fi стандарта 2.4 ГГц тоже возможно, однако качество передаваемого изображе-

ния в таком случае будет посредственным. Поэтому рекомендуется использовать стандарт Wi-Fi 5 ГГц.

2.3.2. Устройство и технические характеристики

Технические характеристики Oculus Quest 2 приведены в табл. 2.9.

Таблица 2.9

Спецификации шлема Oculus Quest 2

Элемент	Характеристики
Размер и тип экрана	Единый Fast-Switch LCD дисплей разрешения 3664×1920 пикселей
Разрешение экрана	1832×1920 пикселей на каждый глаз (суммарно 3664×1920 пикселей)
Частота обновления	90 Гц (120 Гц в экспериментальном режиме)
Угол обзора	90°
Вывод звука	Встроенные наушники. Разъем mini Jack 3,5 мм. Разъем USB-C 3.0 с поддержкой наушников USB-C
Ввод звука	Встроенный микрофон
Ввод видео	Четыре широкоформатные фронтальные камеры
Соединение и коммуникация	– USB-C 3.0 – Mini jack 3.5 мм – Wi-Fi 6 и Wi-Fi 802.11ay 60 ГГц – Bluetooth (в экспериментальном режиме)
Датчики	Система отслеживания Oculus Insight (Inside-Out система на основе 4 встроенных камер (6DoF) и 15 ИК-датчиков на каждом контроллере) – Акселерометр – Гироскоп – Датчик приближения
Эргономика	– Подстройка расстояния от глаз до линз (два состояния: с рамкой для ношения очков и без) – Подстройка межзрачкового расстояния (в трех позициях 58 мм, 63 мм и 68 мм) – Ремень с регулировкой длины и глубины посадки на голову
Процессор	Qualcomm Snapdragon XR2 (вариант чипа Snapdragon 865)
Оперативная память (ОЗУ)	6 ГБ

Элемент	Характеристики
Встроенная память (накопитель)	64/128/256 ГБ
Провод	USB-C 3.0 (m) – USB-C 3.0 (m) для зарядки устройства и передачи данных (~0.5 м)
Зарядное устройство	USB-C 3.0 (f)

Схематичное изображение шлема Oculus Quest 2 с подписанными элементами конструкции приведено на рис. 2.29.

Цифрами на рисунке обозначены следующие элементы: 1 – разъем USB-C 3.0; 2 – разъем mini jack 3.5 мм; 3 – встроенный микрофон; 4 – объективы фронтальных камер; 5 – ремешок крепления; 6 – встроенные наушники; 7 – лицевая прокладка; 8 – кнопка; 9 – кнопка; 10 – кнопка; 11 – кнопка.

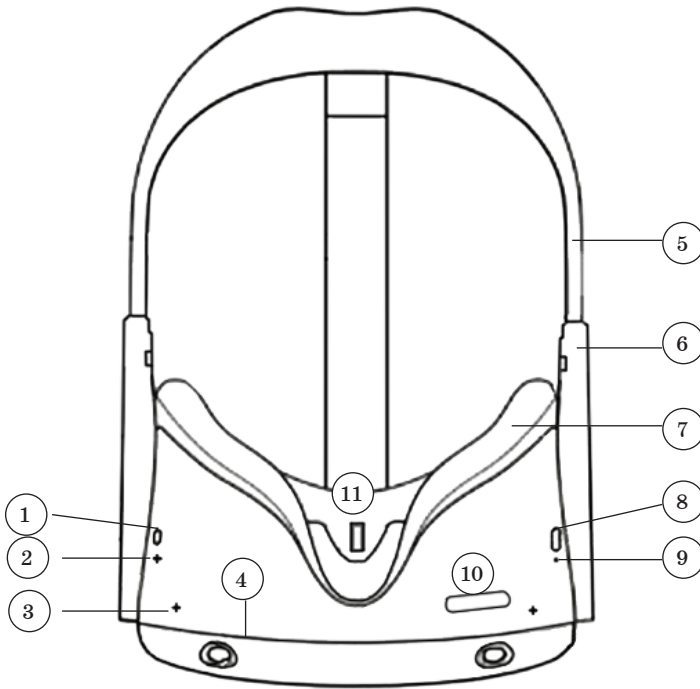


Рис. 2.29. Схематичное изображение шлема Oculus Quest 2

включения/перехода в спящий режим; 9 – индикатор состояния; 10 – кнопка изменения уровня громкости +/-; 11 – датчик приближения.

Штатные наушники не отстегиваются, но при желании пользователь может подключить другие наушники через разъем mini Jack 3.5мм или воспользоваться наушниками с интерфейсом USB-C. Также через разъем USB-C к шлему можно подключить внешнее зарядное устройство, но в этом случае воспользоваться USB-C-совместимыми наушниками будет невозможно. Если к шлему одновременно подключить двое наушников: одни через mini Jack 3.5 мм, а другие через USB-C, активными будут только те, что подключены к mini Jack 3.5 мм.

По аналогии с контроллерами VIVE, контроллеры Oculus используются для взаимодействия пользователя с виртуальными объектами и отслеживания положения и ориентации рук пользователя в пространстве за счет встроенных ИК датчиков (всего их по 15 штук на каждом контроллере). Технические характеристики контроллеров Oculus Quest 2 представлены в табл. 2.10.

Таблица 2.10

Спецификации контроллера Oculus Quest 2

Элемент	Характеристики
Технология трекинга	Oculus Insight
Элементы ввода (на каждом контроллере)	– Джойстик – Две кнопки нажатия – Аналоговый триггер – Боковой захват – Кнопка вызова меню (на левом) / системная кнопка (на правом)
Питание	Разъем для аккумуляторов типа AA

Схематичное изображение контроллера с подписанными органами управления и элементами конструкции приведено на рис. 2.30.

Цифрами на рисунке обозначены следующие элементы: 1 – аналоговые джойстики; 2 – кнопки дополнительных действий: X, Y, A, B; 3 – кнопка вызова меню; 4 – кнопка «Oculus» (системная кнопка); 5 – кнопка «Захват»; 6 – курок (Trigger).

Примечание: контроллеры для шлема Oculus Quest 2 носят название Oculus Touch, однако это название уже использовалось ранее для других контроллеров Oculus. Так, на рис. 2.31 изображены контроллеры Oculus Touch, выпущенные для устройств Oculus Rift

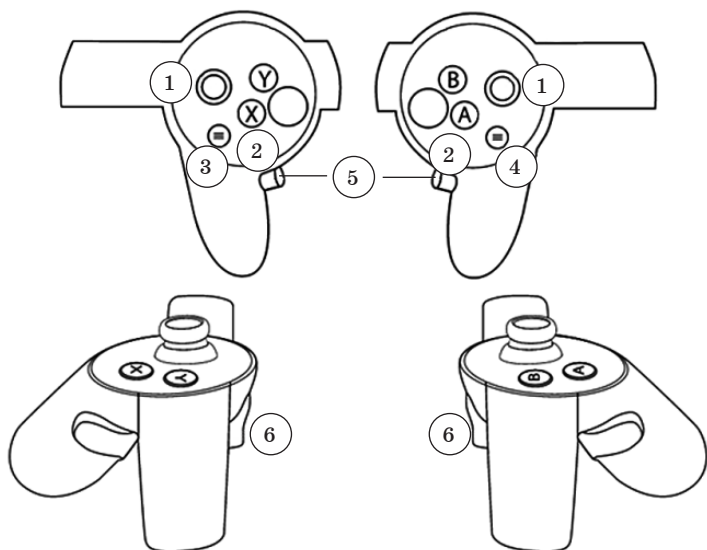


Рис. 2.30. Схематичное изображение контроллера Oculus Touch



Рис. 2.31. Внешний вид контроллеров Oculus Touch (Rift S, Quest, Quest2)

S, Oculus Quest и Oculus Quest 2. Несмотря на внешние отличия, расположения кнопок и их функционал не менялся.

2.3.3. Включение и настройка шлема

Для работы со шлемом необходимо его включить нажатием на кнопку, расположенную на правой боковой стенке (см. рис. 2.29, 8). При этом рядом с кнопкой включения загорится диод (см. рис. 2.29, 9). На переносице шлема с внутренней стороны расположен датчик

приближения, поэтому после снятия его с головы экраны дисплеев гаснут, а сам шлем уходит в спящий режим.

Защитная система – это программа, которая помогает пользователю видеть очерченные для перемещения границы внутри виртуальной реальности. После запуска шлема защитная система постарается определить его местоположение, основываясь на последней сохраненной конфигурации. В случае, если восстановить месторасположение не удалось, либо если шлем используется в новом месте – необходимо задать границы защитной системы заново. В этом случае сперва необходимо указать расстояние до пола, опустив один из контроллеров на пол. Затем нужно указать границы зоны перемещения. Пользователю будет дан выбор: «Использовать границы без перемещения» или «Использовать границы с перемещением». В случае выбора границ без перемещения защитная система нарисует небольшой круг вокруг пользователя с диаметром в один метр и запомнит его как границы. В случае выбора границ с перемещением пользователю потребуется очертить зону при помощи одного из контроллеров. Границы рисуются на полу с помощью виртуальной лазерной указки при зажатой кнопке курка. После удачной настройки защитной системы запустится интерфейс окружения, который состоит из панели внизу экрана и активных окон. По умолчанию открывается элемент панели «Лента» с актуальными для пользователя новостями (рис. 2.32).

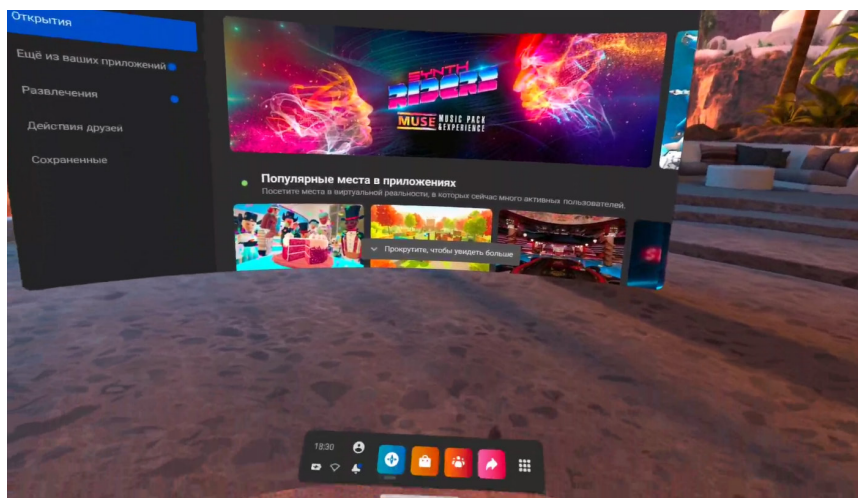


Рис. 2.32. Интерфейс Oculus Quest 2 после включения и настройки

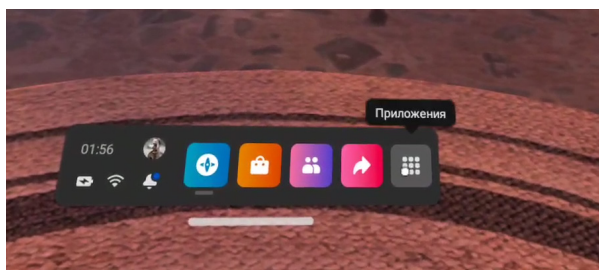


Рис. 2.33. Элемент нижней панели «Приложения»

При использовании шлема пользователь может настраивать те или иные параметры под свои нужды. Рассмотрим некоторых из них.

Сперва на нижней панели необходимо выбрать пункт «**Приложения**» (рис. 2.33), а затем – «**Настройки**» (Settings, рис. 2.34).

После запуска приложения настроек перед пользователем откроется окно, в котором все настройки шлема поделены на категории в виде списка в левой части экрана. Для активации режима использования жестов рук вместо контроллеров необходимо открыть категорию «**Устройство**», выбрать там пункт «**Настройки использования рук и контроллеров**», после чего включить параметры

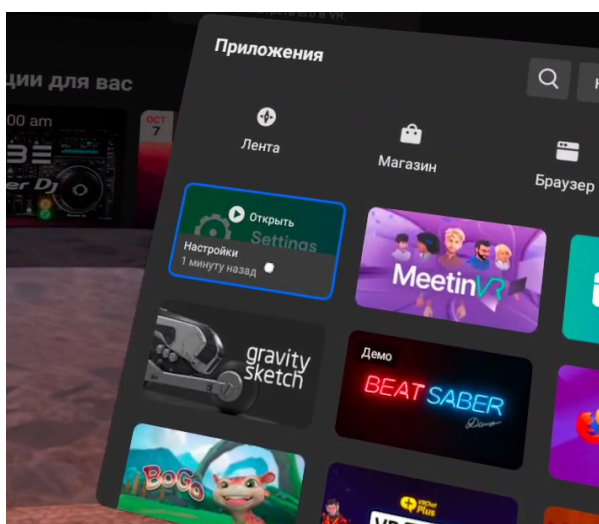


Рис. 2.34. Приложение «Настройки»

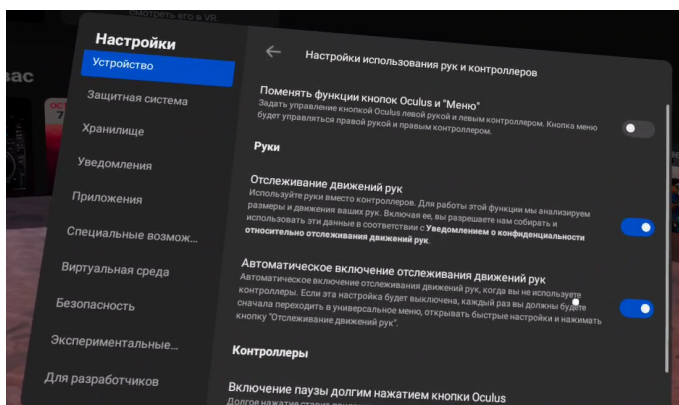


Рис. 2.35. Настройки использования жестов рук

«Отслеживание движений рук» и «Автоматическое включение отслеживания движений рук» (рис. 2.35).

Функции отключения/включения микрофона и смены языка доступны в категории «Устройство >> Выключить микрофон» и «Устройство >> Язык» (рис. 2.36).

Для изменения параметров защитной системы, например ее сброса и перенастройки, необходимо открыть категорию «Защитная система» и изменить параметры «Настроить защитную систему» и «Настройка уровня пола». Кроме этого, рекомендуется включить параметр «Коснуться дважды» для режима «Сквозная камера», поскольку это позволит пользователю в любой момент времени включить сквозную камеру и увидеть окружение, щелкнув два раза по шлему со стороны одного из висков. Повторение этого действия вернет пользователя обратно в виртуальную реальность, что

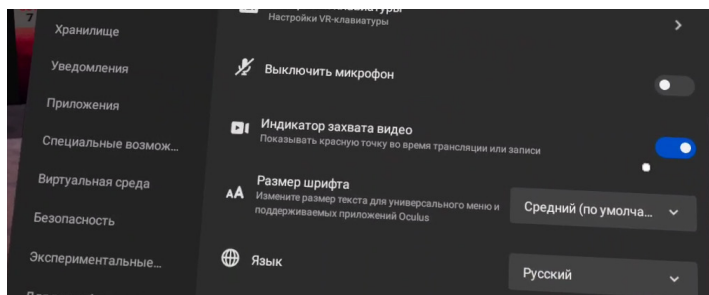


Рис. 2.36. Настройки микрофона и языка

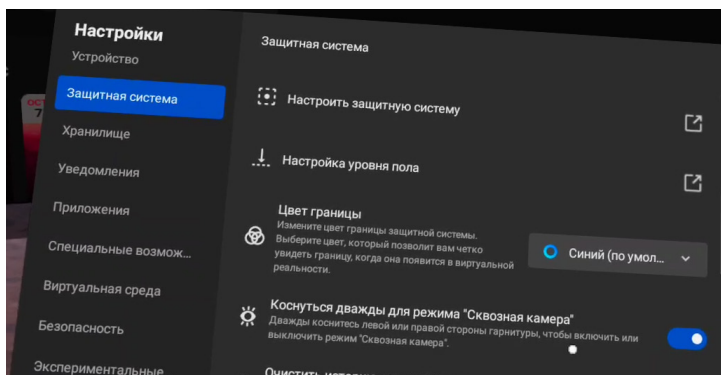


Рис. 2.37. Настройка защитной системы

значительно снизит риск травм и поможет ориентироваться в пространстве (рис. 2.37).

Для подключения шлема к ПК через Wi-Fi необходимо открыть категорию «**Экспериментальные функции**» и включить там параметр «**Air Link**» (рис. 2.38).

Пользователь может менять и другие параметры, которые находятся в разных категориях, однако некоторые параметры будут оставаться скрытыми. Один из них – это возможность выполнять запись экрана в шлеме через пункт нижней панели «**Поделиться**» в формате 16:9, а не в формате 4:3 по умолчанию. Для получения возможности управлять скрытыми параметрами пользователю придется скачать приложение **SideQuest** на компьютер и выполнить установку по пунктам, указанным на сайте приложения (ссылка: <https://sidequestvr.com/setup-howto>). После установки и запуска

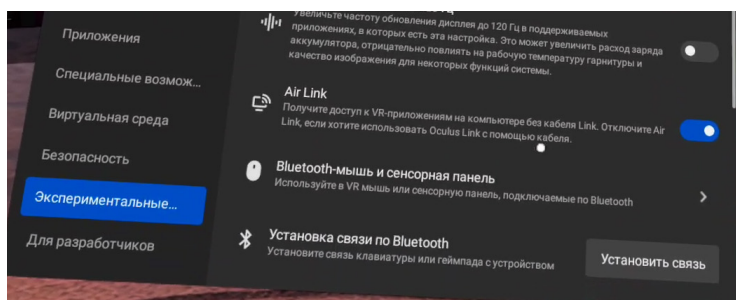


Рис. 2.38. Настройка Air Link

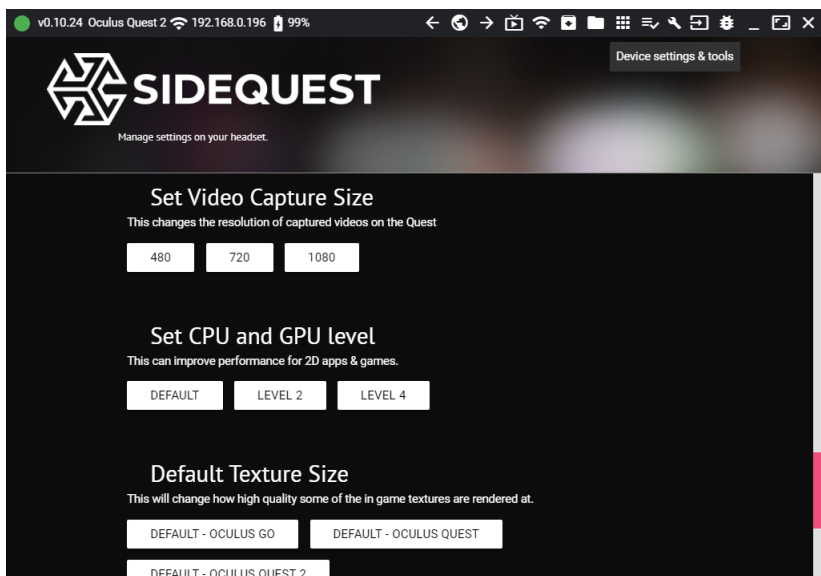


Рис. 2.39. Настройка шлема в приложении SideQuest

приложения SideQuest нужно подключить шлем к компьютеру через USB-провод и выбрать на верхней панели пункт «**Device settings & tools**». Это откроет доступ к скрытым настройкам, в том числе формата видео (рис. 2.39).

Приложение SideQuest позволяет не только применять настройки из списка на рис. 2.39, но и выполнять любые введенные пользователем команды для Android-устройств (какими, в том числе, являются шлемы серии Oculus Quest), удобно скачивать файлы с внутренней памяти шлема, удаленно запускать приложения и многое другое.

Примечание 1: если подключить шлема Oculus Quest 2 к компьютеру в тот момент, когда запущено приложение SideQuest, на экране шлема появится экран подтверждения, запрашивающий разрешение на отладку шлема с устройства компьютера. Для правильной работы SideQuest необходимо ответить на запрос положительно.

Примечание 2: при настройке шлема Oculus Quest через приложение SideQuest все применяемые настройки действуют только в течение одной рабочей сессии шлема, т. е. до его перезагрузки или выключения. Если пользователь захочет получить доступ к неко-

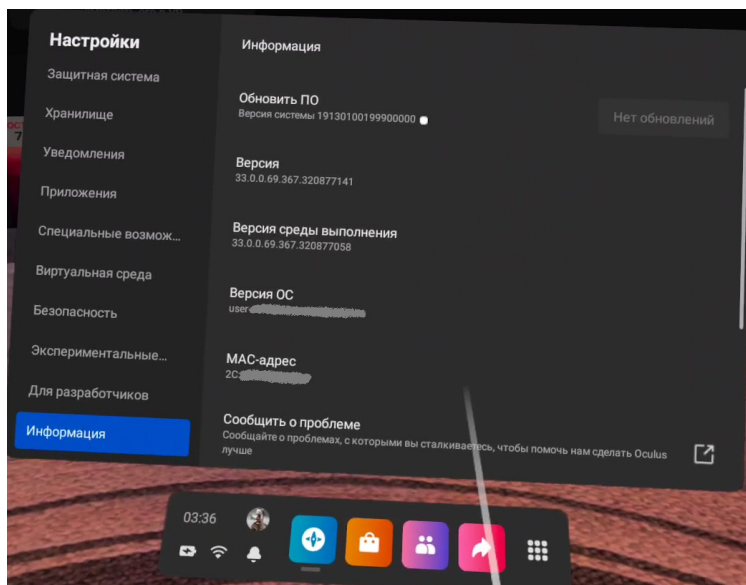


Рис. 2.40. Просмотр версии прошивки и поиск обновлений

торым функциям после выключения шлема, их придется активировать заново.

Примечание 3: внешний вид окружения в шлеме после запуска, а также вид панели, значков и элементов интерфейса, может отличаться в разных версиях прошивки шлема. Данное руководство было написано для обновления № 33. В случае, если шлем работает с более ранней версией прошивки, рекомендуется ее обновить. Проверить и обновить версию прошивки можно в приложении настроек, в категории «Информация» (параметр «Обновить ПО» и «Версия», рис. 2.40).

2.3.4. Использование шлема в приложениях без ПК

Для установки приложений необходимо подключиться к сети Wi-Fi. Подключение устанавливается через меню левого элемента нижней панели «Быстрые настройки» (рис. 2.41). Там необходимо выбрать пункт Wi-Fi (рис. 2.42) с одним из вариантов подключения и нажать «Включить».

После подключения к сети интернет, пользователю придется авторизоваться в своем аккаунте Facebook. При этом последующая смена аккаунта на устройстве производится только со сбросом всех

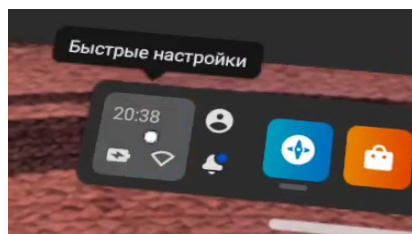


Рис. 2.41. Элемент панели «Быстрые настройки»

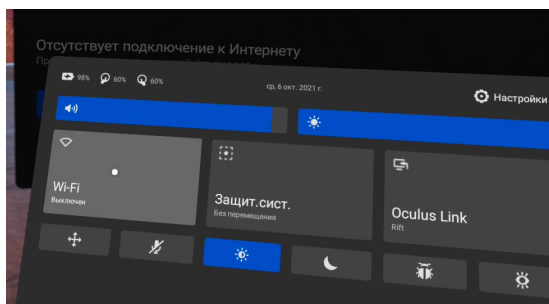


Рис. 2.42. Пункт «Wi-Fi»

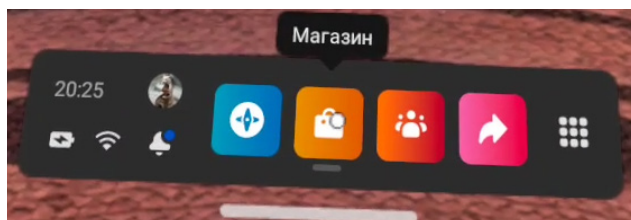


Рис. 2.43. Элемент нижней панели «Магазин»

настроек шлема до заводских и полной очистки внутренней памяти шлема.

Затем пользователь может зайти в элемент нижней панели «Магазин» (рис. 2.43), выбрать интересующее приложение (рис. 2.44), нажать на него и выбрать пункт «Получить» для бесплатных приложений (рис. 2.45) или пункт «Купить» для платных. Также для платных приложений в магазине будет отображаться цена.

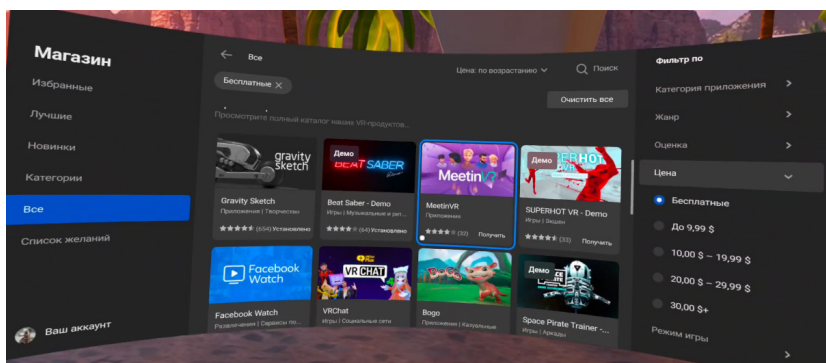


Рис. 2.44. Выбор приложения «MeetinVR» в магазине

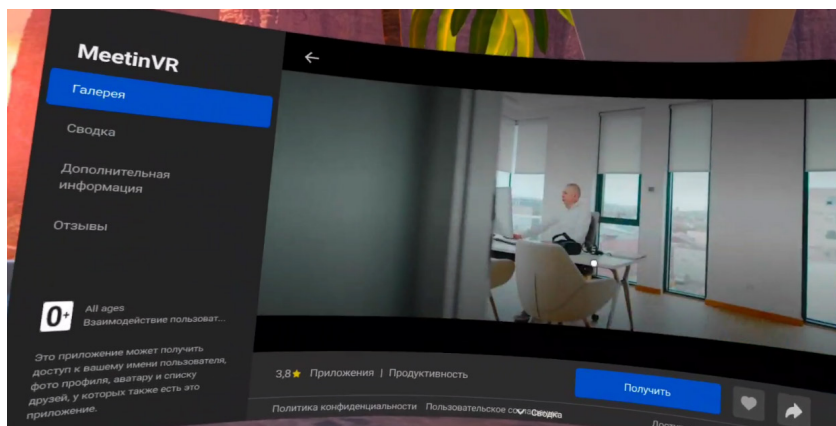


Рис. 2.45. Страница выбранного в магазине приложения

Полученное или купленное приложение будет записано в список приложений аккаунта, а кнопка «Получить»/«Купить» сменится индикатором загрузки (рис. 2.46).

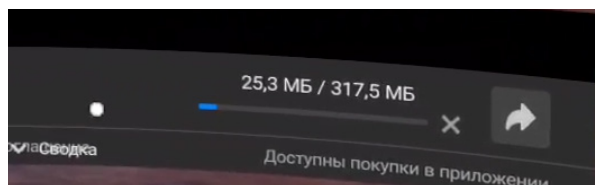


Рис. 2.46. Загрузка приложения

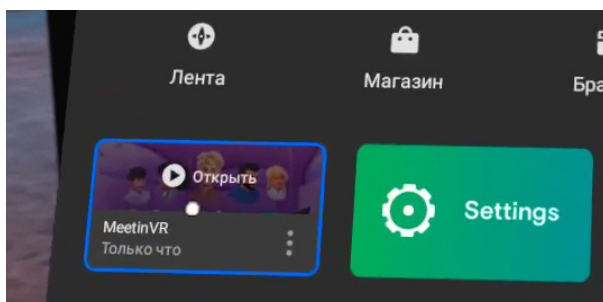


Рис. 2.47. MeetinVR в меню «Приложения»

После скачивания приложения индикатор загрузки сменится на кнопку «**Запустить**», а само приложение появится в меню «Приложения» нижней панели, в том же списке, что и приложение «Настройки» (рис. 2.47)

При запуске приложения в первый раз система может запросить дополнительные разрешения. Например, приложение «**MeetinVR**» использует микрофон для общения с другими пользователями, поэтому система выдаст уведомление с просьбой разрешить записывать звук (рис. 2.48). Для полноценного использования приложения необходимо дать ему соответствующие разрешения, нажав на кнопку «**Разрешить**».

При возникновении необходимости изменить разрешения приложения необходимо перейти в «Настройки», в категорию «Прило-

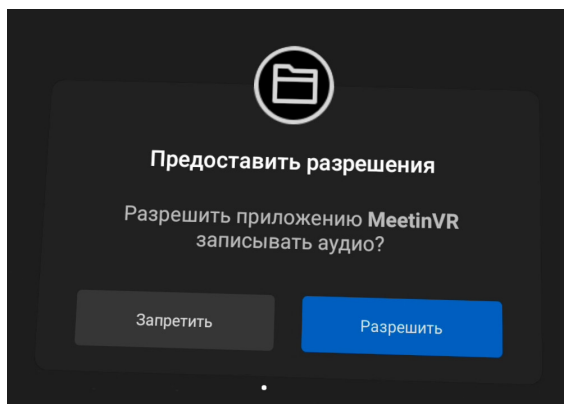


Рис. 2.48. Запрос разрешений в приложении

жения», и в открывшемся списке выбрать интересующее приложение. После этого для данного приложения отобразится список решений, которые можно будет изменить.

2.3.5. Подключение шлема к ПК по проводу

Для активации штатных средств ОС использования шлема Oculus Quest 2 (функции Oculus Link), которые позволят использовать его в качестве периферийного устройства для ПК, необходимо скачать приложение Oculus (ссылка: <https://www.oculus.com/setup/>).

Установка приложения Oculus возможна только на ПК, работающие под управлением ОС Windows 10 с разрядностью x64. Само приложение не только выполняет функцию подключения шлема к ПК, но также является магазином с библиотекой приложений и игр, новостным порталом и площадкой для общения. Для использования приложения потребуется наличие аккаунта Facebook. Внешний вид приложения после его запуска представлен на рис. 2.49.

Для подключения шлема к ПК необходимо соединить его с компьютером кабелем USB-C, после чего нажать в шлеме на элемент нижней панели «Быстрые настройки». Окно быстрых настроек пополнится дополнительным пунктом «Oculus Link» (рис. 2.50).

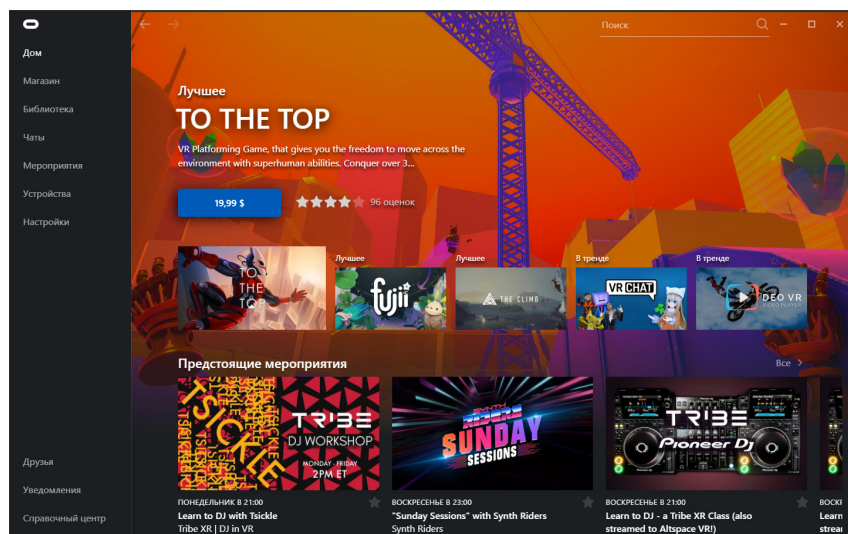


Рис. 2.49. Приложение Oculus на ПК

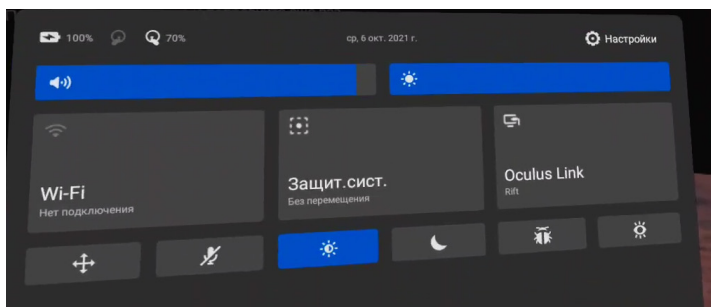


Рис. 2.50. Окно быстрых настроек после подключения провода USB-C

На этом этапе в шлеме может появиться окно, запрашивающее у пользователя разрешение на отладку и передачу данных с компьютера. Для корректной работы шлема необходимо выдать это разрешение. Теперь при открытии вкладки «Устройства» в приложении Oculus на компьютере в списке устройств отобразится подключенный шлем (рис. 2.51).

Примечание: в прошлых версиях прошивки шлема соединение по кабелю «Oculus Link» находилось в стадии тестирования. Для его запуска требовалось активировать параметр «Oculus Link» на

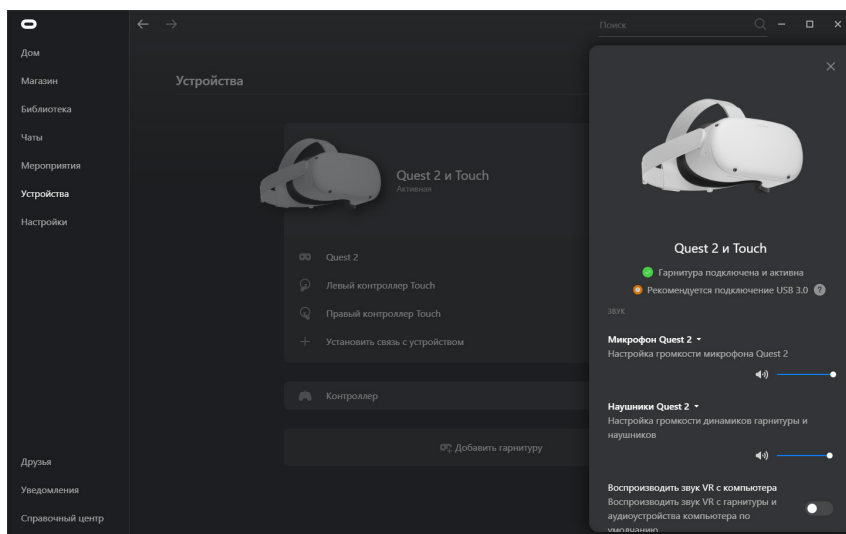


Рис. 2.51. Отображение подключенного шлема в приложении Oculus на ПК

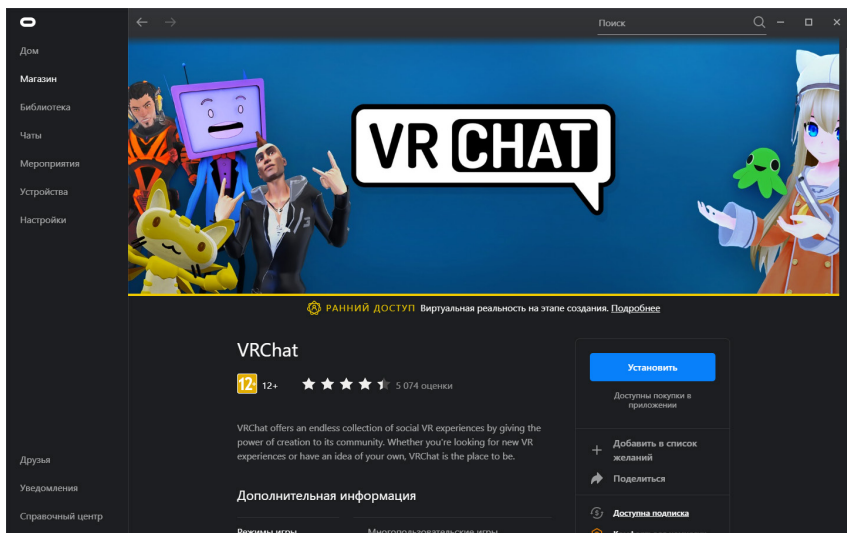


Рис. 2.52. VRChat в магазине Oculus на ПК

шлеме, в приложении «Настройки», в категории «Экспериментальные функции», и в приложении Oculus на ПК, во вкладке «Настройки», в верхнем пункте меню «Бета». Причем это соединение оставалось активным до отключения компьютера или шлема, либо в течение 24 часов.

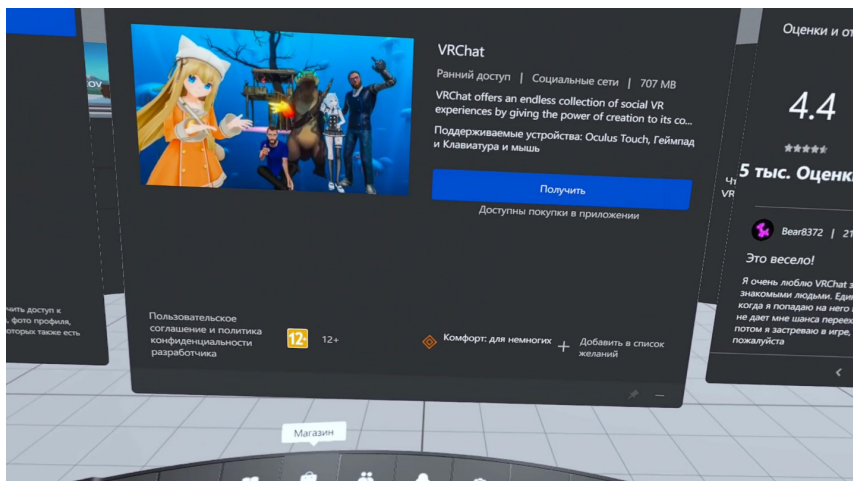


Рис. 2.53. VRChat в магазине Oculus в шлеме

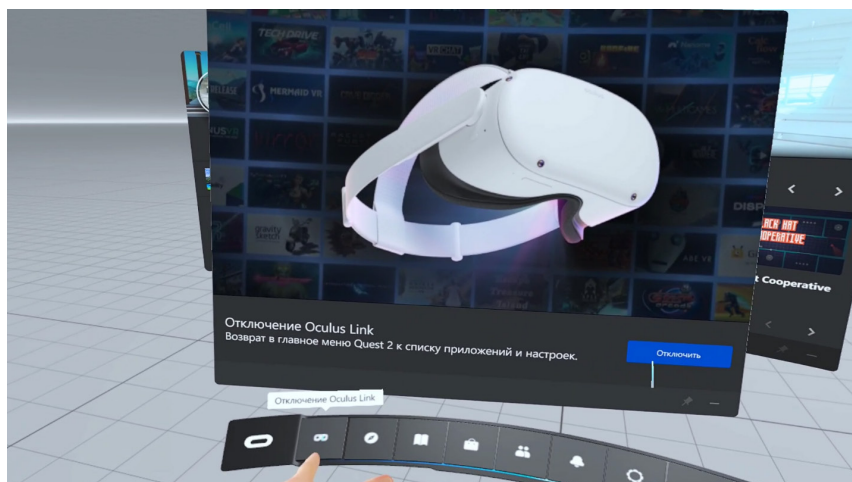


Рис. 2.54. Выход из приложения Oculus в шлеме

Теперь в окне быстрых настроек шлема можно выбрать пункт «Oculus Link». Перед пользователем откроется приложение Oculus со списком различных сторонних приложений и игры. Причем устанавливать их можно как с компьютера (рис. 2.52), так и со шлема, во вкладке «Магазин» нижней панели приложения Oculus (рис. 2.53).

Установленные приложения можно просмотреть и запустить из библиотеки. Для этого на ПК в приложении Oculus нужно открыть вкладку «Библиотека», а в шлеме вкладку «Магазин» нижней панели приложения Oculus.

Выход из приложения Oculus осуществляется нажатием вкладки «Отключение Oculus Link» на нижней панели, а затем нажатием на синюю кнопку «Отключить» в открывшемся окне (рис. 2.54).

2.3.6. Подключение шлема к ПК через Air Link

Подключаться к приложению Oculus на ПК можно не только по проводному соединению, но и используя беспроводное соединение Wi-Fi (функция Oculus Air Link). Для этого в шлеме необходимо перейти в приложение «Настройки», в категорию «Экспериментальные функции», и включить «Air Link». На компьютере в приложении Oculus необходимо перейти во вкладку «Настройки», пункт верхнего меню «Бета», и активировать параметр «Air Link» (рис. 2.55).

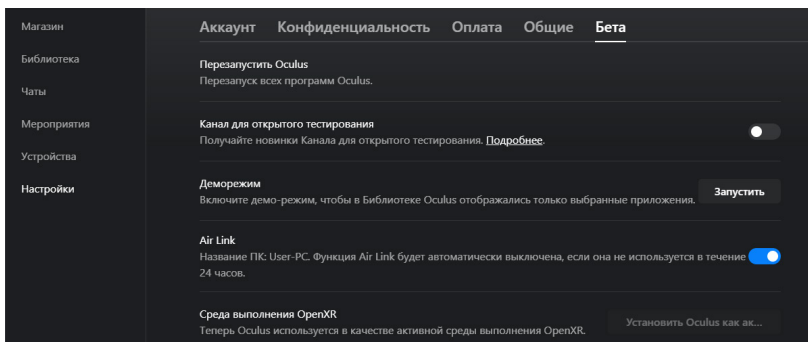


Рис. 2.55. Активация функции «Air Link» приложения Oculus на ПК

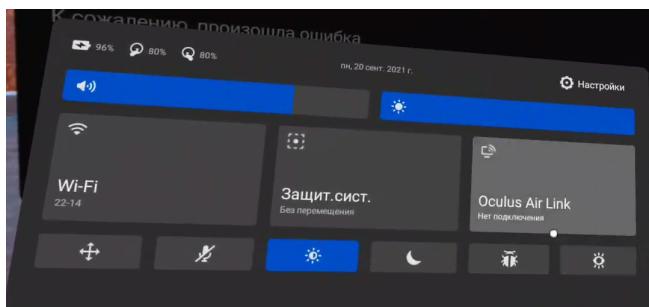


Рис. 2.56. Активация функции «Air Link» приложения Oculus в шлеме

Теперь при подключении шлема по Wi-Fi к той же сети, где находится ПК, на шлеме при открытии пункта нижней панели «Быстрые настройки» станет доступен параметр «Air Link» (рис. 2.56).

Примечание: для работы функции «Air Link» необходимо, чтобы шлем Oculus Quest 2 и ПК находились в одной сети. Рекомендуется использовать высокоскоростное подключение Wi-Fi с частотой 5 ГГц, однако работа допускается и на скорости 2,4 ГГц.

После выбора пункта «Air Link» появится окно доступных к подключению ПК. Там необходимо выбрать интересующий ПК и нажать сначала кнопку «Связать», а затем кнопку «Запустить». На шлеме запустится приложение Oculus, точно так же, как и при использовании проводного соединения «Oculus Link». Дальнейшая работа с приложением не отличается от работы с проводным соединением.

2.3.7. Запуск приложения SteamVR

Для работы шлема Oculus Quest 2 с библиотекой приложений Steam и приложением SteamVR (которые были описаны в пп. 2.1.4 и 2.1.5) необходимо подключить шлем Oculus Quest 2 к ПК любым доступным способом, как это было описано в пп. 2.3.5 и 2.3.6. Также на компьютере нужно запустить клиент Steam и внутри него – приложение SteamVR (рис. 2.57).

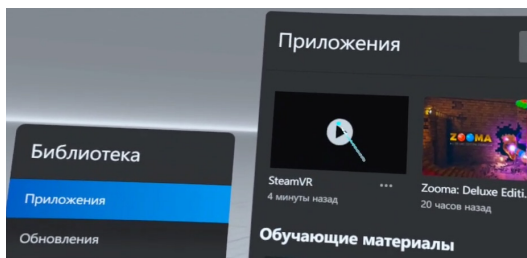


Рис. 2.57. Запуск SteamVR из приложения Oculus в шлеме

Окружение и интерфейс Oculus сменятся на окружение и интерфейс SteamVR, а в нижнем правом углу экрана появится информационное окно «Запущено» со статусом подключения (рис. 2.58).

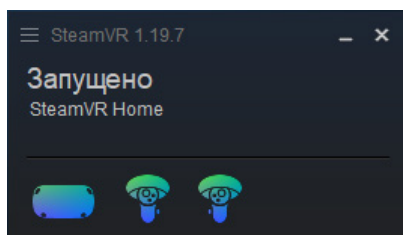


Рис. 2.58. Статус работы SteamVR со шлемом Oculus Quest 2

В случае, если статус подключения выглядит иначе, необходимо перезапустить приложение SteamVR.

2.3.8. Подключение шлема к ПК через Virtual Desktop

Использование приложения Virtual Desktop – это альтернативный метод подключения Oculus Quest 2 к компьютеру по Wi-Fi или проводу USB. Однако Virtual Desktop – платное приложение. Его придется сперва приобрести в магазине Oculus и установить на шлем.

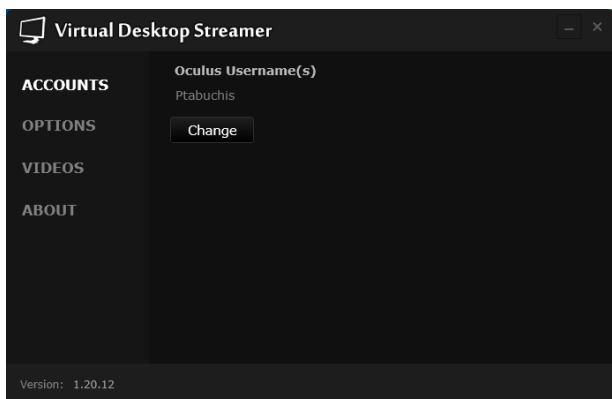


Рис. 2.59. Окно приложения Virtual Desktop Streamer после запуска на ПК

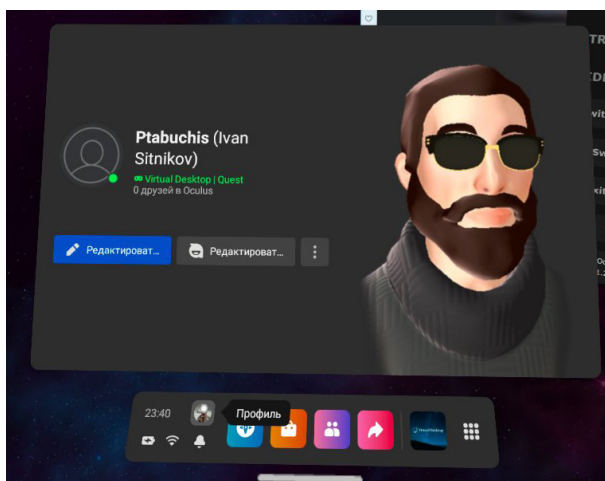


Рис. 2.60. Страница профиля пользователя шлема

Кроме того, пользователю понадобится скачать и установить бесплатное приложение Virtual Desktop Streamer на ПК (ссылка: <https://www.vrdesktop.net>). После его запуска во вкладке аккаунта в строке «Oculus Username(s)» (рис. 2.59) нужно указать такой же псевдоним, как и в профиле на нижней панели интерфейса шлема (рис. 2.60).

После запуска Virtual Desktop на шлеме откроется окно, где отобразятся доступные к подключению ПК, на которых запущено при-

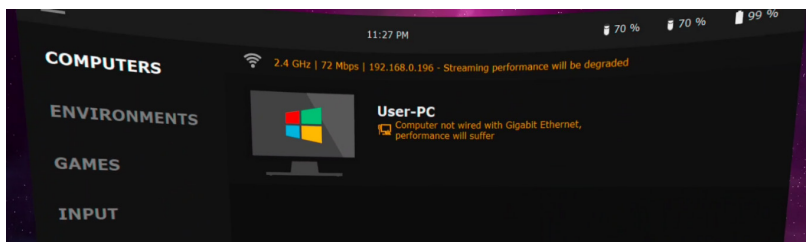


Рис. 2.61. Отображение доступных ПК для подключения

ложение Virtual Desktop Streamer с верно указанным псевдонимом пользователя (рис. 2.61). В данном случае, в отличие от подключения к ПК через Oculus Air Link, шлем и подключаемый ПК могут не иметь локального соединения.

После клика по доступному компьютеру с ним будет установлено соединение, и экран ПК начнет транслироваться на шлем.

Для управления компьютером со шлема пользователь может пользоваться контроллерами и виртуальной клавиатурой. При нажатии кнопки «Меню» на левом контроллере откроется боковое меню, где во вкладке «GAMES» будут перечислены все приложения, доступные в режиме VR как из приложения Oculus для ПК, так и из клиента Steam. Кроме того, внизу бокового меню расположена кнопка «Launch SteamVR», при нажатии на которую на компьютере запустится приложение SteamVR, и шлем автоматически синхронизируется с ним (рис. 2.62).

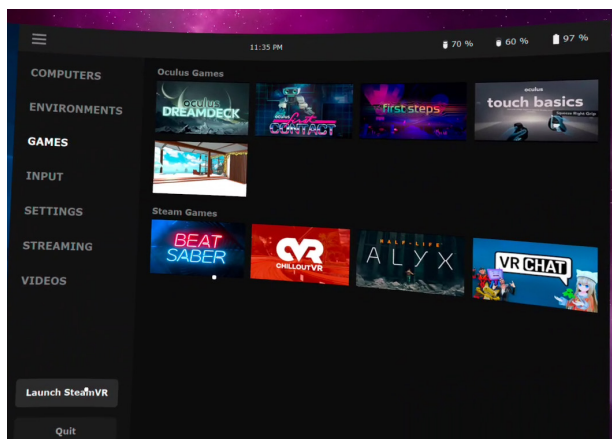


Рис. 2.62. Боковое меню Virtual Desktop

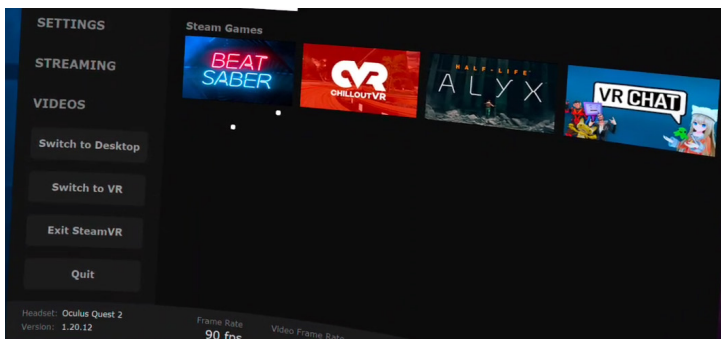


Рис. 2.63. Боковое меню Virtual Desktop после запуска SteamVR

При повторном вызове бокового меню Virtual Desktop на месте кнопки запуска SteamVR появятся три другие кнопки (рис. 2.63):

- «**Switch to Desktop**» – переключает вывод шлема на рабочий стол ПК;
- «**Switch to VR**» – переключает вывод шлема на окружение SteamVR или запущенное VR-приложение;
- «**Exit SteamVR**» – закрывает SteamVR и возвращает кнопку «Launch SteamVR», что полезно для перезагрузки SteamVR.

2.3.9. Работа с Oculus Quest 2 в редакторе Unity

Для работы в шлеме Oculus Quest 2 в редакторе Unity сперва необходимо запустить SteamVR, как это было ранее описано в пп. 2.3.7

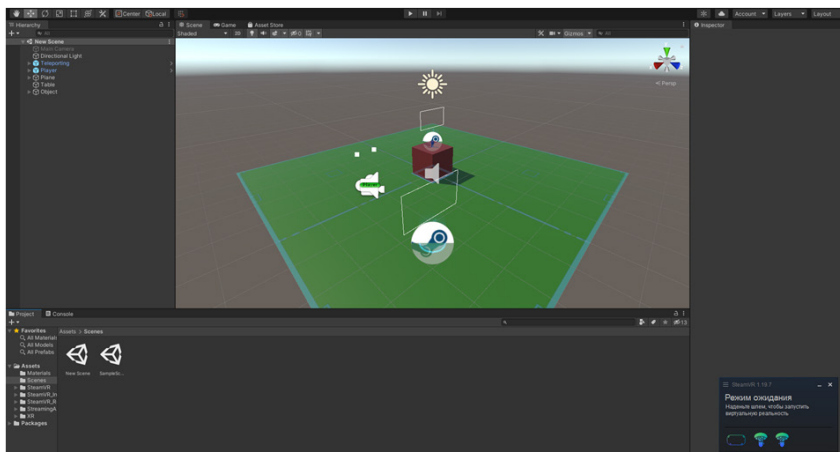


Рис. 2.64. Подготовленный проект Unity для работы с Oculus Quest 2

и 2.3.8. Статус подключения SteamVR должен показывать «Запущено». Затем необходимо создать или запустить проект в редакторе Unity, как было ранее описано в п. 2.1.8. На рис. 2.64 показан вид проекта с настроенным персонажем и плоскостью телепортации.

Для корректной работы потребуется сконфигурировать файлы настроек контроллеров Oculus Touch. Остальная же логика работы с приложением повторяет таковую для VIVE Pro.

Для конфигурации контроллеров необходимо открыть пункт меню «Window → SteamVR Input» и там нажать на кнопку «Open binding UI». На экране появится список раскладок для всех контроллеров, запущенных в данном проекте. Здесь нужно выбрать текущую раскладку «Oculus Touch» и нажать кнопку «Изменить» (рис. 2.65).

После этого откроется окно настроек контроллера Oculus Touch (рис. 2.66), аналогичное рассмотренному ранее окну настроек кон-

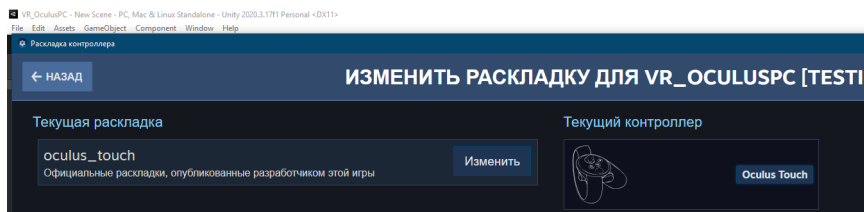


Рис. 2.65. Окно настроек SteamVR Input



Рис. 2.66. Окно настроек контроллера Oculus Touch

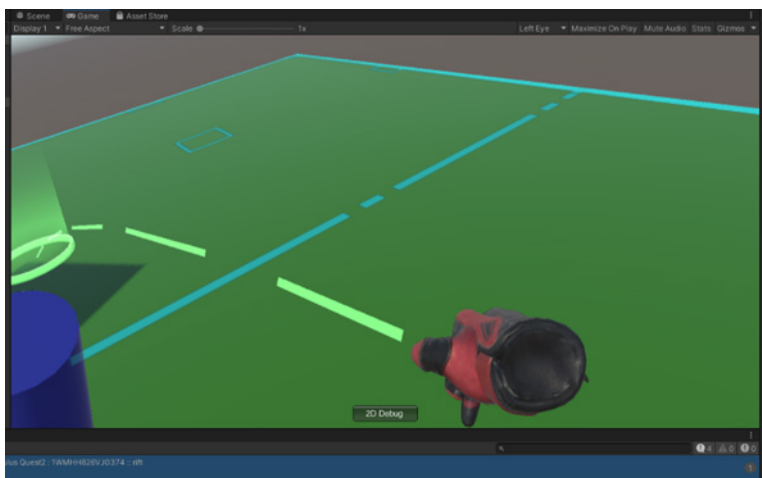


Рис. 2.67. Телепортация с удержанием объекта в руке

троллера VIVE (см. рис. 2.18). При желании здесь можно внести изменения в стандартную схему управления контроллеров. После завершения индивидуальной настройки надо нажать кнопку «Сохранить личную раскладку».

Затем в окне настроек SteamVR Input необходимо нажать кнопку «Save and generate», чтобы система сконфигурировала JSON-файл с сохраненными настройками. Если все выполнено верно, после запуска сцены пользователь сможет использовать контроллеры для выполнения различных манипуляций, например, хватания объектов или телепортации по сцене (рис. 2.67).

2.4. Дополненная реальность на базе ARCore

ARCore – это платформа Google для разработки приложений дополненной реальности (AR) под мобильные устройства, функционирующие под управлением операционной системы Android (например, смартфоны, планшеты, «умные» очки). Некоторые API-интерфейсы также доступны и для iOS.

ARCore использует три ключевые возможности для интеграции виртуального контента с реальным миром, видимым через камеру телефона:

– *Отслеживание движений* – позволяет телефону определять свое положение в мировом пространстве и отслеживать свое перемещение.

– *Понимание окружающей среды* – позволяет телефону определять размер и положение всех горизонтальных, вертикальных и угловых поверхностей, вроде пола, стен или предметов интерьера.

– *Оценка освещенности* – позволяет телефону проводить примерную оценку текущих условий освещенности в помещении или на открытой местности.

2.4.1. Принципы работы ARCore

ARCore выполняет две основные задачи: отслеживает положение мобильного устройства при его перемещении и выстраивает свое собственное понимание окружающего мира [73].

Технология отслеживания движения ARCore задействует камеру смартфона для идентификации точек интереса и следит за их перемещением с течением времени. Комбинируя данные о перемещении этих точек с показаниями внутренних датчиков смартфона, ARCore определяет положение и ориентацию смартфона в пространстве.

В дополнение к идентификации ключевых точек ARCore также умеет определять плоские поверхности, например, пол или столешницу, и приблизительно оценивать средний уровень освещенности в окружающей зоне. Все эти способности ARCore использует для формирования собственного понимания окружающего мира. В итоге это дает пользователю возможность размещать в пространстве объекты, комментарии или другую мультимедиаинформацию, которая плавно интегрируется с реальным миром. Например, при помощи ARCore можно «поставить» вазу на стол или вывести справочную информацию о человеке при наведении камеры на его фотографию. Технология отслеживания движений позволяет в буквальном смысле обойти виртуальный объект со всех сторон или даже выйти из комнаты. Но когда пользователь вернется обратно, он снова увидит объект на том же самом месте, поскольку система запомнит положение и ориентацию.

Комплекты разработки ARCore под различные программные пакеты, в том числе Unity, предоставляют собственные API для реализации всех популярных функций дополненной реальности, вроде отслеживания движения, понимания окружающей среды и оценки освещенности.

2.4.2. Принцип работы маркерной дополненной реальности

Интерфейс программирования ARCore позволяет создавать приложения, которые фиксируют двумерные изображения в ми-

ровом пространстве и дополняют их трехмерными объектами или целыми сценами. Такие изображения называются *маркерами* дополненной реальности.

Пользователь создает набор маркеров и сохраняет их в библиотеку исходных изображений [74]. Система ARCore при помощи алгоритмов компьютерного зрения анализирует эти исходные (референсные) маркеры в черно-белой цветовой палитре и извлекает их характерные признаки. Эти признаки записываются в специальную базу данных изображений дополненной реальности. Затем во время работы приложения дополненной реальности ARCore сканирует окружающее пространство через камеру мобильного устройства и находит среди картинок реального мира запомненные признаки. Соответственно, по распознанным признакам ARCore находит исходные маркеры, после чего проводит оценку их положения, ориентации и размеров для правильного позиционирования проецируемых 3D-объектов. Одновременно ARCore может отслеживать до 20 изображений и хранить в базе данных информацию о признаках 1000 изображений. Признаки каждого изображения занимают в памяти примерно 6 кБ. Добавление нового изображения в базу данных в процессе работы приложения занимает примерно 30 мс.

При добавлении маркера пользователь также может указать его физические размеры для облегчения процесса детектирования. В противном случае ARCore попытается приблизительно оценить размеры маркера при помощи своих алгоритмов. Детектирование работает как для неподвижных, так и для анимированных/движущихся маркеров. Причем даже если маркер на короткое время выйдет из области видимости камеры мобильного устройства, трекинг не нарушится, поскольку в таком случае система посчитает, что положение и ориентация маркера за это время не изменились.

2.4.3. Требования к маркерам

Для корректной работы системы трекинга ARCore маркеры должны:

- быть нанесены на плоскую поверхность;
- заполнять не менее 25% площади кадра камеры при первом обнаружении;
- оставаться в поле зрения камеры;
- быть хорошо освещены и не иметь сильного наклона относительно камеры.

Отсюда можно выделить некоторые практические рекомендации по созданию маркеров и выбору эталонных изображений:

– Разрешение изображения должно быть не менее 300×300 пикселей.

– Эталонные изображения могут быть представлены в форматах .PNG или .JPEG.

– Цвет изображения не имеет значения, поскольку алгоритм трекинга преобразует их в оттенки серого.

– Не стоит использовать изображения с высокой степенью сжатия.

– Не стоит использовать изображения с большим количеством однотипных геометрических признаков (например, штрих-коды) или с повторяющимися узорами (мозаики).

– Пакет разработки под ARCore включает в себя инструмент `arcoreimg`, который оценивает изображения по шкале от 0 до 100 баллов. Рекомендуется использовать изображения с оценкой не ниже 75 баллов.

2.4.4. Поддерживаемые ARCore устройства

Полный список поддерживаемых мобильных устройств включает несколько сотен наименований и доступен по ссылке: <https://developers.google.com/ar/devices?hl=en>.

Для работы ARCore мобильное устройство должно удовлетворять нескольким требованиям:

1. Иметь установленные «Сервисы Google Play для AR». Загрузить и установить их можно из магазина Google Play Store по ссылке: <https://play.google.com/store/apps/details?id=com.google.ar.core>.

Примечание: на поддерживаемых устройствах сервисы Google Play для AR устанавливаются и обновляются автоматически.

2. Работать на Android версии не ниже 7.0.

3. Изначально поставляться с предустановленным магазином Google Play Store.

4. Версия SDK для приложений с тегом «AR Required» должна быть заявлена не ниже API Level 14.

5. Версия SDK для приложений с тегом «AR Required» должна быть заявлена не ниже API Level 24.

2.4.5. Настройка смартфона для работы с ARCore

В данном разделе будет рассмотрена подготовка смартфона для работы с сервисами дополненной реальности ARCore. Для приме-

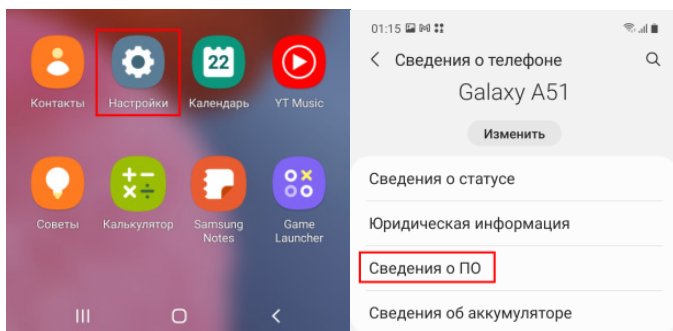


Рис. 2.68. Переход в меню сведений о ПО

ра был взят смартфон Samsung Galaxy A51, поэтому приведенные ниже параметры и скриншоты актуальны для этой модели.

Примечание: в зависимости от конкретной модели смартфона вид настроек, расположение пунктов меню и порядок действий могут отличаться.

На первом этапе подготовки необходимо перевести смартфон в режим разработчика. Для этого надо зайти в «Настройки» через главное меню (значок шестеренки), перейти в категорию «Сведения о телефоне», а оттуда в раздел «Сведения о ПО» (рис. 2.68).

Для включения режима разработчика достаточно несколько раз нажать на пункт «Номер сборки». После этого следует вернуться в «Настройки», выбрать там вкладку «Параметры разработчика» и активировать в ней «Отладку по USB» (рис. 2.69).

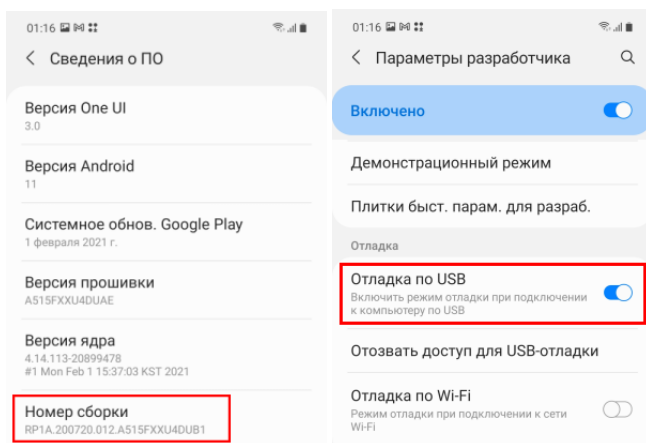


Рис. 2.69. Активация отладки по USB

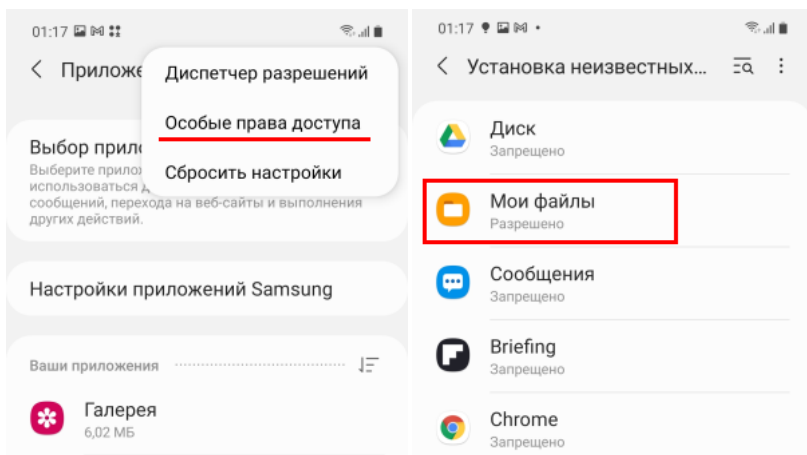


Рис. 2.70. Настройка прав для установки приложений

Теперь нужно разрешить файловому менеджеру телефона устанавливать неизвестные приложения. Это делается в разделе **«Приложения»** в настройках телефона. Надо нажать на символ меню в виде трех точек в правом верхнем углу (рис. 2.70), в открывшемся меню выбрать опцию **«Особые права доступа»** → **«Установка неизвестных приложений»** и разрешить установку неизвестных приложений для файлового менеджера **«Мои файлы»**.

После внесения всех перечисленных изменений на смартфон можно будет устанавливать любые сторонние APK-файлы через штатный файловый менеджер **«Мои файлы»**.

Примечание: скриншот (снимок экрана) Samsung Galaxy A51, будь то меню телефона или сеанс работающего приложения, можно сделать двумя способами:

- Нажав одновременно кнопку питания и кнопку уменьшения громкости. Причем это должно быть именно быстрое однократное нажатие без удержания клавиш.
- Проведя вдоль экрана ребром ладони.

Сделанные снимки экрана сохраняются в приложении **«Галерея»**.

Если функция создания снимка экрана проведением по нему ладонью недоступна, ее можно активировать в **«Настройках»** → **«Дополнительные функции»** → **«Движения и жесты»** → **«Снимок экрана ладонью»**.

2.4.6. Установка Android Studio

Android Studio – это официальная интегрированная среда разработки (IDE) для операционной системы Google Android [75]. Она основана на программном обеспечении JetBrains IntelliJ IDEA и доступна для Windows, macOS и Linux. В качестве языков программирования в Android Studio можно использовать C++, Java или Kotlin. Последний является предпочтительным с мая 2019 года. Текущая версия Android Studio предлагает широкий функционал, например:

- Сборка проектов на основе Gradle.
- Расширенный редактор макетов WYSIWYG, позволяющий работать с UI при помощи функции «Drag-and-Drop».
- Рефакторинг кода и поддержка быстрых исправлений.
- Статический анализатор кода (Lint), позволяющий находить проблемы с производительностью и несовместимостью версий.
- Встроенные механизмы защиты ProGuard и утилита для подписи приложений.
- Шаблоны разработки для основных макетов и компонентов Android.
- Поддержка разработки приложений для Android Wear и Android TV.
- Встроенная поддержка платформы Google Cloud, интеграция с Firebase Cloud Messaging и движком Google App.

Загрузить и установить Android Studio можно бесплатно с официального сайта: <https://developer.android.com/studio#downloads>. При первом запуске студия попросит установить **Android SDK** (рис. 2.71).

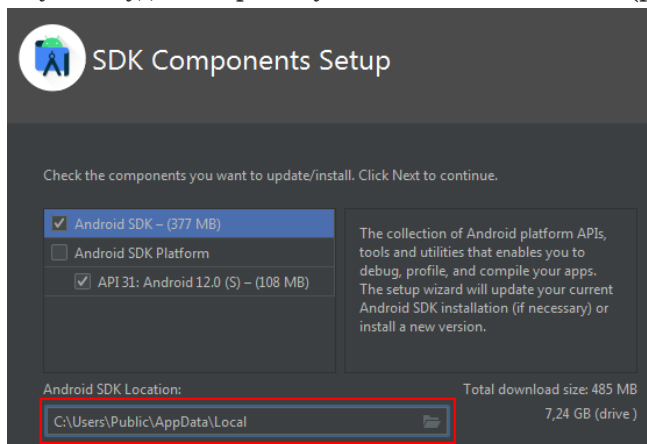


Рис. 2.71. Установка компонентов Android SDK

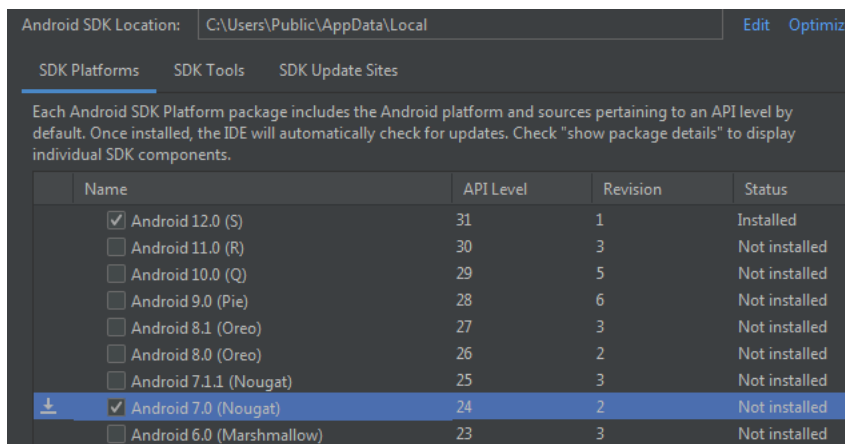


Рис. 2.72. Установка Android 7.0 (Nougat)

Внимание: путь к Android SDK не должен содержать русских букв!

По умолчанию устанавливается наиболее актуальная версия, но при необходимости к установке можно добавить и другие релизы (например, для разработки AR-приложений под смартфоны с Android 7.0). Это делается при помощи менеджера пакета разработка (**SDK Manager**, доступный в меню «System Settings → Android SDK»). Для работы с AR Foundation понадобится установить Android SDK 7.0 (API Level 24) или более свежую версию (рис. 2.72).

2.4.7. Добавление поддержки сборки под Android в Unity

Для создания и сборки проектов Unity с поддержкой ARCore потребуется некоторое дополнительное программное обеспечение. К счастью, разработчики Unity уже позаботились об этом, и все необходимое можно установить вместе с редактором Unity через приложение **Unity Hub**. Достаточно при выборе желаемой версии редактора в разделе установок (в данном случае это **Unity 2020.3 (LTS)**) отметить галочками поддержку сборки под Android (**Android Build Support**, рис. 2.73). Тогда вместе с редактором на компьютер также будут установлены инструменты Android SDK & NDK, OpenJDK и Gradle.

OpenJDK – полностью совместимый аналог Java Development Kit, состоящий исключительно из свободного и открытого исходного кода.

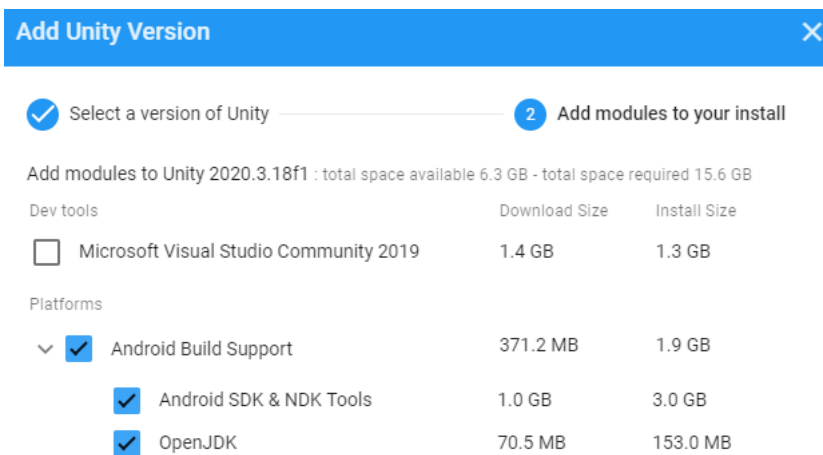


Рис. 2.73. Добавление поддержки сборки под Android

Android NDK (Native Development Kit) – это набор инструментов, которые позволяют реализовать отдельные элементы Android-приложения с использованием языков C и C++ [76]. При этом Google рекомендует прибегать к использованию NDK только в крайних случаях, например, когда требуется повысить производительность приложения, при подключении сторонних библиотек или программировании на низком уровне.

Gradle – это инструмент для автоматизации процесса сборки (компиляции) приложений Android, написанных на различных языках программирования (Java (Kotlin, Groovy, Scala), C/C++, JavaScript). Gradle построен на принципах Apache Ant и Apache Maven, но в отличие от них использует направленный ациклический граф для определения порядка выполнения задач [77]. Основная сфера применения Gradle – создание расширяемых многопроконтных сборок, в том числе с поддержкой каскадной модели разработки (Waterfall).

2.4.8. Подготовка проекта Unity для поддержки ARCore

Поддержка ARCore на движке Unity 2020 или в более новых версиях осуществляется через кроссплатформенный фреймворк **AR Foundation**. Он позволяет собирать проекты для устройств на базе Android и iOS без внесения каких-либо дополнительных изменений в код.

Подготовка проекта Unity для работы с ARCore происходит в несколько этапов. При этом важно устанавливать именно указанные версии плагинов и расширений во избежание проблем совместимости.

1. *Установка пакета дополнений ARCore и его зависимостей.* Свежая версия пакета **arcore-unity-extensions-*.tgz** доступна для скачивания с GitHub по ссылке: <https://github.com/google-ar/arcore-unity-extensions/releases/>. Для работы с ARCore в Unity ее необходимо скачать и сохранить в пользовательскую папку с документами или в папку загрузок.

Примечание: при сохранении пакета расширений ARCore на диск D он может не установиться в Unity из-за отсутствия прав доступа к файлу.

Чтобы добавить дополнения ARCore в Unity, нужно открыть менеджер пакетов («**Window** → **Package Manager**»), там нажать на значок «+» в верхней левой части экрана и выбрать пункт «**Add package from tarball...**» (добавить пакет из tar-архива, рис. 2.74). Откроется окно выбора файла, где потребуется указать путь к архиву «**arcore-unity-extensions-*.tgz**».

Следующие плагины будут загружены и добавлены в проект автоматически после установки пакета **ARCore Extensions**:

- XR Plugin Management 3.2.16;
- XR Legacy Input Helpers 2.1.4;
- AR Foundation 4.1.0-preview.10 (версии новее 4.1.1 могут не работать);
- ARCore XR Plugin 4.1.0-preview.10 (версии новее 4.1.1 могут не работать).

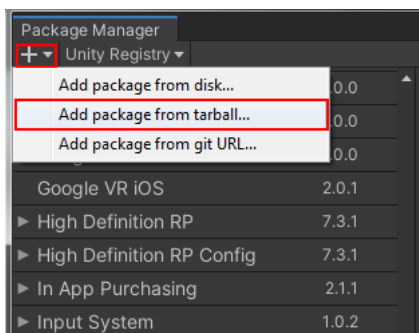


Рис. 2.74. Установка пакета из архива

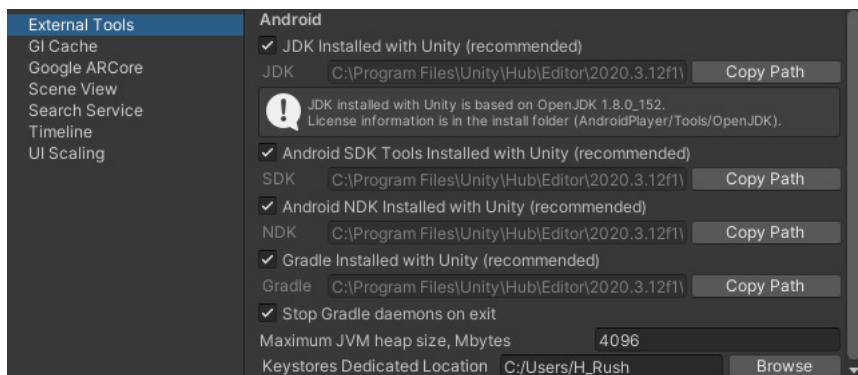


Рис. 2.75. Установка пакета из архива

Внимание: при использовании других версий плагинов могут наблюдаться проблемы совместимости!

2. *Проверка готовности всех дополнений.* Перед настройкой сборки полезно проверить состояние всех установленных дополнительных компонентов, пакетов и плагинов. Если открыть окно «**Edit** → **Preferences** → **External Tools**», там можно обнаружить пути к **JDK**, **Android SDK/NDK** и **Gradle** (рис. 2.75). Unity предпочитает использовать собственные версии вышеперечисленных средств разработки, которые он устанавливает в директорию редактора. В случае возникновения каких-либо проблем движок выдаст предупреждение в виде желтого треугольника с восклицательным знаком. Обычно это происходит, когда версии установленных компонентом отличаются от стандартных, либо они располагаются в других директориях.

Для работы с ARCore используется версия **Java SE Development Kit 8 (JDK 1.8.0_231)**. Если она не установилась автоматически вместе с редактором Unity, ее можно скачать вручную по адресу: <https://mega.nz/folder/Eo1CyShY#D0QUZdcafG1mUXbqWywFsg> (учетная запись Oracle при этом не требуется).

Внимание: версия JDK 9 не поддерживается Unity!

3. *Настройки проекта Unity для сборки под Android.* После создания проекта в Unity необходимо сменить целевую платформу с Windows (активна по умолчанию) на Android. Это делается в окне настройки сборки («**File** → **Build Settings**») простым нажатием на кнопку «**Switch Platform**» (переключить платформу, рис. 2.76).

Теперь необходимо внести изменения в настройки плеера, чтобы адаптировать его под особенности ОС Android. Для этого в глав-

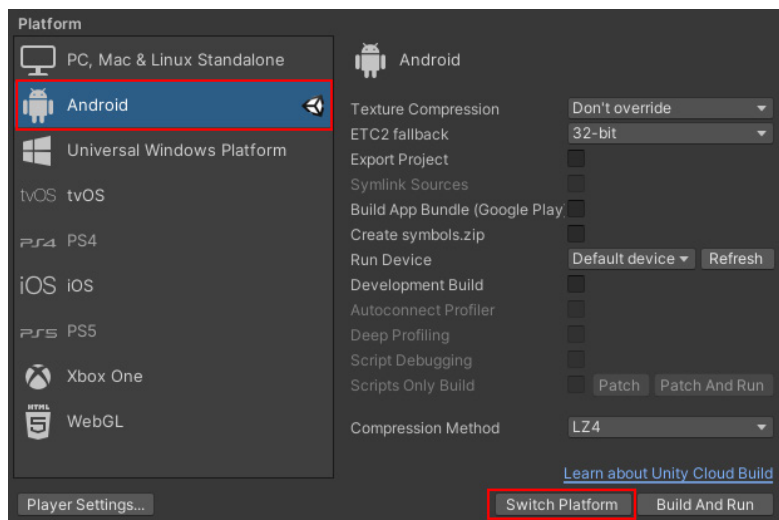


Рис. 2.76. Смена целевой платформы

ном меню требуется перейти в раздел «File → Build Settings» и там нажать кнопку «Player Settings» в нижнем левом углу окна. В табл. 2.11 указаны изменения, которые необходимо внести для корректного процесса сборки проектов Unity под Android.

Таблица 2.11

Настройки плеера Unity для ARCore

Пункт настроек	Значение и действие
OtherSettings	При использовании AR Foundation 2.1 или 3.1 необходимо снять галочку с пункта Multithreaded Rendering во избежание проблем при рендеринге
Other Settings → Rendering	Снять галочку с пункта Auto Graphics API . Если в категории Graphics API присутствует Vulkan , его необходимо удалить
Other Settings → Package Name	Требуется задать уникальный ID приложения в системе имен Java . Например: com.example.helloAR
Other Settings → Minimum API Level	Если приложение требует обязательного использования AR, здесь стоит выбрать Android 7.0 'Nougat' (API Level 24) или более новую версию. Если использование AR в приложении является опциональным, можно выбрать Android API Level 14 или более новую версию

Пункт настроек	Значение и действие
Other Settings -> Scripting Backend	Для поддержки архитектуры ARM64 необходимо установить IL2CPP вместо Mono
Other Settings -> Target Architectures	Требуется включить поддержку ARM64 (64-bit ARM) . Также ради сохранения совместимости с 32-битными устройствами стоит оставить галочку в пункте ARMv7 (32-bit ARM)

4. *Активация плагина XR и настройка ARCore.* Следующим шагом нужно активировать поддержку плагина **Android XR Plug-in** для работы с дополненной реальностью. Именно этот плагин добавляет поддержку ARCore в Unity и реализует функционирование таких подсистем как сессия, камера, захват глубины, пользовательский ввод, плоскости проецирования, отбрасывание лучей, якоря, отслеживание лиц и изображений (маркеров), захват данных об окружающей среде и окклюзии.

Плагин активируется через раздел **«Edit -> Project Settings -> XR Plug-in Management»** в главном меню (рис. 2.77). Там необходимо выбрать платформу **Android** и поставить галочку в пункте **«ARCore»** в разделе **«Plug-in Providers»**. Также будет полезно поставить галочку **«Initialize XR on Startup»**, чтобы плагин запускался сразу после открытия проекта Unity.

Последний этап подготовки к работе – это конфигурирование ARCore в разделе **«Edit -> Project Settings -> XR Plug-in**

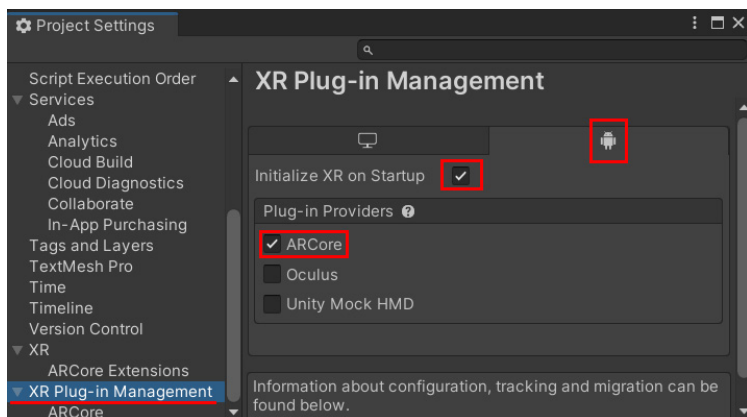


Рис. 2.77. Настройка плагина XR

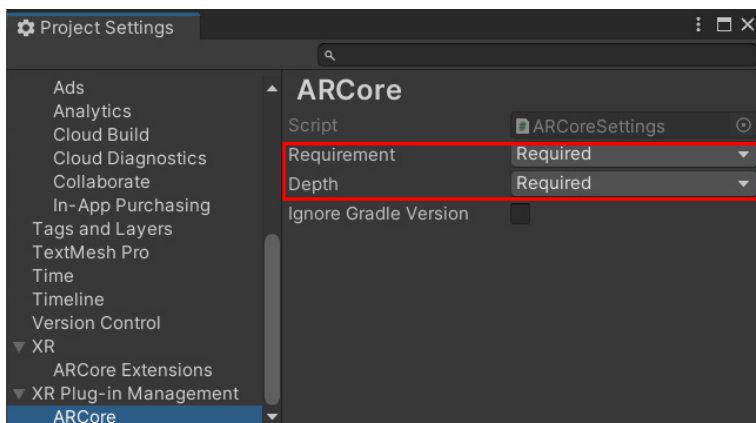


Рис. 2.78. Настройка ARCore

Management → **ARCore**» главного меню. Здесь требуется указать, является ли использование дополненной реальности в разрабатываемом приложении обязательным (пункт «**AR Required**») или опциональным (пункт «**AR Optional**») [78]. Также тут указывается, является ли использование **Depth API** в проекте обязательным или опциональным (рис. 2.78).

Depth API (или интерфейс глубины) помогает камере устройства понимать размер и форму реальных объектов в сцене. Он создает изображения глубины (карты глубины), которые затем можно использовать при разработке иммерсивного и реалистичного пользовательского интерфейса.

После применения всех описанных настроек можно, наконец, перейти к разработке мобильных приложений с поддержкой ARCore.

2.4.9. Дополненная реальность на основе маркера

Первым делом после создания нового проекта Unity с пресетом 3D необходимо перейти в иерархию сцены и из контекстного меню (ПКМ) добавить два объекта:

- «**XR** → **AR Session Origin**» – устанавливает начало координат в дополненном окружении;
- «**XR** → **AR Session**» – инициализирует сессию дополненной реальности.

Поскольку приложение дополненной реальности будет работать с камерой смартфона, штатная камера Unity под названием «**Main Camera**» здесь не нужна. Вместо нее будет использоваться специ-

альная камера «AR Camera», находящаяся в объекте «AR Session Origin». Для удобства дальнейшей работы не лишним будет пометить ее тэгом «MainCamera» (рис. 2.79).

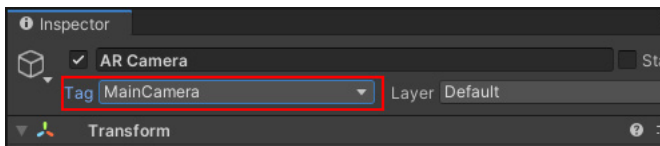


Рис. 2.79. Подготовка AR-камеры

Для работы с маркерами нужно в окне иерархии объекта «AR Session Origin» добавить к нему компонент «AR Tracked Image Manager», который как раз отвечает за распознавание маркеров. Сами маркеры удобно хранить в отдельной папке с названием «Markers» в ассетах проекта. Маркеры помещаются в библиотеку референсных изображений, поэтому ее тоже нужно будет создать в проекте (через пункт «XR → Reference Image Library»).

Библиотека переносится в поле «Serialized Library» компонента «AR Tracked Image Manager» (рис. 2.80). Здесь же в переменную «Tracked Image Prefab» записывается префаб трехмерного объекта, который будет проецироваться на маркер.

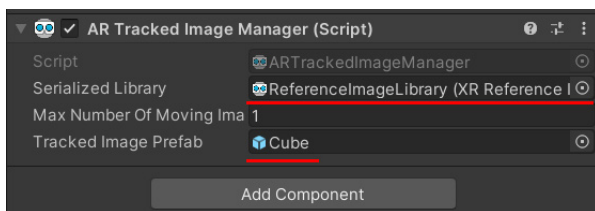


Рис. 2.80. Подключение библиотеки маркеров

Изображением для маркера может служить любая картинка разрешением не менее 300×300 пикселей и обладающая достаточным количеством неповторяющихся деталей. Если деталей на изображении будет слишком мало, ARCore не сможет выделить ключевые признаки, и выдаст ошибку при компиляции проекта. Картинка сохраняется в папке «Markers» и добавляется в библиотеку маркеров (рис. 2.81).

Важно: не следует использовать в качестве маркера примитивы (круги, квадраты, треугольники и т. п.), поскольку ARCore не сможет выделить у них достаточное количество признаков.

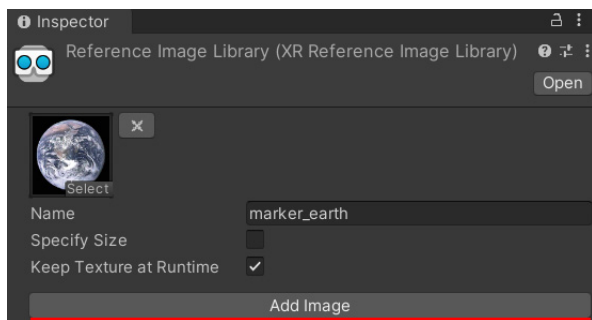


Рис. 2.81. Добавление маркера в библиотеку

Осталось сделать билд проекта и установить его на смартфон. Переслать APK-файл на смартфон можно по USB-кабелю, через локальное или облачное файловое хранилище или при помощи функции передачи файлов по Bluetooth. Сама установка ничем не отличается от локальной установки любого другого Android-приложения (рис. 2.82).

После запуска приложения необходимо разрешить ему использовать камеру телефона. Теперь, если все настройки были сделаны правильно, при наведении камеры на маркер (его можно отобразить на экране монитора или распечатать на бумаге) на нем должен появиться объект, указанный в переменной «**Tracked Image Prefab**» (рис. 2.83).

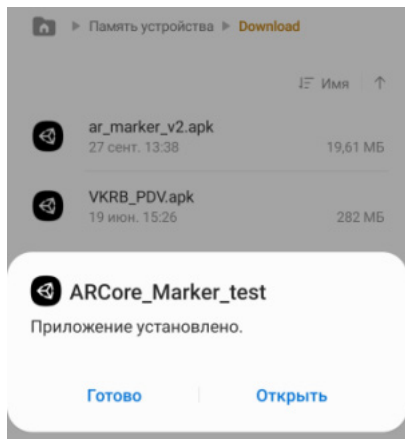


Рис. 2.82. Установка Android-приложения ARCore

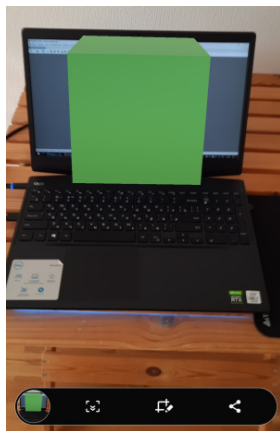


Рис. 2.83. Проецирование куба на маркер

2.4.10. Пространственное картирование в ARCore

Альтернативный вариант размещения объектов в дополненной реальности – использование пространственного картирования, когда система анализирует геометрию помещения на основе данных с камеры смартфона и строит «застилает» вертикальные и/или горизонтальные плоскости полигональными сетками.

За отображение этих сеток (а если точнее – плоскостей) отвечает компонент «**AR Plane Manager**». Для реализации пространственного картирования его будет необходимо добавить к объекту «**AR Session Origin**».

Для наглядности работы ARCore плоскость можно визуализировать. Для этого к пустому игровому объекту нужно применить скрипты «**AR Plane**» и «**AR Plane Mesh Visualizer**» (рис. 2.84). Первый из них определит игровой объект как плоскость для дополненной реальности, а второй визуализирует плоскость.

Кроме того, к плоскости потребуется добавить еще четыре базовых компонента:

– «**Mesh Filter**» – отвечает за построение геометрической сетки плоскости;

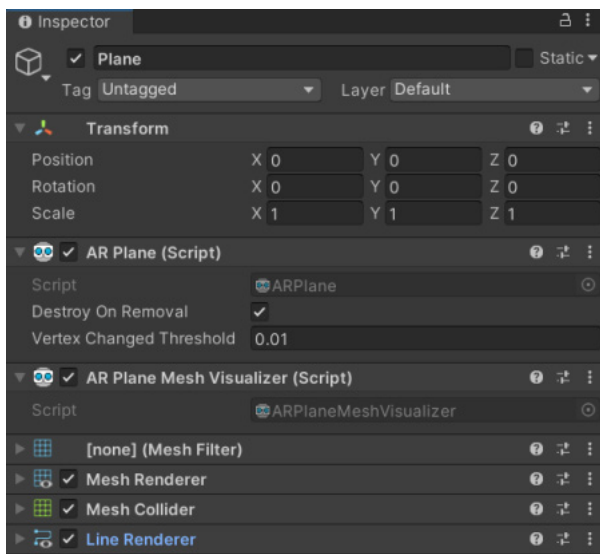


Рис. 2.84. Добавление компонентов для AR-плоскости

– «**Mesh Renderer**» – позволяет визуализировать плоскость (отображать на экране);

– «**Mesh Collider**» – добавляет физическую оболочку, необходимую для обработки коллизий плоскости с другими объектами;

– «**Line Renderer**» – обрисовывает границы плоскости линиями.

На полученную плоскость можно наложить полупрозрачный материал, после чего сохранить ее в префаб и перенести в компонент «**AR Plane Manager**» объекта «**AR Session Origin**».

Если теперь сделать билд проекта и установить полученное приложение на телефон, можно будет увидеть, как скрипт «**AR Plane Manager**» рисует на полу плоскости (рис. 2.85).

Для размещения объектов на этих плоскостях можно воспользоваться системой испускания лучей (Raycast). Простейший алгоритм можно описать следующим образом: скрипт определяет место и фазу касания пальцем экрана смартфона и испускает из точки касания луч в глубь сцены. Если луч пересекает одну из плоскостей дополненной реальности, то в месте их пересечения появляется трехмерный объект.

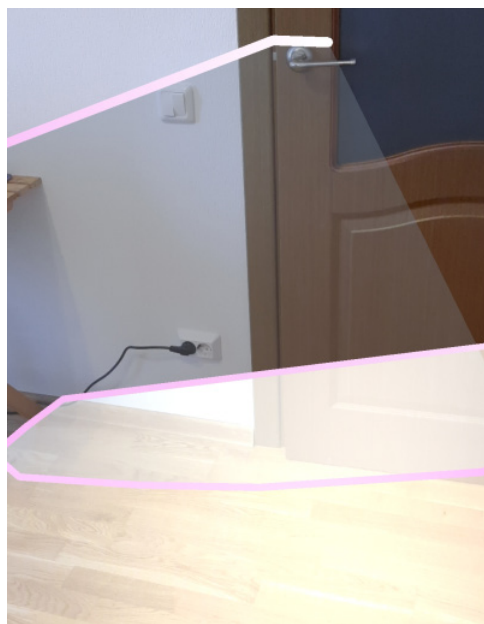


Рис. 2.85. Рисование плоскостей в дополненной реальности

Чтобы активировать систему рейкаста, на объект «AR Session Origin» необходимо добавить компонент (скрипт) «AR Raycast Manager», который как раз отвечает за испускание лучей и детектирование их столкновений с преградами. Для размещения объектов можно использовать готовый скрипт «SceneManager». Его также необходимо применить к объекту «AR Session Origin», а в переменных указать префаб объекта, который будет проецироваться в дополненной реальности.

Код скрипта «SceneManager.cs»

```
using System.Collections.Generic;
using UnityEngine;
// Библиотеки XR Foundation
using UnityEngine.XR.ARFoundation;
using UnityEngine.XR.ARSubsystems;

public class SceneManager : MonoBehaviour
{
    // Менеджер лучей
    private ARRaycastManager raycastScript;

    void Start()
    {
        raycastScript = GetComponent<ARRaycastManager>();
    }

    void PlaceObject()
    {
        // Если зафиксировано однократное нажатие на экран и фаза является
        // начальной
        if (Input.touchCount > 0 && Input.touches[0].phase == TouchPhase.
        Began)
        {
            // Переменная для определения момента касания экрана
            Touch tap = Input.GetTouch(0);

            // Контейнер для лучей
            List<ARRaycastHit> hits = new List<ARRaycastHit>();
            touchPosition = tap.position;

            // Запуск луча из точки касания экрана
            raycastScript.Raycast(touchPosition, hits, TrackableType.
            Planes);

            // Создание копии 3D-объекта
            Instantiate(primitiveObject, hits[0].pose.position,
            primitiveObject.transform.rotation);
        }
    }
}
```

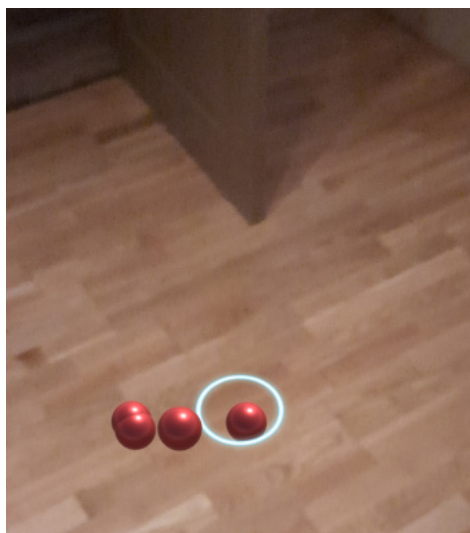


Рис. 2.86. Проецирование объектов на невидимой плоскости

```
    }  
  }  
  
  void Update()  
  {  
    PlaceObject();  
  }  
}
```

При корректной работе механизмов ARCore, определяющих горизонтальные поверхности в помещении, на полу реальной комнаты будут проецироваться трехмерные объекты (рис. 2.86).

ЗАКЛЮЧЕНИЕ

Методы и технологии ЦР имеют ряд основных преимуществ по сравнению с традиционными технологиями [24].

1. Целостное сенсорное восприятие, в том числе за счет трехмерной визуализации данных. Это биологически естественно для человеческого организма; позволяет исключить из когнитивного процесса необходимую стадию мысленного достраивания воспринимаемой картины и тем самым облегчает работу анализаторов, ускоряя процесс понимания и восприятия материала. Использование трехмерной графики способствует более реалистичному и детальному раскрытию тем, а также подразумевает больше точек зрения и возможностей интерпретации по сравнению с двухмерным изображением. В виртуальных и смешанных мирах с помощью полисенсорных стимулов, таких как трехмерное пространственное звучание или стимулы осязания (сила, вибрация), можно активизировать все человеческие возможности.

2. Непосредственное формирование индивидуального и коллективного опыта является одной из наиболее значимых особенностей виртуальных и смешанных миров. Во многих случаях взаимодействие посредством подобных миров может заменить непосредственное общение, они предоставляют возможности спонтанного приобретения знаний и требуют меньших когнитивных затрат, чем традиционные методы общения и образования. Виртуальные среды способствуют развитию необходимой реалистичности и интерактивности и поэтому способны заменять непосредственное общение и образование, поддерживая систему изучения ситуаций.

3. Возможность приобретения знаний и умений способами, недоступными в реальном мире вследствие их дистанцированности, высокой стоимости, опасности или непрактичности. Например, становится возможным реконструировать античные здания и города, чтобы посмотреть, как они могли выглядеть на самом деле, и как жили люди в древние времена. Или можно тренировать артистов до их выхода на реальную сцену. Виртуальные и смешанные миры обеспечивают следующие три вида опыта построения знания, которые недоступны в реальном мире.

Размер – в подобных мирах пользователи могут изменять свои пространственные характеристики для более детального изучения исследуемого предмета. Например, они могут увеличиваться до планетарных размеров или уменьшаться до такой степени, чтобы можно было различить атомы и молекулы.

Трансдукция – преобразование данных в доступный для органов чувств формат, например, в формы, цвета, движения, звуки или вибрации, т. е. в то, что можно видеть, слышать, чувствовать и осязать.

Материализация – процесс создания вещественного представления абстрактных понятий.

Другим преимуществом является возможность анализа одного и того же предмета или явления с разных точек зрения. Таким образом, пользователи могут лучше усвоить изучаемый предмет и создать более правильное и полное представление о нем.

4. Возможность поддержки в виртуальных мирах с аватарами невербальной коммуникации, связанной с чувствами и эмоциями человека, его внешним видом и поведением, что дает ряд преимуществ.

Во-первых, они вносят социальный аспект с реакцией одобрения или переживания в целевое общение (обучение, рассказ, экскурсия и т. п.), которое часто воспринимается как отстраненное, безличное и, следовательно, не ведущее к возникновению мотивации.

Во-вторых, они могут показать, как выполняется то или иное задание в случае образовательных приложений, а не просто объяснить его, что сокращает время, необходимое для усвоения знаний, поскольку обучение через примеры более эффективно, чем обучение через объяснение.

В-третьих, они способны использовать средства невербальной коммуникации как для подтверждения объяснения, так и для обратной связи с пользователем. Например, они могут обратить его внимание на определенный объект непосредственно с помощью движения или взгляда, или способны положительно или отрицательно реагировать на его ответы только с помощью мимики. Такой вид коммуникации более предпочтителен, так как он не перебивает и не отвлекает отвечающего.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Навыки будущего: что нужно знать и уметь в новом сложном мире / Е. Лошкарева, П. Лукша, И. Ниненко, И. Смагин, Д. Судаков. АСИ, 2018. 92 с.
2. Навыки будущего для 2020-х: Новая Надежда. URL: <https://futureskills2020s.com/ru> (дата обращения: 30.09.2021).
3. Атлас новых профессий 3.0 / под ред. Д. Варламовой, Д. Судакова. М.: Интеллектуальная Литература, 2020. 456 с.
4. Агентство стратегических инициатив – Национальная Технологическая Инициатива. Сквозные технологии. Нейротехнологии, технологии виртуальной и дополненной реальности. URL: <http://www.nti2035.ru/technology/> (дата обращения: 25.10.2020).
5. Дорожная карта развития сквозной цифровой технологии «Технологии виртуальной и дополненной реальности». Министерство цифрового развития, связи и массовых коммуникаций Российской Федерации. URL: <https://digital.gov.ru/uploaded/files/07102019vrgar.pdf> (дата обращения: 25.10.2020).
6. *Купер А., Рейман Р., Кронин Д.* Основы проектирования взаимодействия. Пер. с англ. СПб.: Символ-Плюс, 2009. 688 с.
7. *Сергеев С. Ф [и др.]* Введение в проектирование интеллектуальных интерфейсов: учеб. пособие. СПб: СПбГУ ИТМО, 2011. 108 с.
8. *Анатольев А. Г.* Проектирование человеко-машинных интерфейсов. ОмГТУ. 2007.
9. ГОСТ Р ИСО 9241-210-2016. Эргономика взаимодействия человек – система Часть 210 Человеко-ориентированное проектирование интерактивных систем.
10. *Joseph J. LaViola Jr., Ernst Kruijff, Ryan P. McMahan, Doug A. Bowman, Ivan Poupyrev.* 3D User Interfaces: Theory and Practice / Second Edition. Boston, Addison Wesley, 2017.
11. *Силаева В. Л.* Об использовании понятия «виртуальный» // Социологические исследования. 2010. № 8. С. 19–35.
12. Виртуальный // Викисловарь. URL: <https://ru.wiktionary.org/wiki/виртуальный> (дата обращения 25.10.2020).
13. Новая философская энциклопедия: В 4 т. Т. 1. / ред. совет: В. С. Степин, А. А. Гусейнов, Г. Ю. Семигин, А. П. Огурцов [и др.]. М.: Мысль, 2010. 744с.
14. Ключевые понятия музеологии / сост.: А. Desvallées, F. Mairesse. Armand Colin, ИКОМ России, 2012. 104 с.

15. V-MusT.net – D2.1 Terminology, definitions and types of Virtual Museums. 52 p. URL: <http://www.v-must.net/library> (дата обращения: 25.10.2020).

16. ГОСТ 15971–90. Системы обработки информации. Термины и определения.

17. *Milgram P., Kishino F.* A Taxonomy of Mixed Reality Visual Display // IEICE Transactions on Information Systems special issue on Networked Reality. 1994. № 94. P. 1321–1329.

18. *Wickens C., Carswell C.* Information Processing. In G. Salvendy (ed.), Handbook of Human Factors and Ergonomics. New York: John Wiley & Sons., 1998. P. 130–149.

19. *Крысько В. Г.* Психология и педагогика в схемах и таблицах. Мн.: Харвест, 1999. 384 с.

20. *Черенкова Л. В., Краснощекова Е. И., Соколова Л. В.* Психофизиология в схемах и комментариях. СПб.: Питер, 2006. 240 с.

21. *Авербух Н. В.* Психологические аспекты феномена присутствия в виртуальной среде // Вопросы психологии. 2010. № 5. С. 105–113.

22. *Авербух Н. В.* Субъективный метод изучения особенностей переживания феномена присутствия // Национальный психологический журнал. 2018. № 3 (31). С. 69–89.

23. *Величковский Б. Б.* Психологические факторы возникновения чувства присутствия в виртуальных средах // Национальный психологический журнал. 2014. № 3 (15). С. 31–38.

24. Архитектура виртуальных миров: монография / под ред.: М. Б. Игнатьева, А. В. Никитина, А. Е. Войскунского. 2-е изд., перераб. и доп. СПб.: ГОУ ВПО «СПбГУАП», 2009. 287 с.

25. *Ковалев А. И., Меньшикова Г. Я.* Векция в виртуальных средах: психологические и психофизиологические механизмы формирования // Национальный психологический журнал. 2015. № 4 (20). С. 91–104.

26. *Joseph J. LaViola Jr.* 3D USER INTERFACES FOR GAMES AND VR. Spring, 2018.

27. *Bernhard E. Riecke, Joseph J. LaViola Jr., Charles N. Millican, Ernst Kruijff.* 3D user interfaces for virtual reality and games: 3D selection, manipulation, and spatial navigation // Conference Paper, Siggraph 2018 Course Notes. August 2018, Bali, Indonesia. January 7–9, 2014.

28. *Joseph J. LaViola Jr., Ernst Kruijff, Ryan P. McMahan, Doug A. Bowman, Ivan Poupyrev.* 3D User Interfaces: 31 Theory and Practice // Second Edition. Boston, Addison Wesley, 2017.

29. *Jaemoon Jung and etc.* A Review on Interaction Techniques in Virtual Environments. Proceedings of the 2014 International Conference on Industrial Engineering and Operations Management. Bali, Indonesia. January 7–9, 2014. P. 1582–1590.

30. *Jacek Jankowski, Martin Hachet.* A Survey of Interaction Techniques for Interactive 3D Environments // Eurographics 2013. STAR. May 2013. Girona, Spain.

31. *Дойч Д.* Структура реальности. Ижевск: НИЦ «Регулярная и хаотическая динамика», 2001. 400 с.

32. *Ланье Д.* На заре новой эры. М: Эксмо, 2019. 496 с.

33. *Steven M. LaValle.* Virtual Reality. Cambridge University Press, 2017. P. 417.

34. *John N. Latta, David J. Oberg.* A Conceptual Virtual Reality Model // Journal IEEE Computer Graphics and Applications. Vol. 14. Is 1, January 1994. P. 23–29.

35. *Philippe Fuchs.* Virtual Reality Headsets – A Theoretical and Pragmatic Approach. Ecole des Mines. ParisTech. Paris, France, 2017. 198 p.

36. Культурное наследие в реально-виртуальном континууме: учеб. пособие / под ред. канд. техн. наук А. В. Никитина. СПб.: ГУАП, 2017. 91 с.

37. *Azuma R. et al.* Recent Advances in Augmented Reality: survey // IEEE Computer Graphics and Applications. 2001. nov./dec. P. 34–47.

38. *Azuma R. T.* A Survey of Augmented Reality // Presence: Teleoperators and Virtual Environments. 1997. Vol. 6. № 4. P. 355–385.

39. *Milgram P. et al.* Augmented Reality: A class of displays on reality-virtuality continuum // Proceedings of SPIE. 1994. Vol. 2351. P. 282–292.

40. SO/IEC 18039:2019 (en) Information technology – Computer graphics, image processing and environmental data representation – Mixed and augmented reality (MAR) reference model.

41. *Henrysson A.* Bringing Augmented Reality to Mobile Phones: dissertations. 2007. URL: <http://liu.diva-portal.org/smash/record.jsf?pid=diva2:16967> (дата обращения: 25.10.2020).

42. Spatial Mapping. URL: <https://docs.microsoft.com/en-us/windows/mixed-reality/spatial-mapping> (дата обращения: 25.10.2020).

43. *Kiriakos N. Kutulakos, James R. Vallino.* Calibration-Free Augmented Reality // IEEE transactions on visualization and computer graphics. 1998. Vol. 4. № 1. P. 1–41.

44. *Bimber O., Raskar R.* Modern Approaches to Augmented Reality. ACM SIGGRAPH 2006. URL: <https://www.merl.com/publications/docs/TR2006-105.pdf>. (дата обращения: 25.10.2020).

45. *Christne Perey.* Mixed and Augmented Reality: Reference Model, Introduction and Overview. PEREY Research & Consulting, 2015. URL: <https://www.perey.com/ARStandards/mixed-and-augmented-reality-reference-model-overview-and-use-examples/>. (дата обращения: 15.10.2021).

46. *Gerard J. Kim.* New Work Item Proposal: A Standard Reference Model for Generic MAR Systems. ISO JTC 1 SC 24 WG9, Korea University. URL: <https://www.web3d.org/content/standard-reference-model-generic-mar-systems> (дата обращения: 15.10.2021).

47. *Butchart B.* Architectural styles for augmented reality in smartphones. EDINA. University of Edinburgh, 2011. URL: [http://www.perey.com/ARStandards/\[EDINA\]Mobile_AR_Architectural_Styles.pdf](http://www.perey.com/ARStandards/[EDINA]Mobile_AR_Architectural_Styles.pdf) (дата обращения: 25.10.2020).

48. *Zineb Rebbani, et al.* Definitions and Applications of Augmented / Virtual Reality: A Survey. International Journal of Emerging Trends in Engineering Research. Volume 9. No. 3, March 2021. URL: <http://www.warse.org/IJETER/static/pdf/file/ijeter21932021.pdf> (дата обращения: 15.10.2021).

49. *Никитин А. В., Решетникова Н. Н., Арнст А. В., Ситников И. А., Фильковский С. В.* Разработка интерактивных мультимедийных 3D приложений с использованием виртуальной и дополненной реальности: учеб.-метод. пособие. СПб.: ГУАП, 2020. 142 с.

50. What Does Virtual World Mean? URL: <https://www.techopedia.com/definition/25604/virtual-world> (дата обращения: 15.10.2021).

51. UPDATED: What is the Best Definition of Social VR? URL: <https://ryanschultz.com/2018/07/10/what-is-the-definition-of-social-vr/> (дата обращения: 15.10.2021).

52. A Complete List of Every Social VR Space and Virtual World Platform I Have Written About on The RyanSchultz.com Blog. URL: <https://ryanschultz.com/2018/05/10/a-complete-list-of-every-social-vr-space-and-virtual-world-platform-i-have-written-about-on-the-ryanschultz-com-blog/> (дата обращения: 15.10.2021).

53. Обзор сети Microsoft. URL: <https://docs.microsoft.com/ru-ru/mesh/overview> (дата обращения: 15.10.2021).

54. Framework for the Metaverse, Jun 29, 2021. Written By Matthew Ball. URL: <https://www.matthewball.vc/all/forwardtothemetaverseprimer> (дата обращения: 15.10.2021).

55. The Verge. URL: <https://www.theverge.com/22701104/metaverse-explained-fortnite-roblox-facebook-horizon> (дата обращения: 15.10.2021).

56. Multimodal Interaction Working Group. URL: <http://www.w3.org/2002/mmi/Overview.html> (дата обращения: 25.10.2020).

57. *Turk M.* Multimodal interaction: A review. *Pattern Recognition Lett.* (2013). URL: <http://dx.doi.org/10.1016/j.patrec.2013.07.003> (дата обращения: 25.10.2020).

58. *Bruno Dumas, Denis Lalanne, Sharon Oviatt.* Multimodal Interfaces: A Survey of Principles, Models and Frameworks // *Human Machine Interaction. Lecture Notes in Computer Science Volume 5440*, 2009. P. 3–26.

59. *Muhammad Z. Baig, Manolya Kavakli.* Multimodal Systems: Taxonomy, Methods, and Challenges. A PREPRINT, June 9, 2020. 18 p.

60. *Lalanne D., Khaled O., Bapst J.* Multimodal interfaces. 2006. DIVA research group, Department of Informatics of the University of Fribourg. URL: <http://diuf.unifr.ch/courses/05-06/mmi> (дата обращения: 25.10.2020).

61. *Martin J., Grimard S., Alexandri K.* On the annotation of the multimodal behavior and computation of cooperation between modalities // *Proceedings of the workshop on Representing, Annotating, and Evaluating Non-Verbal and Verbal Communicative Acts to Achieve Contextual Embodied Agents*. Montreal, 2001. P. 1–7.

62. Основы разработки интерактивных 3D-приложений на движке Unity: учеб.-метод. пособие / Д. А. Булгаков, Е. Е. Майн, Н. Н. Решетникова. СПб.: ГУАП, 2021. 137 с.

63. Анонс продаж VIVE Pro HMD – официальный блог VIVE. URL: <https://blog.vive.com/us/2018/03/19/htc-vive-announces-price-vive-pro-hmd-799-pre-orders-start-today-price-vive-reduced-499/> (дата обращения: 10.10.2021).

64. Анонс продаж и спецификации VIVE Pro Full Kit – новостной портал TechPowerUp. URL: https://www.techpowerup.com/245659/htc-announces-the-vive-pro-full-kit-steam-vr-2-0-base-stations-pro-controllers-included (дата обращения: 10.10.2021).

65. Поддержка VIVE Pro. URL: <https://www.vive.com/ru/support/vive-pro/> (дата обращения: 10.10.2021).

66. *Buckley S.* This Is How Valve’s Amazing Lighthouse Tracking Technology Works. URL: <https://gizmodo.com/this-is-how-valve-s->

amazing-lighthouse-tracking-technol-1705356768 (дата обращения: 10.10.2021).

67. Документация Steamworks по среде выполнения SteamVR. URL: <https://partner.steamgames.com/doc/features/steamvr/info?l=russian> (дата обращения: 10.10.2021).

68. Документация по OpenVR SDK. URL: <https://github.com/ValveSoftware/openvr> (дата обращения: 10.10.2021).

69. Документация XR Plug-in Framework из официального руководства Unity. URL: <https://docs.unity3d.com/Manual/XRPluginArchitecture.html> (дата обращения: 10.10.2021).

70. HTC Corporation. HTC VIVE Tracker (2018) Developer Guidelines. URL: [https://dl.vive.com/Tracker/Guideline/HTC_Vive_Tracker\(2018\)_Developer+Guidelines_v1.0.pdf](https://dl.vive.com/Tracker/Guideline/HTC_Vive_Tracker(2018)_Developer+Guidelines_v1.0.pdf) (дата обращения: 10.10.2021).

71. *Новикова Полина*. Обзор шлема Oculus Quest 2: виртуальная реальность без компьютера // Статья на сайте fotosklad.ru. URL: <https://www.fotosklad.ru/expert/reviews/obzor-slema-oculus-quest-2-virtualnaa-realnost-bez-komputera/> (дата обращения: 10.10.2021).

72. Требования к совместимости Oculus Link // Официальная документация Oculus. URL: https://support.oculus.com/articles/headsets-and-accessories/oculus-link/oculus-link-compatibility/?locale=ru_RU (дата обращения: 10.10.2021).

73. Описание платформы разработки Google ARCore. URL: <https://arvr.google.com/arcore/> (дата обращения: 10.10.2021).

74. Официальная документация по AR Foundation и дополнениям ARCore для Unity. URL: <https://developers.google.com/ar/develop/unity-arf?hl=en> (дата обращения: 10.10.2021).

75. Описание и загрузка Android Studio на сайте разработчиков под ОС Android. URL: <https://developer.android.com> (дата обращения: 10.10.2021).

76. Руководство по использованию Android. URL: <https://developer.android.com/ndk> (дата обращения: 10.10.2021).

77. Документация, новости и доступные для скачивания релизы Gradle. URL: <https://gradle.org> (дата обращения: 10.10.2021).

78. Конфигурирование ARCore в Unity. Портал разработчиков ARCore [Электронный ресурс] – режим доступа: <https://developers.google.com/ar/develop/unity-arf/enable-arcore#configure-required-optional>, свободный (дата обращения: 10.10.21).

Пример задания для демонстрационного экзамена

Разработать градостроительное приложение в дополненной реальности (AR). Приложение представляет собой симулятор градостроителя в дополненной реальности. Данные о том, каким образом должен быть застроен город, берутся из json-файла. Путь к этому файлу можно сменить при старте приложения в предлагаемой форме. В файле хранится информация о зданиях и жителях города.

Каждая запись о здании имеет следующие параметры:

– Тип (type): жилой (living), больница (hospital), ратуша (townhall).

– Цвет (color) – цвет здания в шестнадцатеричной системе счисления (FA3F19).

– Количество этажей (floorCount) – значение в виде числа.

Каждая запись о жителях имеет следующие параметры:

– Имя (name) – имя жителя, появляющееся над моделью человека.

– Пол (gender) – мужской (male), женский (female).

– Возраст (age) – значение в виде числа.

По данным, полученным из файла, в дополненной реальности генерируется город. Здания (количество, тип, цвет и этажность обусловлены информацией из json-файла) появляются в случайных частях города. После генерации моделей зданий пользователь может подвинуть и переместить их в рамках города с помощью перетаскивания. При нажатии кнопки «Завершить» к каждому из зданий строится дорога, объединяющая все здания. По дороге начинают прогуливаться жители, их количество, пол и имя, отображаемое над головой, определяются данными из json-файла.

Минимальное количество моделей:

– Четыре типа модульных зданий.

– Шесть типов людей в зависимости от возраста и пола.

– Дорога.

Оценка конкурсного задания

Итоговая оценка складывается из 100 (ста) баллов как сумма баллов за каждый из шести модулей, приведенных ниже.

Модуль 1

Наличие папки проекта на рабочем столе с корректным названием. Наличие внутри папки готовых файлов приложения в требу-

емых форматах для сдачи задания. Наличие в папке только используемых в приложении файлов с ресурсами – **5 баллов**.

Соответствие изображения метки (фрагментов метки) содержанию наполнению приложения. Наличие файлов с изображениями меток в папке проекта. Наличие распечатанных меток на рабочем месте участника. Стабильность распознавания меток (фрагментов меток) – **5 баллов**.

Модуль 2

Наличие в приложении информативной содержательной составляющей и практичность созданного приложения с точки зрения дальнейшего применения в реальной среде (корректность информации, общее соответствие контента тематике проекта) – **10 баллов**.

Использование дополнительных 3D объектов в количестве, превышающем минимальное в задании. Содержательное и качественное наполнение объектов – **5 баллов**.

Модуль 3

Удобство в использовании приложения: крупные кнопки, читаемые тексты, информативные пиктограммы – **2 балла**.

Дизайн приложения: внешняя эстетическая привлекательность приложения, использование тематических изображений, короткие текстовые блоки – **4 балла**.

Экран меню проекта. Наличие кнопок в меню согласно количеству экранов в задании. Соответствие дизайна меню содержанию приложения. Удобство и корректность навигации между экранами приложения (возврат в меню). Отсутствие режима дополненной реальности в меню – **10 баллов**.

Наличие экрана инструкции в приложении. Удобство использования и корректность предоставленной в инструкции информации. Отсутствие режима дополненной реальности в инструкции – **5 баллов**.

Модуль 4

Наличие экрана с дополненной реальностью в приложении. Использование минимального по заданию количества 3D объектов. Использование минимального количества блоков дополнительной информации. Соответствие блоков объектам. Удобство вызова блока дополнительной информации на экран. Наличие комбинаций ресурсов в блоке, корректность отображения информации на экране – **20 баллов**.

Функциональность использования объектов сценария: корректность соединений, отсутствие неиспользуемых событий и действий

в сценарных блоках, отсутствие дублирующих или лишних соединений. Логичность и «чистота» сценария: простота в прослеживании соединений между объектами. Удобство в расположении сценарных блоков – 5 баллов.

Наличие оригинальных нестандартных решений в проекте. – 5 баллов.

Модуль 5

Наличие в проекте приветственного видео. Автоматический запуск видео. Наличие возможности пропуска видео – 4 балла.

Наличие экрана с тестовым заданием. Наличие минимального количества вопросов и вариантов ответов. Корректность последовательности отображения вопросов и ответов на экране. Соответствие тестовых вопросов информационной части приложения: наличие ответов на вопросы в информационных блоках на экране с дополненной реальностью – 10 баллов.

Модуль 6

Наличие экспортного файла APK на демонстрационном мобильном устройстве. Корректное название приложения. Соответствие отображения интерфейса приложения диагонали демонстрационного мобильного устройства. Готовое приложение запускается и работает на демонстрационном устройстве без зависаний – 10 баллов.

СОДЕРЖАНИЕ

Список сокращений.....	3
ВВЕДЕНИЕ.....	4
1. МЕТОДЫ И ТЕХНОЛОГИИ ЦИФРОВЫХ РЕАЛЬНОСТЕЙ7	
1.1. Основы человеко-машинного взаимодействия.....	7
1.1.1. Основные термины человеко-машинного взаимодей-	
ствия	7
1.1.2. Термин «виртуальный»	7
1.1.3. Понятие «реально-виртуальный континуум»	8
1.1.4. поэтапная модель обработки информации....	9
1.2. Погружение, присутствие, киберболезнь	17
1.2.1. Погружение	17
1.2.2. Понятие вовлеченности	18
1.2.3. Присутствие.....	19
1.2.4. Киберболезнь	20
1.3. Интерактивность	23
1.3.1. Понятие интерактивности.....	23
1.3.2. Задачи 3D-манипуляций.....	24
1.3.3. Навигация	29
1.3.4. Управление системой 3D пользовательского интер-	
фейса	37
1.4. Виртуальная реальность	39
1.4.1. Подходы к определению виртуальной реальности	
39	
1.4.2. Подход к разработке VR	41
1.4.3. Присутствие, погружение, интерактивность	48
1.4.4. Основные классификации VR	49
1.5. Дополненная реальность и дополненная виртуальность	50
1.5.1. Определения дополненной реальности	50
1.5.2. Классификация устройств визуального отображе-	
ния.....	54
1.5.3. Критерии анализа ДР	56
1.5.4. Пример архитектуры системы ДР	57
1.5.5. Хранение информации о точках интереса ...	59
1.5.6. Дополненная виртуальность.....	60
1.6. Виртуальные и смешанные миры	61
1.6.1. Виртуальный или смешанный мир	61
1.6.2. Анализ многопользовательских ВСМ.....	61
1.6.3. Проекты Microsoft и Facebook.....	62

1.7. Мультиmodalный интерфейс	64
1.7.1. Мультиmodalное взаимодействие в виртуальной среде.....	64
1.7.2. Архитектура мультиmodalного интерфейса	65
1.7.3. Концептуальные модели мультиmodalного интерфейса	67
2. АППАРАТНО-ПРОГРАММНЫЕ СРЕДСТВА ВИРТУАЛЬНОЙ И ДОПОЛНЕННОЙ РЕАЛЬНОСТИ	70
2.1. Знакомство с комплектом VR-устройств VIVE Pro .	70
2.1.1. Комплект поставки и системные требования	71
2.1.2. Устройство и технические характеристики..	72
2.1.3. Устройство и способ подключение базовой станции	77
2.1.4. Подключение шлема к ПК.....	78
2.1.5. Работа с приложением SteamVR.....	80
2.1.6. Система ввода SteamVR для интерфейса OpenVR	83
2.1.7. Описание плагина XR Plug-in	86
2.1.8. Подготовка проекта Unity для шлема VIVE Pro	87
2.2. Дополненная виртуальность на базе трекеров VIVE	91
2.2.1. Комплект поставки и системные требования	92
2.2.2. Устройство и технические характеристики..	93
2.2.3. Включение и настройка VIVE Tracker 2.0	94
2.2.4. Работа с трекером в редакторе Unity	96
2.2.5. ДВ с использованием двойной камеры шлема	98
2.3. Знакомство с VR-шлемом Oculus Quest 2	99
2.3.1. Комплект поставки и системные требования	100
2.3.2. Устройство и технические характеристики	101
2.3.3. Включение и настройка шлема	104
2.3.4. Использование шлема в приложениях без ПК	110
2.3.5. Подключение шлема к ПК по проводу	114
2.3.6. Подключение шлема к ПК через Air Link ..	117
2.3.7. Запуск приложения SteamVR	119
2.3.8. Подключение шлема к ПК через Virtual Desktop	119
2.3.9. Работа с Oculus Quest 2 в редакторе Unity ..	122
2.4. Дополненная реальность на базе ARCore	124
2.4.1. Принципы работы ARCore	125
2.4.2. Принцип работы маркерной дополненной реальности.....	125
2.4.3. Требования к маркерам	126

2.4.4. Поддерживаемые ARCore устройства.....	127
2.4.5. Настройка смартфона для работы с ARCore	127
2.4.6. Установка Android Studio	130
2.4.7. Добавление поддержки сборки под Android в Unity	131
2.4.8. Подготовка проекта Unity для поддержки ARCore	132
2.4.9. Дополненная реальность на основе маркера	137
2.4.10. Пространственное картирование в ARCore	140
Заключение	144
Список использованных источников.....	146
Приложение. Пример задания для демонстрационного экзамена	152

Учебное издание

**Булгаков Дмитрий Алексеевич,
Никитин Александр Васильевич,
Решетникова Нина Николаевна,
Ситников Иван Александрович**

**РАЗРАБОТКА ВИРТУАЛЬНОЙ
И ДОПОЛНЕННОЙ РЕАЛЬНОСТИ**

Учебное пособие

*Под общей редакцией доктора технических наук,
профессора М. Б. Сергеева*

Редактор *Лазарева С.*
Компьютерная верстка *А. Н. Колешко*

Подписано к печати 00.00.21. Формат 60 × 84 1/16.
Усл. печ. л. . Уч.-изд. л. . Тираж 50 экз. Заказ №.

Редакционно-издательский центр ГУАП
190000, Санкт-Петербург, Б. Морская ул., 67