

## **ОГЛАВЛЕНИЕ**

<b>В. В. Кугуракова</b> <b>ПРЕДИСЛОВИЕ РЕДАКТОРА-СОСТАВИТЕЛЯ</b>	<b>403</b>
<b>Р. Р. Газизов, А. В. Шубин</b> <b>ПРОЦЕДУРНЫЕ МЕТОДЫ СКИННИНГА ГУМАНОИДНЫХ ПЕРСОНАЖЕЙ</b>	<b>404–440</b>
<b>В. В. Кугуракова, И. Д. Сергунин, Е. Ю. Зыков, А. В. Уланов, О. Д. Сергунин, Д. Р. Габдуллина, А. Ш. Гилемянов</b> <b>VR-ТЕЛЕУПРАВЛЕНИЕ «МНОГОРУКИМИ» УСТРОЙСТВАМИ: ПРОБЛЕМЫ, ГИПОТЕЗЫ, ПОСТАНОВКА ЗАДАЧИ</b>	<b>441–471</b>
<b>И. Р. Мухаметханов, М. Р. Хафизов, А. В. Шубин</b> <b>СРАВНЕНИЕ КЛИЕНТ-СЕРВЕРНЫХ РЕШЕНИЙ ПРИ РАЗРАБОТКЕ МНОГОПОЛЬЗОВАТЕЛЬСКИХ ОНЛАЙН-ИГР НА UNITY</b>	<b>472–488</b>
<b>Р. А. Шараева, В. В. Кугуракова, Р. Р. Галиева, С. В. Зинченко</b> <b>ПОДХОДЫ К ПРОЕКТИРОВАНИЮ ВИРТУАЛЬНЫХ ТРЕНАЖЁРОВ ХИРУРГИЧЕСКИХ ОПЕРАЦИЙ</b>	<b>489–532</b>
<b>А. В. Шубин, Г. Ф. Сахибгареева, В. В. Кугуракова</b> <b>ОБЗОР ПРАКТИК УПРАВЛЕНИЯ ПРОЕКТАМИ В ИГРОВОЙ РАЗРАБОТКЕ</b>	<b>533-552</b>

## **ПРЕДИСЛОВИЕ РЕДАКТОРА-СОСТАВИТЕЛЯ**

Настоящий тематический выпуск журнала «Электронные библиотеки» содержит новые научные результаты, полученные сотрудниками и студентами Учебно-практической лаборатории визуализации и разработки игр (Digital Media Lab – DML) Института информационных технологий и интеллектуальных систем Казанского (Приволжского) федерального университета в рамках Программы стратегического академического лидерства «ПРИОРИТЕТ-2030». В проведенных исследованиях разработаны способы ускорения процесса скиннинга при создании анимаций с использованием различных процедурных методов, представлен подход к разработке в виртуальной реальности тренажёров хирургических операций, предложено технологическое решение для обеспечения более качественного соединения между пользователями и сервером для многопользовательских видеоигр, развиты наиболее удачные методологии разработки видеоигр, а также предложен подход к созданию наиболее качественного и нативного управления удалённым роботизированным устройством.

Сфера применения названных результатов – виртуальная и дополненная реальности в образовательном процессе и разработке видеоигр; компьютерная графика и анимации; дизайн и разработка видеоигровых проектов, в том числе для большого числа одновременных пользователей; телеуправление роботизированным устройством посредством нативного манипулирования двумя и более конечностями при помощи систем виртуальной реальности. Проводимые исследования расширяются и находят все новые приложения.

**В.В. Кугуракова**, к. т. н., доцент кафедры программной инженерии Института ИТИС КФУ, руководитель НИЛ разработки интеллектуальных инструментов для компьютерных игр.

## ПРОЦЕДУРНЫЕ МЕТОДЫ СКИННИНГА ГУМАНОИДНЫХ ПЕРСОНАЖЕЙ

Р. Р. Газизов<sup>1</sup> [0000-0002-8349-264X], А. В. Шубин<sup>2</sup> [0000-0002-6203-3268]

<sup>1,2</sup> *Институт информационных технологий и интеллектуальных систем  
Казанского (Приволжского) федерального университета;*

<sup>1</sup>gazizov782@gmail.com, <sup>2</sup>shubin.aleksey.kpfu@gmail.com

### **Аннотация**

Процедура установки вершинных весов – очень трудоёмкая и сложная задача для любого художника трёхмерных моделей. Поэтому применение процедурных методов для облегчения этой процедуры является очень важным.

В статье проведён анализ различных методик скиннинга и выявлены их преимущества и недостатки. Описаны наиболее частые варианты дефектов скиннинга, возникающие при использовании стандартных подходов. Проведён анализ инструментария для скиннинга в среде трёхмерного моделирования Maya. Предложены методы решения некоторых из имеющихся проблем, но не подразумевающие процедурного решения. Также на основе нейронных сетей в качестве дополнительного инструмента для программы Maya предложена идея собственного решения. Этот инструмент позволит преодолеть большинство недостатков других методов и ускорить процесс скиннинга модели.

**Ключевые слова:** *трёхмерное моделирование, вершинные коэффициенты, оснастка, процедурные методы, нейронные сети.*

### **ВВЕДЕНИЕ**

Многие предметы нашего повседневного окружения демонстрируют упругое поведение при контакте с более жесткими объектами (например, кошка идет по подушке, рука прижата к окну, мягкий мяч прыгает на стойке ворот) – иначе говоря, происходят сплющивание внутри области контакта и выпучивание при перераспределении объема за ее пределами. Такие эффекты сжатия и выпучивания необходимы для передачи правдоподобных деформаций в различных ситуациях, таких как анимационные фильмы, визуальные эффекты или видеоигры. Они особенно важны в анимации персонажей, например, когда художники ставят

перед собой задачу передать действия персонажа на окружающую среду или на другого персонажа (хватание, толкание, нажатие и т. д.). Таким образом, деформации при контакте часто требуются на практике, однако существующие инструменты компьютерной графики остаются пока мало пригодными для художественного использования в таких ситуациях.

Трёхмерный объект со скелетом и костями по сути представляет собой отдельные компоненты геометрии, которые соединены друг с другом с помощью такого скелета. Скелет в свою очередь работает при помощи рига. Риг<sup>1</sup> – это набор зависимостей между управляющими и управляемыми элементами, которые созданы таким образом, чтобы управляющих элементов было меньше, чем управляемых. Основная задача, выполняемая при помощи риггинга, – упрощение процесса анимирования объекта.

Риггинг<sup>2</sup> – это подготовка персонажа к анимации. Важным недостатком риггинга является то, что он плохо работает для анимации деталей поверхности и может занять очень много времени для реализации сложных проектов.

Для подобных случаев есть различные инструменты, позволяющие проводить автоматический риггинг. В программе Maya доступен *Quick Rig tool*, который имеет возможность быстро создать риг для стандартного меша, а при помощи подробной инструкции [1] имеется возможность сделать риг точнее и лучше.

Проблема автоматического риггинга состоит в том, что не всегда скелет модели получается правильным, то есть данный функционал можно использовать лишь тогда, когда это позволяет модель. Большой проблемой риггинга может быть неправильный скиннинг модели. Скиннинг<sup>3</sup> – это процесс привязки точек трёхмерной модели к специально созданным для этого контактам-костям (см. рис. 1). Они оказывают влияние на вершины (или вертексы<sup>4</sup>) модели, двигая их в 3D-пространстве соответствующим образом.

---

<sup>1</sup> Риг (от англ. rig) – стенд.

<sup>2</sup> Риггинг (от англ. rigging) – оснастка.

<sup>3</sup> Скиннинг (от англ. skinning) – покрывать кожей.

<sup>4</sup> Вертекс (от англ. vertex – вершина) – точки из которых строится поверхность трёхмерной модели.

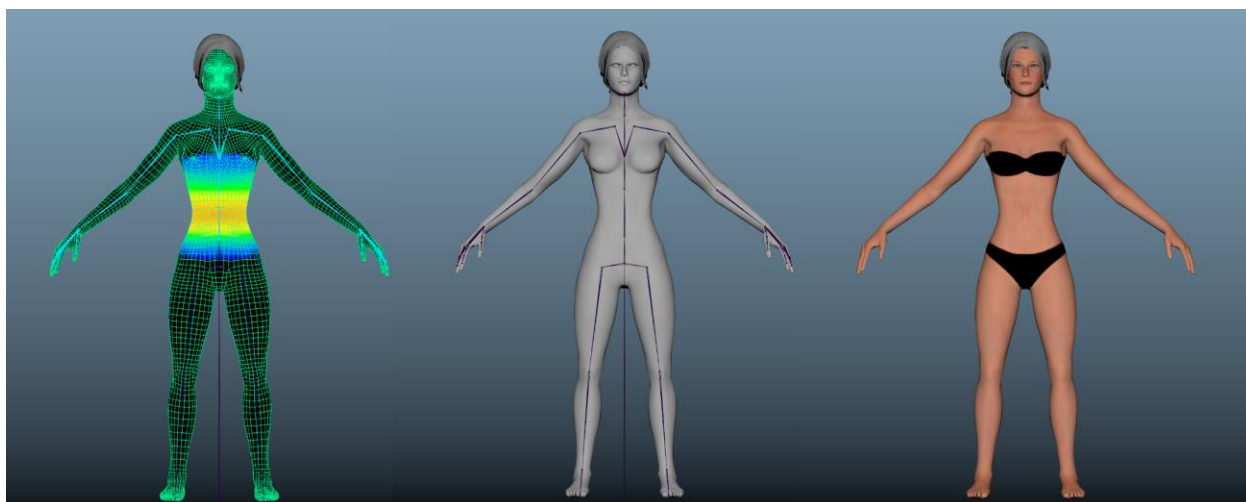


Рис. 1. Пример распределения весовых коэффициентов на трёхмерной модели (слева)

По сути при скиннинге для всех вертексов модели даются функциональные зависимости, как и какая кость влияет на модель: при вращении сустава вертексы, на которые действует кость, будут перемещаться по поверхности сферы с радиусом, равным расстоянию от вертекса до привязанной кости, на угол вращения, умноженный на коэффициент веса. Соответственно, если вес равен нулю, то кость на вертекс никак не влияет; если же вес равен единице, то производится полное вращение по радианам. Кроме того, не только вращение кости влияет на вертексы – перемещение работает схожим образом. Вертексы будут повторять перемещение кости, но их векторы будут умножаться на коэффициент влияния (вес).

Тщательный ручной скиннинг с использованием лишь стандартных инструментов часто бывает утомительным и затратным по времени. Среди стандартных методов установки весов можно назвать:

- ввод весов каждой из множества вершин и костей при помощи таблиц;
- использование ограничителей (envelope);
- раскрашивание веса точек модели «кистью».

Со временем скиннинг становился проще не только с расширением инструментария – появились алгоритмы, позволяющие автоматически распределить вес на нужные кости. Такой тип скиннинга называется *процедурным*. К сожалению, на данный момент, несмотря на большое число различных инструментов, многие из

них не предоставляют качественного автоматизированного решения и оставляют художнику задачу исправления результата до достижения надлежащего уровня.

Настоящая работа призвана выявить большинство проблем скиннинга персонажа, чтобы впоследствии предложить некоторый автоматизированный способ их решения.

### **ПРЕДЫДУЩИЕ РАБОТЫ**

Классический подход к решению проблем правильного задания деформации при контакте модели опирается на физические свойства упругих объектов, чтобы явно моделировать их поведение [2]. Очевидным преимуществом таких алгоритмов является их физическая точность при условии, что художникам удастся найти физические параметры, которые обеспечивают искомое поведение. На практике это требует обширной подготовки и навыков, а также частых проб и ошибок даже у эксперта по моделированию. Физическая точность может быть даже недостатком для постановок, которые стремятся к мультяшным, преувеличенным деформациям [3]. Кроме того, основным ограничением симуляций в интерактивном контексте является их зависимость от времени, что не позволяет использовать их на этапе монтажа или анимации.

Для анимации персонажей физически обоснованные подходы [4–7] обходят это ограничение, используя квазистатическое моделирование, не зависящее от времени. Хотя деформации удаленных контактов все еще зависят от пути, пройденного до состояния столкновения, они требуют существенных вычислительных затрат, поскольку большой набор нелинейных уравнений, описывающих моделируемый процесс, должен решаться итеративно, и поэтому такие подходы все еще не подходят для живого взаимодействия с пользователем. В качестве альтернативы, если использовать более простые модели на основе положения [8, 9], можно поддерживать локальные столкновения с интерактивной скоростью [10–12]. Несмотря на то, что эти методы могут помочь исправить артефакты скиннинга, такие как локальные самопересечения поверхности вблизи соединений, они не предназначены для представления удаленных контактов между частями тела или с внешними объектами.

Альтернативные решения, находящиеся на противоположной стороне методологического спектра, – это ручные подходы, полностью контролируемые художником, например, основанные на формах смещения, свободной форме [13, 14] или деформации пространства позы [15]. Их главное преимущество – простота: они обеспечивают мгновенную обратную связь с художниками, которые затем сами создают убедительные деформации. На практике эффекты выпуклости остаются редкостью в производстве, потому что задача создания деформаций и их анимации вручную требует значительного количества времени даже у опытных художников. Что еще хуже – каждая деформация специфична для формы объектов и их фактических контактов, поэтому ее нельзя повторно использовать в других ситуациях (например, от кадра к кадру).

Для анимации персонажей специальные техники скиннинга обеспечивают локальное или глобальное сохранение объема [16–18], и можно даже избежать самопересечений поверхностей [19], но это требует временной интеграции и предлагает ограниченный, косвенный художественный контроль через нарисованные веса. Неявный скиннинг [20, 21] также может обрабатывать локальные деформации поверхности (кожи) между соседними сочленениями, не прибегая к моделированию, благодаря неявному объемному представлению персонажа. Однако его операторы на основе градиента трудно поддаются художественной обработке, а удаленные контакты слишком дороги для интерактивной обработки.

В общем контексте трехмерных упругих объектов было предложено лишь несколько альтернативных подходов. Среди них нужно выделить работу [22], где деформатор обновляется автоматически на основе клетки, чтобы разрешить столкновение жесткого объекта с упругим, при этом примерно сохраняя объем, заключенный в клетке. Это обеспечивает некоторый контроль жесткости упругого объекта, а деформатор свободен от временных зависимостей, но достигает промежуточных характеристик только с помощью дорогого итерационного алгоритма на GPU. Кроме того, поскольку он требует промежуточного объемного представления, данный метод приводит к глобальным, грубым деформациям, а не к локализованному выпучиванию поверхности.

Другой метод [23] использует геометрический каркас, который решает вопросы пересечения поверхностей интерактивно во время геометрического моделирования, но включает в себя вычислительно сложную численную оптимизацию с ограничениями и не допускает художественного управления деформацией.

Параллельный метод [24] показал улучшенные результаты для конкретного случая деформации на основе «рукоятки» As-Rigid-As-Possible (ARAP) [25], но в остальном обладает аналогичными ограничениями.

Работа [26] предлагает подход к деформации, который позволяет художнику управлять вершинами и эффектами выпуклости.

Процедурные деформаторы, такие как плагин iCollide из Autodesk Maya или деформатор столкновений из Cinema4D, частично решают названные задачи, предлагая мгновенную деформацию с контролем профиля выпуклости. Как описано в работе [27], такие деформаторы состоят из двух основных этапов. Во-первых, они определяют точки упругого объекта, находящиеся внутри жесткого объекта, и проецируют их в ближайшее положение на жесткой поверхности, таким образом полностью сворачивая область пересечения. Во-вторых, упругая поверхность вне области пересечения деформируется по нормали пропорционально глубине взаимопроникновения. Поскольку объем в области пересечения не учитывается, названный инструмент может дать неправдоподобные результаты. Более того, при таком подходе вся пара пересекающихся поверхностей остается в контакте: в результате невозможно добиться эффектов, показанных на рис. 1, а во время анимации могут возникнуть нестабильности, что потребует от художников корректировки параметров во времени, что непрактично.

Применение физически обоснованных методов в процессе создания весовых коэффициентов позволяет значительно повысить правдоподобность движений персонажа. Моделирование на основе физических представлений позволяет вывести анимацию на основе скелета за рамки чисто кинематического подхода за счет имитации вторичных движений, таких как покачивание, сохранение объема и эффекты контактной деформации [28]. Эти вторичные движения обогащают визуальное восприятие анимации и необходимы для создания привлекательной анимации персонажей. Несмотря на такие реалистичные эффекты, физическое моделирование требует больших вычислительных затрат и является сложным,



поэтому в интерактивных приложениях его обычно избегают. Кроме того, физические методы требуют вмешательства человека для создания входных данных, которые описывают физические состояния, прежде чем соответствующая задача может быть решена компьютерной программой. В большинстве случаев такая подготовка является трудоемкой, и художник должен овладеть знаниями как данного программного обеспечения, так и физических представлений, лежащих в основе рассматриваемого явления.

### **ПРОБЛЕМЫ СКИННИНГА**

Для связи трехмерной геометрии с костями используют вершины. Понятие веса позволяет задать, как та или иная кость будет влиять на вершины геометрии модели. Чем сильнее влияние кости на вершину, тем сильнее вершина будет следовать за костью при ее трансформации. И, соответственно, чем слабее влияние кости на вершину, тем меньше она будет зависеть от трансформации кости. Такая система дает возможность создавать эффекты реалистичной деформации геометрии модели. Плавно назначенные веса создают красивые и, главное, реалистичные изгибы в районе суставов персонажа.

В результате правильного назначения веса каждая кость должна оказывать влияние только на тот участок геометрии, который ближе всего к ней находится. Также должен быть обеспечен естественный и плавный переход в участках суставов и естественных изгибов тела модели. Если вес был назначен неверно, то часто можно заметить резкие, «ломаные» движения конечностей. Пример: подняв ногу модели, мы можем увидеть, как вместе с ней потянутся и другие вершины геометрии, например, головы, что не приемлемо. Поэтому обсуждаемый процесс требует времени, внимания и усердия.

Скиннинг в Maya может быть автоматически произведен при помощи большого набора внутреннего инструментария. Подбор соответствующего метода и правильных параметров может обеспечить относительно лёгкий и быстрый скиннинг, но, к сожалению, зачастую алгоритмы стандартных методов являются довольно простыми, что приводит к появлению различных артефактов и проблем. Наиболее известные из них представлены ниже.

### **Перераспределение весов**

Одна из основных проблем автоматического скиннинга – перераспределение весов для получения правильной, реалистичной деформации. Вес кости задается в пределах от 0 до 1, где 0 – это отсутствие влияния, а 1 – максимальное влияние. Возможно также и частичное влияние, если установить дробные числа в указанном интервале. Важно, чтобы суммарный вес вершины был равен 1, то есть, если вершина имеет вес 0,7 для одной кости, то оставшиеся 0,3 веса должны быть распределены между оставшимися костями. Но зачастую использование автоматических методов приводит к неравномерному распределению веса либо может привести к новым артефактам.

Есть вариант, когда оставшийся вес распределяется на так называемую корневую кость<sup>5</sup>, что зачастую является крайним вариантом. Нередко для решения данной проблемы и получения качественной анимации художнику приходится потратить немало времени на ручную проверку и калибровку весов, что особенно заметно при работе с конечностями, имеющими большое количество последовательно соединённых костей (как, например, у хвоста или щупальца персонажа, имеющих большую подвижность).

### **Установка весов для новых моделей**

Несмотря на то, что некоторые модели могут иметь одинаковое количество конечностей, общую морфологию и расположение движущихся частей, художнику, приступая к работе с каждой новой моделью, приходится корректировать веса заново. Дело в том, что даже при всей своей функциональной схожести модели могут отличаться по пропорциям, форме, длине и ширине конечностей, а также другим параметрам, что означает наличие новой геометрии, следовательно, расположение и количество вертексов модели, что в итоге обязывает художника тратить дополнительное время на ручную калибровку.

---

<sup>5</sup> Корневая (от англ. root – корень) кость – старшая кость всего ригга в иерархии, перемещающая ее, мы будем перемещать персонаж по сцене.

---

### **Необходимость в дополнительной ручной доработке**

Несмотря на то, что существует большое количество разнообразных автоматизированных методов, чтобы ускорить или облегчить работу художника, их результат зачастую связан с большим количеством ошибок. Таким образом, практически любой современный инструмент без дальнейшей корректировки опытным художником не сможет дать полноценного, правдоподобного результата, что означает, что практически любой метод автоматизации и ускорения процесса скиннинга на самом деле не является полностью автоматическим.

### **Установка ненулевых весов на лишние точки**

Данный артефакт проявляется при использовании автоматических методов, основанных на удалённости вертекса от заданной кости. Дело в том, что в учёт не идёт то, что некоторые вертексы могут быть частью другой конечности, напрямую не связанной с рассматриваемой (см. рис. 2). В случае невнимательного применения подобных методов скиннинга без последующего своевременного обнуления весов лишних вертексов можно получить нереальные зависимости, такие как, например, участки головы, следующие за движением ноги или руки.

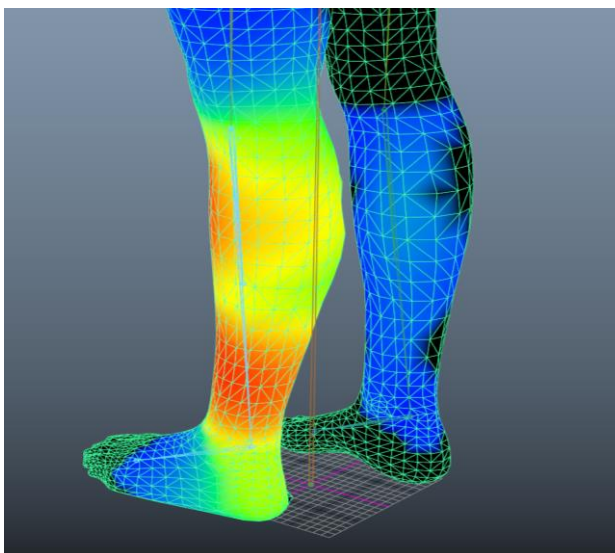


Рис. 2. Артефакт с распределением весовых коэффициентов

### Артефакт «конфетная обертка»

Этот вариант более специфичен и связан скорее не с перемещением, а с вращением суставов при использовании стандартных методов скиннинга, что приводит к эффекту «конфетной обертки». Такой эффект связан с тем, что при использовании автоматических методов вращаемая кость «тащит» за собой связанные с костью вертексы, образуя таким образом подобие конфетной обёртки. Использование подобных методов может быть хорошо применено для моделирования перемещения, однако при вращении некоторые вертексы могут начать сжиматься, что вызывает артефакт потери объема в модели, когда определенную часть «сдавило» (см. рис. 3).



Рис. 3. Эффект сжатия при вращении

### Артефакт с раздуванием суставов при сгибании

Подбор соответствующего метода автоматического скиннинга позволяет избавиться от некоторых проблем, упомянутых выше. Однако бывают случаи, когда метод, решая одну из проблем, может создать другую. Один из таких случаев – дефект раздувания суставов при сгибании (см. рис. 4), когда, например, область плечевого сустава человека влияет также и на подмышку, что вызывает эффект, когда при поднятии руки вертикально вверх она вместо растяжения «выдергивается» из тела персонажа, так как поворачивается вместе с рукой.

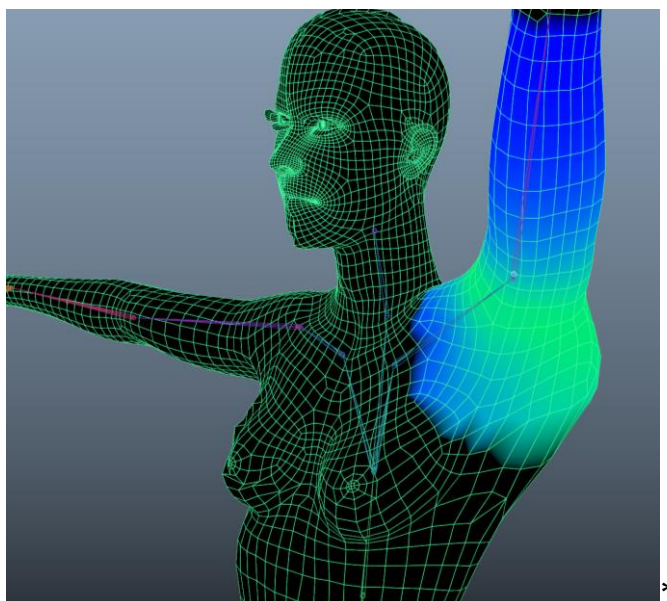


Рис. 4. Артефакт в плечевом суставе при автоматическом скиннинге (слева)

### **Невозможность автоматического создания вторичной и сложной анимаций**

Многие автоматические методы скиннинга являются чисто кинематическими и лишены вторичных эффектов движения, таких как покачивание жировых тканей и эффекты контактной деформации, то есть не могут передать реалистичную сложную деформацию. Упомянутые движения обогащают визуальное восприятие модели и необходимы для создания привлекательной анимации персонажей.

Основная проблема создания правдоподобных деформаций заключается в удовлетворении противоречивых требований интерактивности и правдоподобия в реальном времени. Правдоподобность требует достижения высокой детализации деформации, что означает передачу полного спектра желаемых эффектов, а именно: покачивания, сохранения объема, выпуклости мышц и деформации при контакте с кожей. Для создания таких деформаций требуется как минимум на порядок больше вычислительного времени, чем в существующих интерактивных системах деформации.

### **РУЧНЫЕ МЕТОДЫ СКИНИНГА**

Процесс скиннинга художники начинают с простого: выбирают одну часть модели, доводят её «до ума» и после этого переходят к следующей. Зачастую

предварительно создаются простые анимации для рига и перемещения всех используемых суставов, а затем внимательно изучается и проверяется влияние весов. Такая практика позволит сразу увидеть, где художник столкнулся с проблемами деформации и неверным влиянием суставов на модель, чтобы затем исправить их. Но, конечно же, такая практика не автоматизирована и может занять большое количество времени.

### **Стандартные методы установки весов**

Вес можно задавать различными способами. Его можно назначать каждой вершине и кости вручную, используя таблицу весов, но это зачастую бывает неудобно и долго. Хотя, иногда, только такой подход решает проблему точного назначения веса. Также весовые коэффициенты можно задавать, используя ограничители – виртуальные оболочки, позволяющие определить зону влияния кости. Такой подход является более понятным, но всё еще не всегда удобным. Для многих самым удобным способом назначения и распределения веса является его рисование. В этом случае мы вручную раскрашиваем вес точек модели при помощи кисти. Это дает возможность легко и быстро назначать вес, позволяет очень просто прорабатывать участки перехода веса (места, где вес распределяется между несколькими суставами).

Названные методы являются наиболее распространёнными в работе художников трёхмерных моделей, но скорость и качество работы зависят напрямую от опыта специалиста, так как эти методы не являются автоматизированными.

### **Отзеркаливание весов**

Тщательный риггинг часто бывает утомительным и время затратным, а потому отзеркаливание весов может сэкономить часы работы. Например, при расставленных весах на левой руке повторение всего процесса для правой руки не только станет тратой времени впустую, но и, вероятно, приведёт к несоответствиям, которые можно сразу не заметить. Отзеркаливание весов, то есть копирование их из одной симметричной конечности на другую, позволяет всего за несколько минут сделать в точности аналогичный скиннинг другой половины конечностей без необходимости проводить тот же самый объём работы вручную.

Названный метод отлично подходит для симметричных моделей, таких как реальные животные, но в иных случаях, когда моделируемое существо может иметь асимметричные морфологию или конечности, этот метод никаким образом не поможет ускорить работу.

### **Метод обертки**

При создании моделей с похожей геометрией есть возможность уменьшить объём работы при помощи копирования весов вершин с модели-обёртки. Для этого нужно создать низко полигональную<sup>6</sup> версию модели, затем вручную привязать эту версию к скелету, провести полноценный скиннинг и, наконец, скопировать веса на изначальную версию этой модели при помощи инструментов среды моделирования. После этого необходимо исправить возможные появившиеся ошибки, однако время, затраченного на скиннинг, будет уменьшено.

К сожалению, несмотря на очевидное преимущество применения данного метода, при использовании низко полигональной обёртки теряется часть информации, так как при использовании меньшего числа полигонов копирование весов на обёртку может быть менее сглаженным или вовсе некорректным из-за нехватки вертексов.

Данный метод может помочь и при скиннинге модели, которая разделена на множество объектов. При создании для модели обёртки, которая объединит все объекты вместе и будет использована для скиннинга, появится единый объект, на котором и будет происходить распределение весов.

Также при работе со скином фактически редактируется влияние каждой кости на вертексы модели, поэтому во избежание сложностей необходимо удостовериться, что все вертексы модели сшиты. В противном случае скиннинг таких участков модели образует дыры, которые зачастую можно увидеть только при раскраске весов, что означает, что процесс скиннинга придется переделывать заново.

К сожалению, применение метода обёртки является сложным и требует от художника большого числа манипуляций, что может привести к большому числу ошибок.

---

<sup>6</sup> Полигон (от англ. polygon – многоугольник) – область в пространстве, которая образует поверхность модели.

---

### **Добавление дополнительных костей**

Существует также ручной способ исправления артефактов со скручивани-ями путём добавления суставов вращения. Например, вместо того, чтобы скручи-вать один сустав на 180 градусов, можно разделить кость на четыре, и каждый получившийся сустав скрутить на 45 градусов. При таком решении полученные су-ставы будут также влиять на вращение, делая его намного более сглаженным, по-этому нужно аккуратно применять данный способ, чтобы исключить нежела-тельные эффекты.

Похожий приём может использоваться для борьбы с потерей объёма: в ме-стах сгиба суставов (плечи, локти и колени) можно добавить по дополнительному суставу спереди и сзади сустава сгиба. Тогда при сгибе конечности они будут «от-талкиваться» от основного сустава и частично предотвращать потерю объёма в складках.

Все подходы, описанные выше, рутинные и не предлагают качественного автоматизированного решения.

### **СТАНДАРТНЫЕ МЕТОДЫ АВТОМАТИЧЕСКОГО СКИННИНГА (MAYA)**

Скиннинг в Maya может быть реализован сразу несколькими подходами, например, через стандартный вариант автоматического скиннинга BindSkin или технологию nCloth. Последняя используется в основном для обработки физиче-ского описания одежды, но за счет тонких настроек ее можно использовать и для скиннинга модели персонажа с более реалистичными движениями кожи и мышц. К сожалению, данный вариант менее универсальный и плохо подойдет для ани-мации моделей для видеоигр, так как ориентирован на однотипную продолжаю-щуюся анимацию и в лучшем случае подойдет в качестве метода анимации мо-делей для CG<sup>7</sup>-видеороликов внутри редактора трёхмерной графики.

В свою очередь, BindSkin предлагает обычный набор атрибутов, предостав-ляя, тем не менее, довольно обширный функционал (см. рис. 5).

---

<sup>7</sup> Computer Graphics (англ.) – это техника создания изображений на компьютере.

---



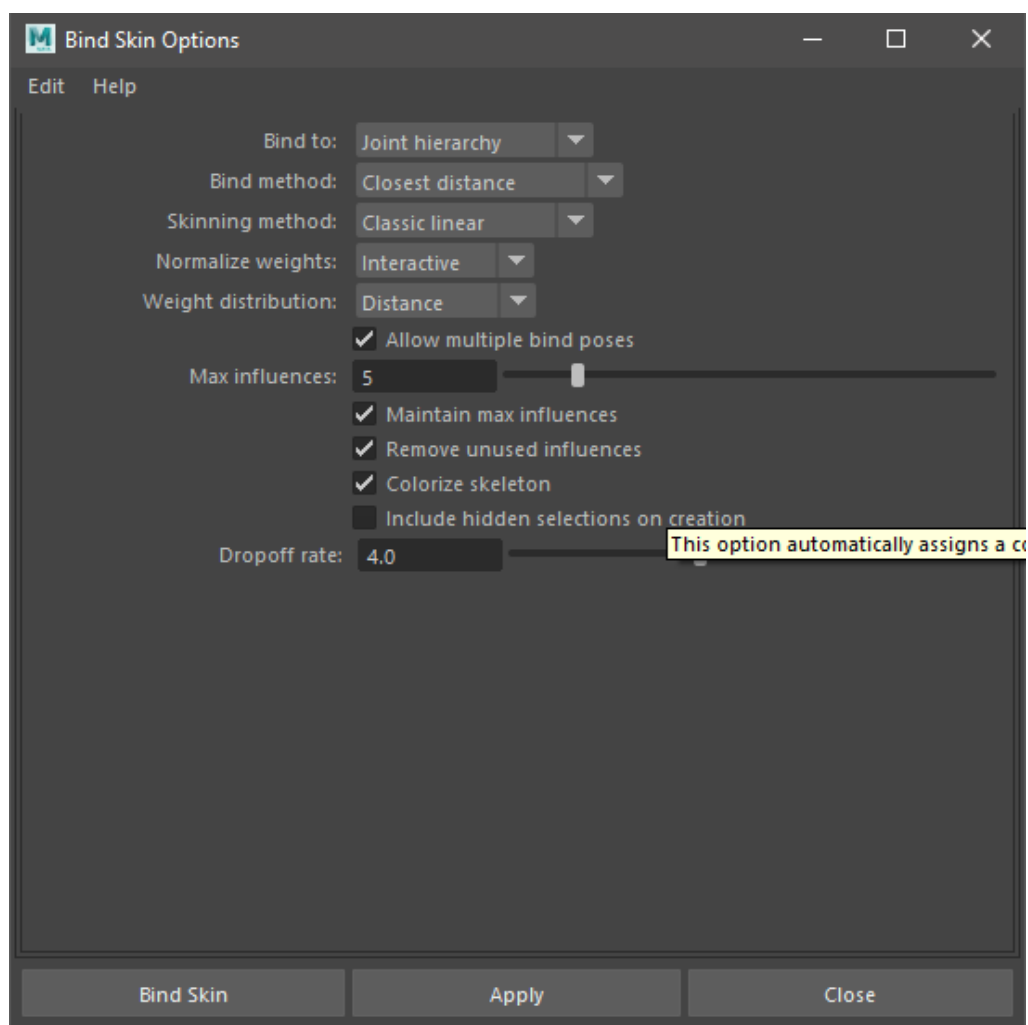


Рис. 5. Привязки меша к скелету в Maya

## Методы привязки

### 1. Методы на основе расстояния

Параметр называется *Bind method*. Он отвечает за метод, который регулирует выбор подходящих вертексов, после чего им назначается значение веса. Метод, указанный в этом параметре, влияет на алгоритм процедурного скиннинга напрямую и, по сути, составляет чуть ли не третью часть всей алгоритмики процедурного скиннинга.

Распределение веса может происходить несколькими способами:

- *Closest Distance* – относительно расстояния вертекса до каждого из суставов в привязке. Чем дальше сустав находится от вертекса, тем меньше он на него влияет.

- *Closest In Hierarchy* – относительно расстояния, но сначала ближе всего по иерархии, а потом уже по расстоянию.

Проблемой этих методов является то, что несмотря на математически справедливое распределение, чаще всего настолько простой алгоритм вызывает артефакты в риге персонажей, и их приходится исправлять вручную. Например, чаще всего в риге человека область плечевого сустава влияет еще и на подмышку, что вызывает эффект раздувания суставов, когда при поднятии руки вертикально вверх подмышечная область вместо растяжения «выдергивается» из тела персонажа, поворачиваясь вместе с рукой.

## 2. Карта теплоты

Heat Map – метод, который является более продвинутым, так как исходит из следующей концепции.

Существует такое подпространство фигуры, которое можно описать как внутреннюю полость модели, определяемую тем, что если выпустить любой луч из точки этого подпространства, то при попадании на поверхность модели он попадет на обратную сторону грани. Но вместо начальной точки в алгоритме используются непосредственно суставы: если сустав находится снаружи модели, он никак на нее не влияет. Если же он внутри модели, то влияет только на те вертексы, для которых отрезок между вертексом и суставом не выходит из модели (не пересекает ни одну из граней, кроме той, на которую попал). Это более универсальный алгоритм, так как он позволяет, например, избавиться от артефакта при анимации персонажа, когда участок, принадлежащий одной ноге, перемещается за второй.

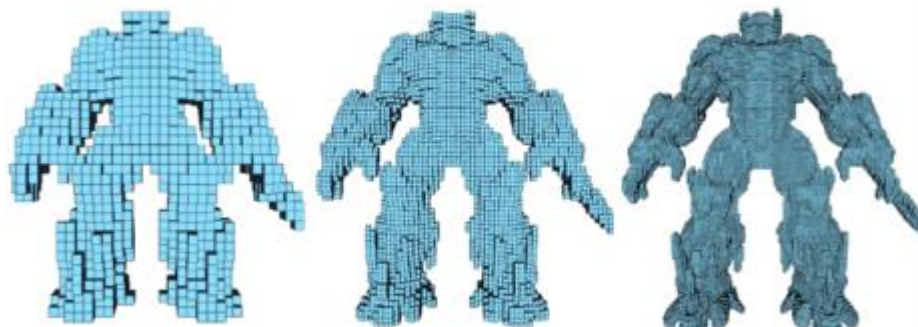
## 3. Геодезическая привязка вокселями

Geodesic Voxel Binding – геодезическая привязка при помощи вокселей. Этот метод более грубый в сравнении с остальными, но позволяет работать с не совсем правильными моделями, которые содержат нулевые грани, ребра и т. д. Данный метод делает из модели ее воксельную копию<sup>8</sup>, после чего на основе расстояния до вершин вокселя строит свою сетку, в которой каждому вертексу присваивается значение на основе расстояния от вертекса до вершин того вокселя, в котором

---

<sup>8</sup> Воксельная копия модели – вариант модели, состоящей из одних кубов-вокселей и примерно повторяющей изначальную форму.

вертекс находится. Также добавляется новый атрибут Resolution<sup>9</sup>, который отвечает за воксельную детализацию. Чем выше Resolution, тем меньше воксели и тем больше очертания воксельной копии схожи с реальной моделью (см. рис. 6). При уменьшении разрешения топология персонажа может изменяться, что изменяет минимальное расстояние между вокселями, падающими на кости, и не внешними вокселями.



Рису. 6. Воксельные представления трехмерных моделей с разной детализацией [29].

Но и последние два метода не решают всех проблем со скиннингом, так как, например, тот же артефакт с раздуванием суставов все равно будет присутствовать как при использовании карты теплоты, так и при геодезической привязке вокселями. Поэтому для каждого случая нужно выбирать свой подходящий метод скиннинга, конечный результат которого пришлось бы меньше исправлять.

### **Методы сглаживания**

При необходимости инструмент Bind Skin дополнительно поддерживает применение сглаживания при автоматической установке весов.

*Простой скиннинг:* каждая вершина меша привязывается к одной конкретной кости скелета (см. рис. 7, б). При таком подходе скин персонажа моделируется как единая непрерывная сетка. Каждая вершина в сетке привязана ровно к одному суставу в скелете, и когда скелет позиционируется, вершины трансформируются матрицей пространства их суставов. Техника простого скиннинга подходит для моделей с низкой детализацией, но явно недостаточна для персонажей

---

<sup>9</sup> Resolution (англ.) – параметр, отвечающий за количество и плотность кубов в воксельной копии.

---

более высокого качества. На практике алгоритм простого скиннинга может работать для персонажей с 500 или даже 1000 полигонами, если соблюдать осторожность при размещении вершин и прикреплении костей. Простого скиннинга может быть достаточно для персонажей с низкой детализацией, но для более высокого качества он слишком ограничен, поэтому требуется лучшее решение.

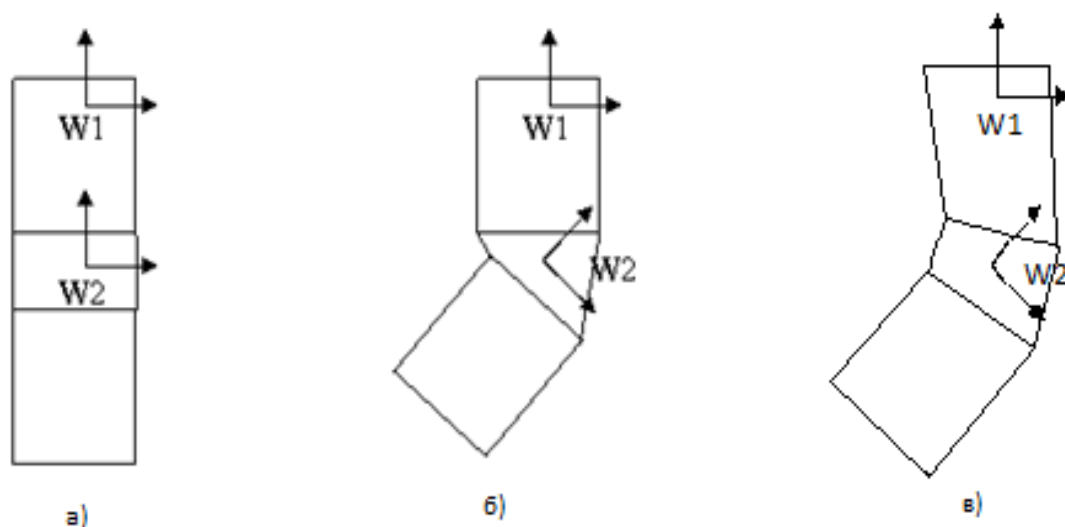


Рис. 7. Различные варианты привязки (а – разогнутое колено, б – согнутое колено с простым скиннингом, в – согнутое колено со сглаженным скиннингом)

Применение методов сглаживания расширяет концепции, используемые в простом скиннинге. При «гладком» скиннинге каждая вершина сетки может быть присоединена к нескольким суставам, каждый из которых влияет на вершину с разными силой или весом (см. рис. 7 в). Окончательное преобразованное положение вершины представляет собой средневзвешенное значение начального положения, преобразованного каждым из присоединенных суставов. Например, вершины в колене персонажа могут быть частично привязаны к тазобедренному суставу (управляющему верхней частью бедра) и коленному суставу (управляющему икрой). Многие вершины должны быть привязаны только к одному или двум суставам, и редко бывает необходимо привязать вершину более чем к четырем [30].

*Скиннинг на основе линейного смешивания (linear blend skinning):* на каждую вершину могут оказывать влияние несколько костей, у каждой кости есть значение веса для каждой связанной с ней вершиной. Этот способ помогает создавать более реалистичные анимации и является классическим видом скиннинга.

С методом линейного смешивания связано несколько артефактов анимации. Поскольку он не заботится о сохранении объёма, при слишком сильном сгибании суставов геометрия модели может проходить сквозь саму себя, а на местах, близких к суставу, может появиться выпуклость. Аналогично, при скручивании сустава может появиться артефакт «конфетная обертка», когда модель в месте скручивания будет более суженной, чем вся другая соседняя поверхность.

*Скиннинг на основе двойного кватерниона:* преобразует матрицы вращения в кватернионы и нормализует их. Этот метод не требует изменений в модели и помогает исправить артефакты с потерей объёма при сгибании и скручивании конечностей. К сожалению, метод является более нагруженным, так как на каждый вертекс приходится больше инструкций, обрабатывающих его, из-за применения сразу нескольких методов: сначала вычисляется линейное результирующее, после чего находится угловое результирующее (на какой угол отклонился объект), после этого финальный вектор для перемещения объекта строится по линейному результирующему вектору, который нормируется до радиуса окружности вращения, и углу.

Таким образом, ни один из стандартных методов автоматического скиннинга не даёт качественного результата и обязывает художника в ручном порядке исправлять полученный результат. Безусловно, применение стандартных методов ускоряет работу художника, так как ему не нужно изначально расставлять веса вручную, но при этом такой инструмент не может в полной мере автоматизировать данный процесс.

## **ПРОЦЕДУРНЫЕ МЕТОДЫ СКИННИНГА**

### **Шаблоны скиннинга**

Один из предлагаемых подходов подразумевает использование абстракций для скиннинга, что позволяет повторно использовать поведение скиннинга для схожих суставов и на схожих персонажах. Такая абстракция воплощена в виде шаблонов скиннинга, которые определяют поведение вертексов при деформации для общих типов суставов.

Шаблоны скиннинга реализуют деформации на основе клетки [31], что обеспечивает гибкое пространство проектирования, в котором можно многократно использовать заготовленные модели поведения скиннинга.

Под клеткой понимается управляющая сетка с фиксированной топологией, которая подгоняется под геометрию персонажа. Хотя методы, основанные на клетках, позволяют плавно деформировать геометрию кожи, позиционирование клетки требует ручного манипулирования вершинами клетки. Скелет управляет движением вершин клетки с помощью техники скиннинга, основанной на примерах, благодаря чему клетка плавно деформирует модель персонажа. Клетка имеет простую структуру, которая слабо отделена от модели персонажа, что облегчает задачу разработки деформационного поведения, которое можно использовать совместно. Деформации на основе клетки могут быть эффективно вычислены, поддерживая тем самым интерактивное переключение шаблонов скиннинга. Полуавтоматический этап подгонки используется для обеспечения того, чтобы деформационное поведение клетки оставалось одинаковым при применении к различным геометриям. Например, шаблон для скиннинга мышечного бугра (см. рис. 8), управляемого локтем, позволяет добиться качественно одинаковых деформаций мышечного бугра как для худых рук, так и для полных.

Шаблон скиннинга позволяет достичь определенного эффекта скиннинга, например, выпуклости мышц или сгибания локтя без защемления. Также данный метод позволяет быстро изучить различные модели поведения скиннинга, чего не хватает в традиционных моделях скиннинга. Шаблоны скиннинга могут использоваться пользователями совместно, поскольку они не представлены в виде конкретной модели.

В случае использования такого метода совместно с технологиями нейронной сети художник сможет применять данное решение для автоматического создания рига персонажей со сложной сеткой, таким образом освобождая себя от трудоёмкой ручной работы, благодаря чему остаётся лишь проконтролировать и при наличии ошибок исправить конечный результат.

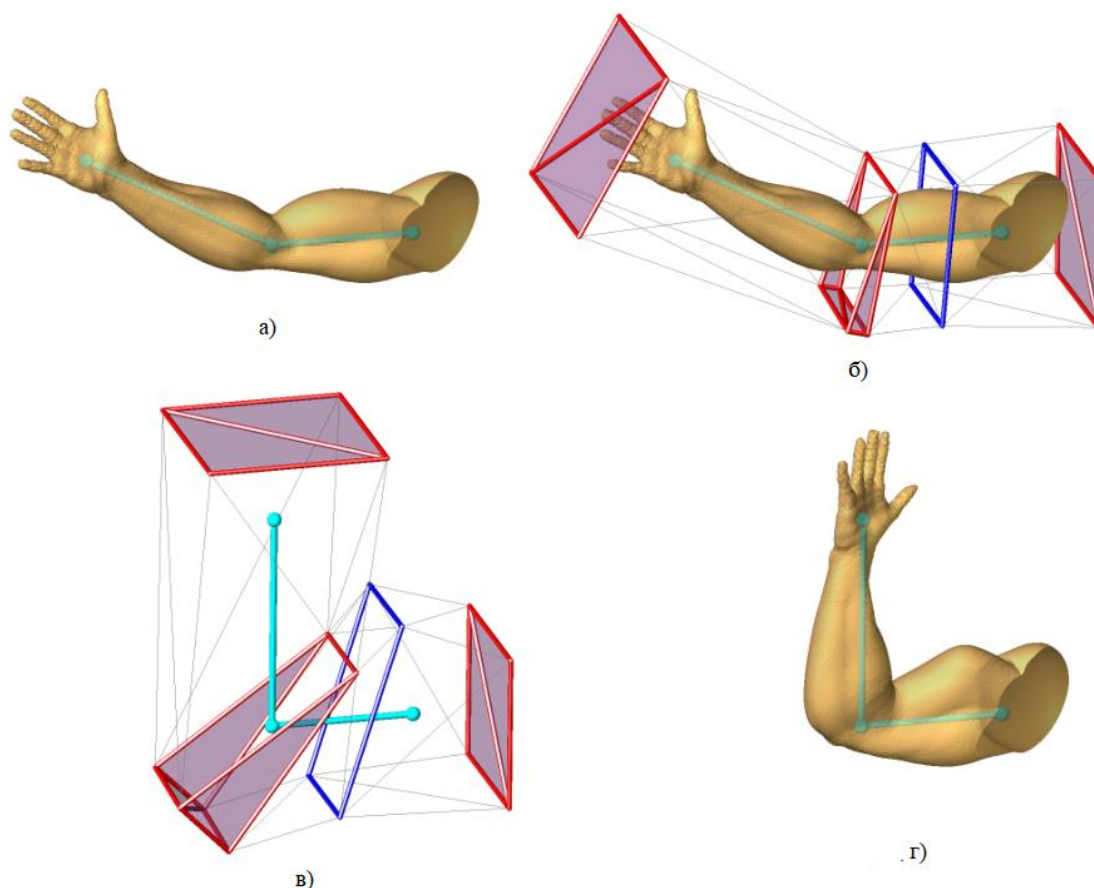


Рис. 8. Скиннинг на основе клетки: (а) первоначальная геометрия модели; (б) установка изначальной клетки; (в) деформирование клетки; (г) полученный результат с учётом реалистичной деформации мышц.

К сожалению, данная технология не позволяет автоматически создавать вторичную анимацию при движении персонажа, что является значимым минусом и обязывает художника провести работу над созданием более реалистичных движений.

### **Вторичный скиннинг**

Вторичные анимационные эффекты – важная часть при создании достоверной анимации, которая обеспечивает реалистичность движения модели. Данные эффекты также добавляются на этапе скиннинга.

Один из предлагаемых вариантов подразумевает добавление вторичных эффектов поверх стандартного скиннинга, что позволит стилизовать движение модели под руководством художника [32].

Такой метод использует в качестве исходных данных готовую сетку с геометрией, скелетом, весами и анимацией скелета. Далее генерируется деформация сетки, которая подчеркивает и стилизует уже реализованное движение, вызванное движением скелета, при помощи расчёта линейных и угловых скоростей вдоль скелетной иерархии. Используя преимущества иерархической природы скелета, можно аппроксимировать динамические деформации, которые в противном случае были бы недоступны без ручного ввода опытным аниматором или применения дорогостоящего физического моделирования.

Основная идея заключается в том, что иерархия скелета, встроенная в расчет скиннинга, предоставляет достаточно информации для разложения движения сетки на значимые, легко управляемые отрезки перемещения, которые могут быть автоматически связаны с деформациями. Этот подход противопоставляется существующим инструментам моделирования, таким как деформаторы в стандартном пакете анимационного программного обеспечения, которые требуют ручной настройки и создания ключевых кадров квалифицированным художником для каждой анимации, что отнимает много времени и является громоздким.

Несмотря на то, что общая структура скиннинга скорости поддерживает широкий набор деформаций, мы продемонстрируем полезность двух специфических характерных деформаций, которые называем "squashiness" (сминаемость) и "floppiness" (болтаемость). Комбинирование этих эффектов позволяет имитировать, улучшать и стилизовать поведение, выглядящее физически реалистичным, в рамках стандартного анимационного конвейера для персонажей с произвольной кожей.

Величина этих двух эффектов деформации может быть тонко настроена с помощью скалярных весовых параметров для отражения различных свойств физических материалов и может быть определена как атрибуты на вершины, «нарисованные» на сетке художником. Вместе с небольшим набором интерфейсных контроллеров это позволяет просто управлять интуитивно понятным пространством возможного поведения деформации, например, для создания эффектов, которые кажутся сделанными из разнородных материалов.

Несмотря на то, что данный метод позволяет во многом уменьшить работу художника по передаче модели более реалистичных движений и сократить



время разработки анимации на многие часы, этот метод всё ещё требует первоначальных манипуляций по созданию изначального движения, скелета и весов. Обсуждаемый метод хорошо применим в качестве одного из этапов комплексного решения для создания анимации, когда изначальные веса устанавливались бы в качестве более раннего этапа при помощи другого автоматизированного метода.

### **Автоматическая система RigNet**

Предсказание формы скелета и кожи анимационного персонажа на основе произвольной статической 3D-сетки – амбициозная задача. Подход к автоматическому риггингу в идеале должен учитывать интуицию аниматоров относительно движущихся частей и деформации. Подход, основанный на обучении, хорошо подходит для этой задачи, особенно если он способен обучаться на большом и разнообразном наборе риггидных моделей.

Такой метод использует нейронную сеть для решения задачи и позволяет автоматически на основе готовой модели прогнозировать скелет модели, а затем провести этап установки весов. Таким образом, данный метод полностью решает задачу риггинга, в том числе реализует этап скиннинга.

Метод предсказывает как форму скелета, так и скиннинг, которые соответствуют ожиданиям аниматора. В отличие от предыдущих работ, в которых на входные 3D-сетки накладывались predetermined шаблоны скелета с фиксированным количеством суставов и топологией, наш метод формирует скелеты, более приспособленные к основной структуре артикуляции входных данных. В отличие от подходов к оценке позы, разработанных для определенных классов фигур, таких как люди или руки, данный подход не ограничен категоризацией фигур или фиксированной структурой скелета. Нейронная сеть представляет собой общую модель предсказания форм скелета и кожи, способную проводить риггинг различных персонажей.

Хотя аниматоры в основном согласны с топологией скелета и расположением суставов для вводимого персонажа, существует также некоторая неоднозначность как в отношении количества, так и точного расположения суставов. Это зависит от того, насколько реалистично точно следует сделать деформацию. Например, в зависимости от замысла анимации, рука может быть представлена с помощью одного лучезапястного сустава или с более тонким разрешением с

иерархией кистевых суставов. Сочленения позвоночника и хвоста могут быть представлены с помощью различного количества суставов. Таким образом, еще одной задачей метода риггинга является обеспечение простого и прямого контроля уровня детализации выходного скелета.

Архитектура подхода RigNet представлена в виде трёх модулей [33], которые последовательно выполняют процедуру риггинга:

Первый модуль – это графовая нейронная сеть, обученная предсказывать соответствующее количество суставов и их расположение при помощи смещения геометрии модели в направлении мест предполагаемого расположения суставов, что позволяет передать реалистичность движений персонажа. Поскольку окончательный скелет может зависеть от предпочтений художников и их задач, наш метод также позволяет пользователю вводить дополнительный параметр для управления уровнем детализации выходного скелета. Например, некоторые приложения, например, моделирование толпы, могут не требовать риггинга мелких деталей (например, руки или пальцев), в то время как в других приложениях, таких как игры от первого лица, риггинг подобных деталей наиболее востребован. Управляя этим параметром, можно вводить меньшее или большее количество суставов для создания разного уровня детализации выходного скелета.

Второй модуль учится предсказывать, какие пары суставов должны быть соединены между собой. Для создания более верной структуры учитывается не только расположение суставов между собой, но и общая геометрия модели путем создания иерархической структуры (анимационные скелеты избегают циклов в качестве выбора дизайна), соединяющей суставы. При построении структуры костей используется алгоритм минимально связывающего дерева, который определяет приоритет наиболее вероятных костей. Приоритезация начинается с корневой кости, которую определяет другая часть модуля.

После создания скелета, заключительным этапом предлагаемой архитектуры является предсказание веса скиннинга для каждой вершины сетки для завершения процесса риггинга. Чтобы выполнить скиннинг, сначала извлекается представление сетки, отражающее пространственные отношения вершин сетки по отношению к скелету. Это представление вдохновлено предыдущими методами скиннинга [34], которые вычисляют влияние костей на вершины в соответствии с объемным геодезическим расстоянием между ними. Это представление

сетки обрабатывается через графовую нейронную сеть, которая выдает веса скиннинга для каждой вершины.

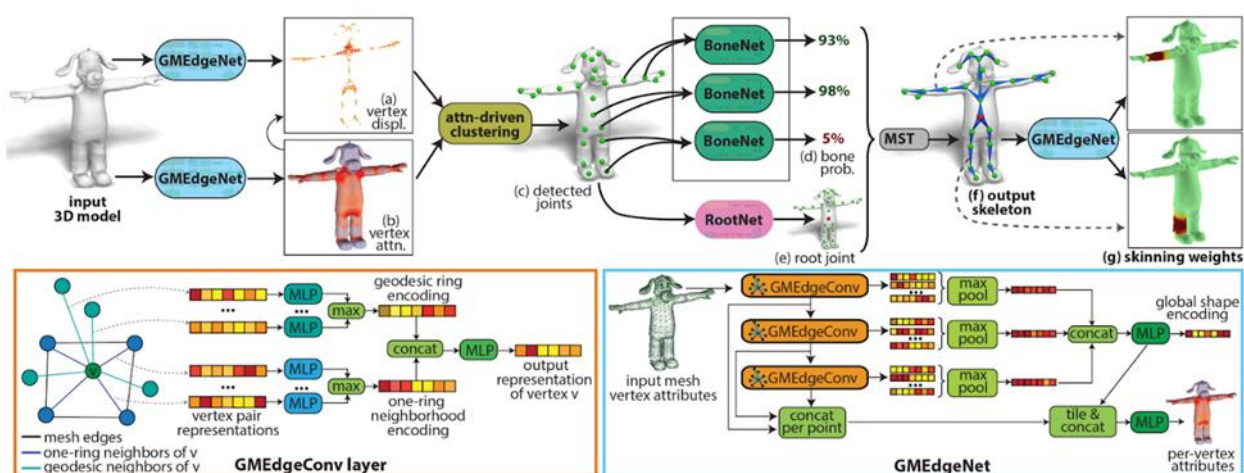


Рис. 9. Конвейер метода RigNet

Таким образом, полный цикл обработки модели при помощи метода RigNet выглядит так, как показано на рис. 9:

A. Учитывая входную 3D-модель, графовая нейронная сеть GMEdgeNet предсказывает смещения вершин по отношению к соседним суставам.

B. Другой модуль GMEdgeNet с отдельными параметрами предсказывает функцию внимания над сеткой, которая указывает на области, более значимые для предсказания расположения суставов (более красные значения указывают на более сильное внимание – смещенные вершины также окрашены в соответствии с этим вниманием).

C. Под влиянием внимания к сетке модуль кластеризации обнаруживает суставы, показанные зелеными шариками.

D. Учитывая обнаруженные суставы, нейронный модуль BoneNet предсказывает вероятность того, что каждая пара суставов будет соединена.

E. Другой модуль RootNet выделяет корневой сустав.

F. Алгоритм минимального связующего дерева MST (Minimum Spanning Tree) использует результаты BoneNet и RootNet для формирования анимационного скелета.

G. Наконец, модуль GMEdgeNet выводит веса скиннинга на основе предсказанного скелета.

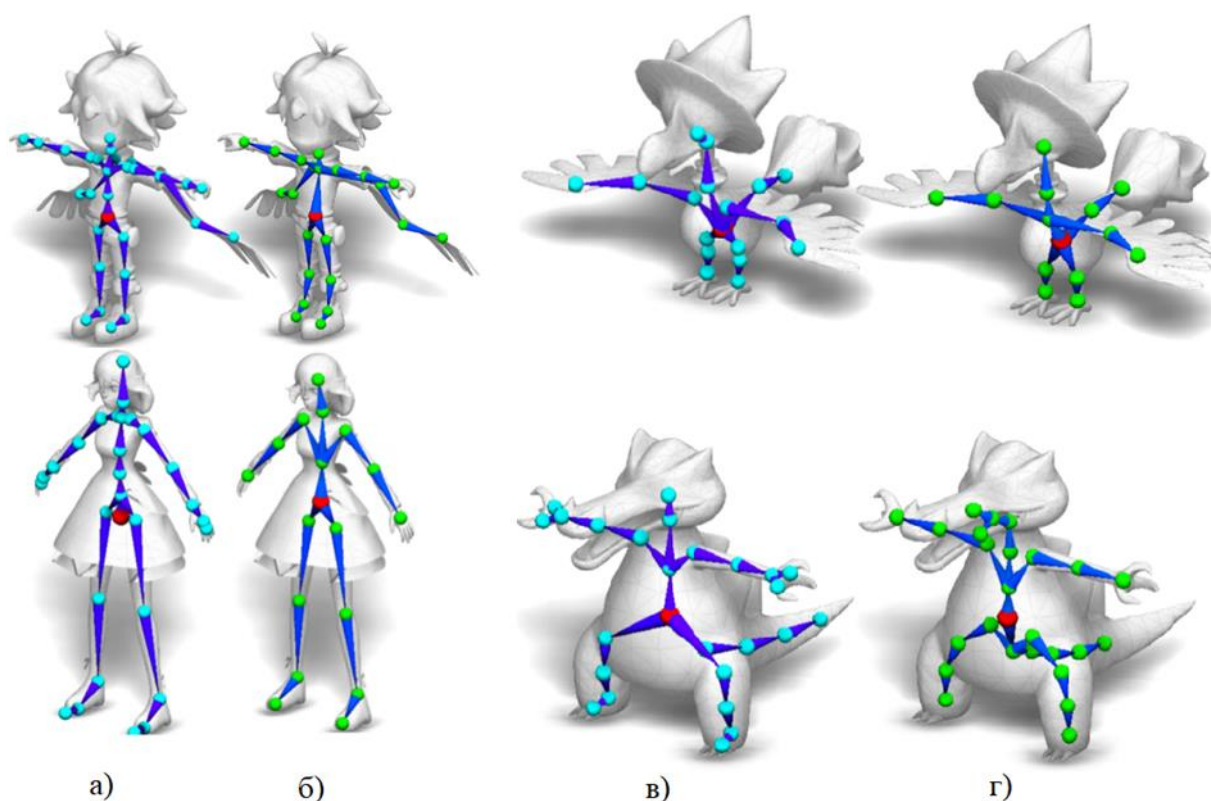


Рис. 10. Примеры результатов работы RigNet

Представленная модификация метода сквозного автоматизированного создания анимационных ригов позволяет автоматически развешивать кости трехмерных моделей (см. рис. 10). Несмотря на это, RigNet имеет некоторые ограничения:

- RigNet использует поэтапный подход к обучению. В идеале потери при скиннинге могли бы передаваться обратно на все этапы сети для улучшения совместного прогнозирования. Однако это подразумевает дифференциацию объемных геодезических расстояний и оценку скелетной структуры, что является сложной задачей.
- Набор данных имеет ограничения. Он содержит один риг на модель. Многие риги часто не включают кости для мелких деталей, таких как ступни, пальцы, одежда и аксессуары, что делает обученную модель менее предсказуемой для этих суставов.

- Обогащение набора данных большим количеством моделей может повысить производительность, хотя это может сделать отображение более многовариантным, чем сейчас.

### **Предлагаемый подход**

Применение искусственного интеллекта и нейронных сетей имеет большие перспективы в рабочих процессах компьютерной графики и позволяет сделать утомительные задачи делом прошлого.

Предлагаемое решение представляет собой плагин для программы Autodesk Maya. Благодаря наличию Maya Python API с поддержкой разработки на языке Python, имеется возможность подключения и использования современных библиотек для обучения нейронных сетей. Наиболее популярными такими библиотеками на данный момент являются TensorFlow, Theano, Torch, CNTK, Keras.

Предполагаемое решение подразумевает комплексный функционал по полноценному риггингу гуманоидного персонажа:

- Создание масштабируемого скелета гуманоидного персонажа на основе готовой трёхмерной модели или облака точек в нескольких вариантах.
- Процедурный скиннинг с привязкой полученных костей к вершинам моделей на основе обучения нейронной сети качественными примерами.
- Запись и загрузка анимаций из облака точек, технологии захвата движения или видео в редактор Maya для их дальнейшего использования.

Предполагается, что данная система будет использовать Maya Python API, Keras в качестве фронтэнда<sup>10</sup>, а TensorFlow – в качестве фреймворка для машинного обучения. TensorFlow является одной из наиболее популярных библиотек для обучения нейронных сетей и имеет возможность создавать рекуррентные, свёрточные сети и другие. Кроме этого, данное решение имеет возможность распараллеливания вычислений по архитектуре CUDA, что позволяет увеличить производительность обучения на графических процессорах Nvidia. В свою очередь, Keras – простой и удобный инструмент, изначально поддерживаемый TensorFlow, что позволяет упростить разработку. Эта связка позволит обучить нейронную сеть для предсказания деформаций на геометрии с помощью машинного обучения.

---

<sup>10</sup> от англ. frontend – презентационная часть программной системы, её пользовательский интерфейс и связанные с ним компоненты.

---

Предлагаемый подход позволит решить задачу создания весовых коэффициентов и их аппроксимации при помощи метода глубокого обучения.

Для первоначальных результатов по внедрению анимации для глубокого обучения нейронной сети планируется использовать данные с сервиса Adobe Mixamo. Онлайн-сервисы Mixamo включают магазин анимации с загружаемыми трехмерными моделями и анимационными последовательностями. Анимации были созданы в Mixamo с использованием захвата движения и очищены аниматорами ключевых кадров, что может говорить о высокой точности данных для обучения, что немаловажно для получения качественного результата. Таким образом, гуманоидный риг научится перемещаться в пространстве с помощью установленной демонстрации.

### **Дальнейшее использование инструмента**

Планируется внедрение полученной анимации в редактор Maya, что позволит вносить прямые и быстрые корректировки прямо на ходу, что, как уже было указано выше, может быть обременительной задачей даже для опытных художников трёхмерных моделей, но при чутком контроле художника качество результата будет улучшаться.

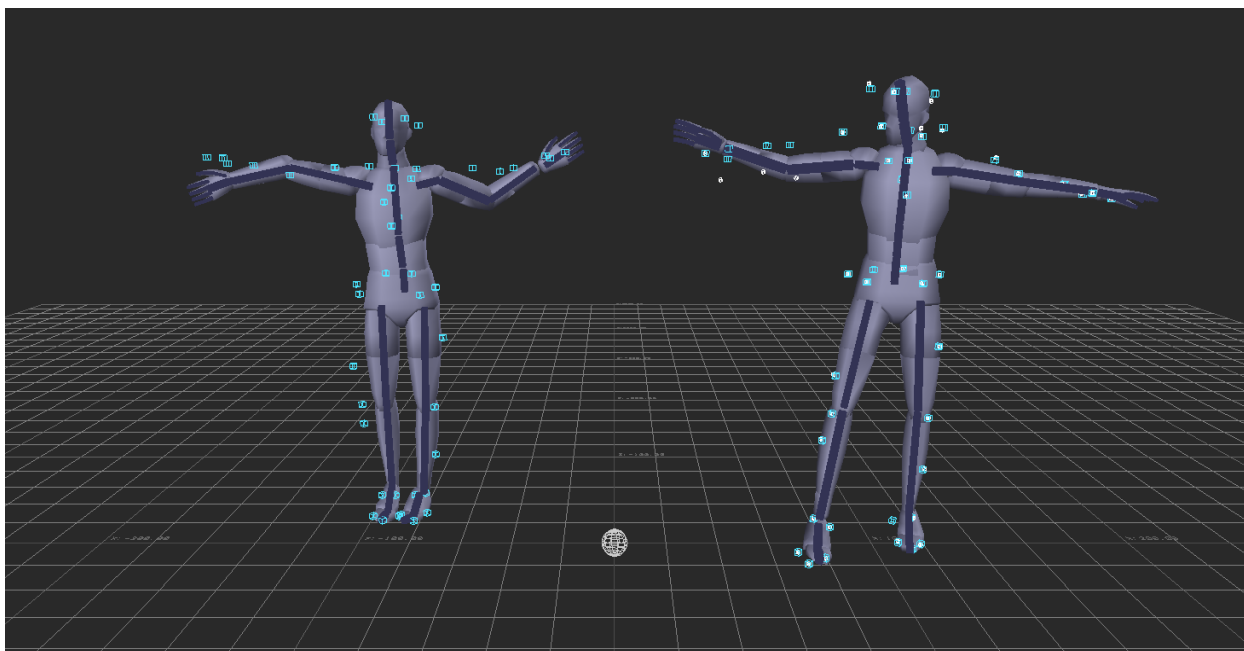


Рис. 11. Создание скелета трёхмерной модели при помощи вводного облака точек

Также предлагается расширить метод для решения задач по извлечению скелета для распознавания или реконструкции облака точек (см. рис. 11), рассмотрев подход с несколькими разрешениями, которые будут уточнять скелет от более общего, грубого – к более детальному. В качестве входных данных могут быть использованы обработанные данные захвата движения, скелетная анимация или видеоматериал.

Подключение автоматического скиннинга с глубоким обучением нейронной сети к инструментарию сценарного прототипирования на основе естественного текста [35, 36] позволит значительно расширить получаемый результат: в такой прототип игры можно будет поиграть, используя не только трехмерные локации, построенные на основе текстовых описаний, но и включить персонажей с реалистичными анимациями.

## **ЗАКЛЮЧЕНИЕ**

Процедура скиннинга – один из ключевых этапов создания анимации для трёхмерных персонажей. От результата выполнения данного этапа напрямую зависит качество будущих анимаций, именно поэтому этот этап и занимает большое количество времени и сил.

Статья описывает множество различных методов и подходов к скиннингу, как ручных, так и автоматических, включая подходы с использованием нейронных сетей. На данный момент эти подходы являются передовыми и помогают ускорить работу художников, но всё ещё не дают полноценного комплексного решения. Процедурные методы скиннинга направлены на то, чтобы облегчить художникам решение стоящей задачи, оставив им лишь итоговую калибровку мелких недочётов. В будущем описанная технология позволит создавать полноценные анимации без помощи специалиста по трёхмерной графике. В связи с этим предложена идея формирования такого инструмента, который мог бы в будущем упростить рассматриваемый процесс, оставив художникам лишь подбор необходимых параметров и выбор наиболее корректного результата.

## **БЛАГОДАРНОСТИ**

Работа выполнена за счет средств Программы стратегического академического лидерства Казанского (Приволжского) федерального университета («ПРИОРИТЕТ-2030»).

## СПИСОК ЛИТЕРАТУРЫ

1. Create an automatic character rig for a mesh // Autodesk Maya. Support and learning. 2020. URL: <https://knowledge.autodesk.com/support/maya/learn-explore/caas/CloudHelp/cloudhelp/2020/ENU/Maya-CharacterAnimation/files/GUID-6CAEA6C2-D4F9-422D-8E0F-522171B47C35-htm.html> (дата обращения 14.07.2022).
2. Nealen A., Müller M., Keiser R., Boxerman E., Carlson M. Physically Based Deformable Models in Computer Graphics // Computer Graphics Forum. 2006. Vol. 25. No. 4. P. 809–836.
3. Lasseeter J. Principles of Traditional Animation Applied to 3D Computer Animation // SIGGRAPH Comput. Graph. 1987. Vol. 21. No. 4. P. 35–44.
4. Gao M., Mitchell N., Sifakis E. Steklov–Poincaré Skinning // In Proceedings of the ACM SIG-GRAPH/Eurographics Symposium on Computer Animation (SCA'14). Eurographics Association. 2014. P. 139–148.
5. McAdams A., Zhu Y., Selle A., Empey M., Tamstorf R., Teran J., Sifakis E. Efficient Elasticity for Character Skinning with Contact and Collisions // ACM Trans. Graph. 2011. Vol. 30. No. 4. P. 37:1–37:12.
6. Smith B., Goes F.D., Kim T. Stable Neo-Hookean Flesh Simulation // ACM Trans. Graph. 2018. Vol. 37. No. 2. P. 12:1–12:15.
7. Teng Y., Otaduy M.A., Kim T. Simulating Articulated Subspace Self-contact // ACM Trans. Graph. 2014. Vol. 33. No. 4. P. 106:1–106:9.
8. Bender J., Müller M., Macklin M. Position-Based Simulation Methods in Computer Graphics // In EG 2015-Tutorials. 2015. 32 p. URL: <https://diglib.eg.org/handle/10.2312/egt.20151045.t3> (дата обращения 14.07.2022).
9. Bouaziz S., Martin S., Liu T., Kavan L., Pauly M. Projective Dynamics: Fusing Constraint Projections for Fast Simulation // ACM Trans. Graph. 2014. Vol. 33. No. 4. P. 154:1–154:11.
10. Rumman N.A., Fratarcangeli M. Position-Based Skinning for Soft Articulated Characters // Computer Graphics Forum. 2015. Vol. 34. No. 6. P. 240–250.
11. Deul C., Bender J. Physically-Based Character Skinning // In Virtual Reality Interactions and Physical Simulations (VRIPhys). Eurographics Association. 2013. 11 p. URL: <http://diglib.eg.org/handle/10.2312/PE.vriphys.vriphys13.025-034> (дата обращения 14.07.2022).



12. *Komaritzan M., Botsch M.* Projective Skinning // Proc. ACM Comput. Graph. Interact. Tech. 2018. Vol. 1. No. 1. P. 12:1–12:19.
13. *Nieto J.R., Susín A.* Cage based deformations: a survey // In Deformation models. Springer. 2013. P. 75–99.
14. *Sederberg T.W., Parry S.R.* Free-form Deformation of Solid Geometric Models // SIGGRAPH Comput. Graph. 1986. Vol. 20. No. 4. P. 151–160.
15. *Lewis J.P., Cordner M., Fong N.* Pose Space Deformation: A Unified Approach to Shape Interpolation and Skeleton-driven Deformation // In Proceedings of SIGGRAPH '00. ACM. 2000. P. 165–172.
16. *Kavan L., Sorkine O.* Elasticity-inspired Deformers for Character Articulation // ACM Trans. Graph. 2012. Vol. 31. No. 6. P. 196:1–196:8.
17. *Rohmer D., Hahmann S., Cani M.P.* Exact Volume Preserving Skinning with Shape Control // In Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer Animation (SCA '09). ACM, 2009. P. 83–92.
18. *Funck W.V., Theisel H., Seidel H.* Volume-preserving Mesh Skinning // Vision, Modelling, and Visualization 2008. Proceedings. 2008. P. 409–414.
19. *Angelidis A., Singh K.* Kinodynamic Skinning Using Volume-preserving Deformations // In Proceedings of the 2007 ACM SIGGRAPH/Eurographics Symposium on Computer Animation (SCA '07). Eurographics Association, 2007. P. 129–140.
20. *Vaillant R., Barthe L., Guennebaud G., Cani M.P., Rohmer D., Wyvill B., Gourmel O., Paulin M.* Implicit Skinning: Real-time Skin Deformation with Contact Modeling // ACM Trans. Graph. 2013. Vol. 32, No. 4. P. 125:1–125:12.
21. *Vaillant R., Guennebaud G., Barthe L., Wyvill B., Cani M.P.* Robust Iso-surface Tracking for Interactive Character Skinning // ACM Trans. Graph. 2014. Vol. 33. No. 6. P. 189:1–189:11.
22. *Aldrich G., Pinskiy D., Hamann B.* Walt Disney Anim. Studios. Collision-Driven Volumetric Deformation on the GPU // In Eurographics (Short Papers). 2011. P. 9–12.
23. *Harmon D., Panozzo D., Sorkine O., Zorin D.* Interference-aware Geometric Modeling // ACM Trans. Graph. 2011. Vol. 30. No. 6. P. 137:1–137:10.
24. *Li Y., Barbič J.* Multi-Resolution Modeling of Shapes in Contact // Symposium on Computer Animation (SCA). 2019.

25. Wang W. A Collision Deformer for Autodesk Maya / Master's thesis // Texas A & M University. 2015. 54 p. URL: <https://core.ac.uk/download/pdf/147244178.pdf> (дата обращения 14.07.2022).

26. Brunel C., Bénard P., Guennebaud G. A time-independent deformer for elastic contacts // ACM Transactions on Graphics. 2021. Vol. 40. No. 4. Art. 3459879.

27. Wang Y., Jacobson A., Barbič J., Kavan L. Linear Subspace Design for Real-time Shape Deformation // ACM Trans. Graph. 2015. Vol. 34. No. 4. P. 57:1–57:11.

28. Aburumman N., Fratarcangeli M. State of the Art in Skinning Techniques for Articulated Deformable Characters // International Conference on Computer Graphics Theory and Applications. Rome. 2016. Vol. 1. P. 198–210. <https://doi.org/10.5220/0005720101980210>.

29. Dionne O., De Lasa M. Geodesic voxel binding for production character meshes // In Proceedings SCA 2013: 12th ACM SIGGRAPH / Eurographics Symposium on Computer Animation, 2013. P. 173.

30. Rotenberg S. Skin / Chapter in course CSE169 // Computer Science & Engineering. 2016. URL: <https://cseweb.ucsd.edu/classes/sp16/cse169-a/readings/3-Skin.html> (дата обращения 14.07.2022).

31. Ju T., Zhou Q., V.D. Panne M., Cohen-Or D., Neumann U. Reusable Skinning Templates Using Cage-based Deformations // ACM Trans. Graph. 2008. Vol. 27. No. 5. Art. 122. 10 p. URL: [https://www.cse.wustl.edu/~taoju/research/skinning\\_final2.pdf](https://www.cse.wustl.edu/~taoju/research/skinning_final2.pdf) (дата обращения 14.07.2022).

32. Rohmer D., Tarini M., Kalyanasundaram N., Moshfeghifar F., Cani M., Zordan V. Velocity Skinning for Real-time Stylized Skeletal Animation // Computer Graphics Forum. 2021. Vol. 40. No. 2. P. 549–561. <https://doi.org/10.1111/cgf.142654>.

33. Zhan X., Yang Z., Evangelos K., Chris L., Karan S. RigNet: Neural Rigging for Articulated Characters // ACM Trans. Graph. 2020. Vol. 39. No. 4. P. 1–14.

34. Liu L., Zheng Y., Tang D., Yuan Y., Fan C., Zhou K. NeuroSkinning: Automatic Skin Binding for Production Characters with Deep Graph Networks // ACM Trans. Graph. 2019. Vol. 38. No. 4. P. 1–12.

35. *Sahibgareeva G., Kugurakova V.* Branched Structure Component for a Video Game Scenario Prototype Generator // CEUR Workshop Proceedings. 2021. Vol. 3066. P. 101–111.

36. *Sahibgareeva G., Bedrin O., Kugurakova V.* Visualization Component for the Scenario Prototype Generator as a Video Game Development Tool // CEUR. Proceedings of the 22nd Conference on Scientific Services & Internet (SSI-2020). 2020. Vol. 2784. P. 267-282.

---

## PROCEDURAL METHODS FOR SKINNING HUMANOID CHARACTERS

**R. R. Gazizov**<sup>1</sup> [0000-0002-8349-264X], **A. V. Shubin**<sup>2</sup> [0000-0002-6203-3268]

<sup>1,2</sup> *Institute of Information Technologies and Intelligent Systems, Kazan (Volga Region) Federal University, ul. Kremlyovskaya, 35, Kazan, 420008*

<sup>1</sup>gazizov782@gmail.com, <sup>2</sup>shubin.aleksey.kpfu@gmail.com

### Abstract

The procedure for setting vertex weights is a very time consuming and difficult task for any 3D model artist. Therefore, the use of procedural methods to facilitate this procedure is very important.

This article analyzes various skinning techniques and identifies their advantages and disadvantages. The most frequent variants of skinning defects that arise when using standard approaches are described. The analysis of tools for skinning in the Maya 3D modeling environment has been carried out. Methods for solving some of the existing problems are proposed, but do not imply a procedural solution. Also, on the basis of neural networks, an idea of their own solution was proposed as an additional tool for the Maya program. This tool will overcome most of the disadvantages of other methods and speed up the skinning process of the model.

**Keywords:** *3D modeling, vertexes, rigging, neural networks.*

### REFERENCES

1. Create an automatic character rig for a mesh // Autodesk Maya. Support and learning. 2020. URL: <https://knowledge.autodesk.com/support/maya/learn-explore/>

---

caas/CloudHelp/cloudhelp/2020/ENU/Maya-CharacterAnimation/files/GUID-6CAEA6C2-D4F9-422D-8E0F-522171B47C35-htm.html.

2. Nealen A., Müller M., Keiser R., Boxerman E., Carlson M. Physically Based Deformable Models in Computer Graphics // Computer Graphics Forum. 2006. Vol. 25. No. 4. P. 809–836.

3. Lasseeter J. Principles of Traditional Animation Applied to 3D Computer Animation // SIGGRAPH Comput. Graph. 1987. Vol. 21. No. 4. P. 35–44.

4. Gao M., Mitchell N., Sifakis E. Steklov–Poincaré Skinning // In Proceedings of the ACM SIG-GRAPH/Eurographics Symposium on Computer Animation (SCA'14). Eurographics Association. 2014. P. 139–148.

5. McAdams A., Zhu Y., Selle A., Empey M., Tamstorf R., Teran J., Sifakis E. Efficient Elasticity for Character Skinning with Contact and Collisions // ACM Trans. Graph. 2011. Vol. 30. No. 4. P. 37:1–37:12.

6. Smith B., Goes F.D., Kim T. Stable Neo-Hookean Flesh Simulation // ACM Trans. Graph. 2018. Vol. 37. No. 2. P. 12:1–12:15.

7. Teng Y., Otaduy M.A., Kim T. Simulating Articulated Subspace Self-contact // ACM Trans. Graph. 2014. Vol. 33. No. 4. P. 106:1–106:9.

8. Bender J., Müller M., Macklin M. Position-Based Simulation Methods in Computer Graphics // In EG 2015-Tutorials. 2015. 32 p. URL: <https://diglib.eg.org/handle/10.2312/egt.20151045.t3> (дата обращения 14.07.2022).

9. Bouaziz S., Martin S., Liu T., Kavan L., Pauly M. Projective Dynamics: Fusing Constraint Projections for Fast Simulation // ACM Trans. Graph. 2014. Vol. 33. No. 4. P. 154:1–154:11.

10. Rumman N.A., Fratarcangeli M. Position-Based Skinning for Soft Articulated Characters // Computer Graphics Forum. 2015. Vol. 34. No. 6. P. 240–250.

11. Deul C., Bender J. Physically-Based Character Skinning // In Virtual Reality Interactions and Physical Simulations (VRIPhys). Eurographics Association. 2013. 11 p. URL: <http://diglib.eg.org/handle/10.2312/PE.vriphys.vriphys13.025-034>.

12. Komaritzan M., Botsch M. Projective Skinning // Proc. ACM Comput. Graph. Interact. Tech. 2018. Vol. 1. No. 1. P. 12:1–12:19.

13. Nieto J.R., Susín A. Cage based deformations: a survey // In Deformation models. Springer. 2013. P. 75–99.

14. *Sederberg T.W., Parry S.R.* Free-form Deformation of Solid Geometric Models // SIGGRAPH Comput. Graph. 1986. Vol. 20. No. 4. P. 151–160.

15. *Lewis J.P., Cordner M., Fong N.* Pose Space Deformation: A Unified Approach to Shape Interpolation and Skeleton-driven Deformation // In Proceedings of SIGGRAPH '00. ACM. 2000. P. 165–172.

16. *Kavan L., Sorkine O.* Elasticity-inspired Deformers for Character Articulation // ACM Trans. Graph. 2012. Vol. 31. No. 6. P. 196:1–196:8.

17. *Rohmer D., Hahmann S., Cani M.P.* Exact Volume Preserving Skinning with Shape Control // In Proceedings of the 2009 ACM SIGGRAPH/Eurographics Symposium on Computer Animation (SCA '09). ACM, 2009. P. 83–92.

18. *Funck W.V., Theisel H., Seidel H.* Volume-preserving Mesh Skinning // Vision, Modelling, and Visualization 2008. Proceedings. 2008. P. 409–414.

19. *Angelidis A., Singh K.* Kinodynamic Skinning Using Volume-preserving Deformations // In Proceedings of the 2007 ACM SIGGRAPH/Eurographics Symposium on Computer Animation (SCA '07). Eurographics Association, 2007. P. 129–140.

20. *Vaillant R., Barthe L., Guennebaud G., Cani M.P., Rohmer D., Wyvill B., Gourmel O., Paulin M.* Implicit Skinning: Real-time Skin Deformation with Contact Modeling // ACM Trans. Graph. 2013. Vol. 32, No. 4. P. 125:1–125:12.

21. *Vaillant R., Guennebaud G., Barthe L., Wyvill B., Cani M.P.* Robust Iso-surface Tracking for Interactive Character Skinning // ACM Trans. Graph. 2014. Vol. 33. No. 6. P. 189:1–189:11.

22. *Aldrich G., Pinskiy D., Hamann B.* Walt Disney Anim. Studios. Collision-Driven Volumetric Deformation on the GPU // In Eurographics (Short Papers). 2011. P. 9–12.

23. *Harmon D., Panozzo D., Sorkine O., Zorin D.* Interference-aware Geometric Modeling // ACM Trans. Graph. 2011. Vol. 30. No. 6. P. 137:1–137:10.

24. *Li Y., Barbič J.* Multi-Resolution Modeling of Shapes in Contact // Symposium on Computer Animation (SCA). 2019.

25. *Wang W.* A Collision Deformer for Autodesk Maya / Master's thesis // Texas A & M University. 2015. 54 p. URL: <https://core.ac.uk/download/pdf/147244178.pdf>.

26. *Brunel C., Bénard P., Guennebaud G.* A time-independent deformer for elastic contacts // ACM Transactions on Graphics. 2021. Vol. 40. No. 4. Art. 3459879.

27. Wang Y., Jacobson A., Barbič J., Kavan L. Linear Subspace Design for Real-time Shape Deformation // ACM Trans. Graph. 2015. Vol. 34. No. 4. P. 57:1–57:11.

28. Aburumman N., Fratarcangeli M. State of the Art in Skinning Techniques for Articulated Deformable Characters // International Conference on Computer Graphics Theory and Applications. Rome. 2016. Vol. 1. P. 198–210.

<https://doi.org/10.5220/0005720101980210>.

29. Dionne O., De Lasa M. Geodesic voxel binding for production character meshes // In Proceedings SCA 2013: 12th ACM SIGGRAPH / Eurographics Symposium on Computer Animation, 2013. P. 173.

30. Rotenberg S. Skin / Chapter in course CSE169 // Computer Science & Engineering. 2016.

URL: <https://cseweb.ucsd.edu/classes/sp16/cse169-a/readings/3-Skin.html>.

31. Ju T., Zhou Q., V.D. Panne M., Cohen-Or D., Neumann U. Reusable Skinning Templates Using Cage-based Deformations // ACM Trans. Graph. 2008. Vol. 27. No. 5. Art. 122. 10 p. URL: [https://www.cse.wustl.edu/~taoju/research/skinning\\_final2.pdf](https://www.cse.wustl.edu/~taoju/research/skinning_final2.pdf).

32. Rohmer D., Tarini M., Kalyanasundaram N., Moshfeghifar F., Cani M., Zordan V. Velocity Skinning for Real-time Stylized Skeletal Animation // Computer Graphics Forum. 2021. Vol. 40. No. 2. P. 549–561. <https://doi.org/10.1111/cgf.142654>.

33. Zhan X., Yang Z., Evangelos K., Chris L., Karan S. RigNet: Neural Rigging for Articulated Characters // ACM Trans. Graph. 2020. Vol. 39. No. 4. P. 1–14.

34. Liu L., Zheng Y., Tang D., Yuan Y., Fan C., Zhou K. NeuroSkinning: Automatic Skin Binding for Production Characters with Deep Graph Networks // ACM Trans. Graph. 2019. Vol. 38. No. 4. P. 1–12.

35. Sahibgareeva G., Kugurakova V. Branched Structure Component for a Video Game Scenario Prototype Generator // CEUR Workshop Proceedings. 2021. Vol. 3066. P. 101–111.

36. Sahibgareeva G., Bedrin O., Kugurakova V. Visualization Component for the Scenario Prototype Generator as a Video Game Development Tool // CEUR. Proceedings of the 22nd Conference on Scientific Services & Internet (SSI-2020). 2020. Vol. 2784. P. 267-282.

## СВЕДЕНИЯ ОБ АВТОРАХ



**ГАЗИЗОВ Рим Радикович** – старший преподаватель Института информационных технологий и интеллектуальных систем (ИТИС) Казанского федерального университета (КФУ). Область научных интересов: компьютерная графика, 3D-моделирование, анимации, виртуальная реальность.

**Rim Radikovich GAZIZOV** – senior lecturer, assistant of the Institute of Information Technologies and Intelligent Systems, Kazan Federal University. Research interests: computer graphics, 3D modeling, 3D animation, VR.

e-mail: gazizov782@gmail.com

ORCID: 0000-0002-8349-264X



**ШУБИН Алексей Витальевич** – лаборант кафедры программной инженерии Института ИТИС КФУ. Сфера научных интересов – разработка видеоигр, игровой дизайн.

**Aleksey Vitalevich SHUBIN** – lab assistant at the Department of Software Engineering of the Institute of ITIS KFU. Research interest – videogame development, game design.

e-mail: shubin.aleksey.kpfu@gmail.com

ORCID: 0000-0002-6203-3268

*Материал поступил в редакцию 15 сентября 2022 года*

УДК 004.946; 62.519

## VR-ТЕЛЕУПРАВЛЕНИЕ «МНОГОРУКИМИ» УСТРОЙСТВАМИ: ПРОБЛЕМЫ, ГИПОТЕЗЫ, ПОСТАНОВКА ЗАДАЧИ

В. В. Кугуракова<sup>1</sup> [0000-0002-1552-4910], И. Д. Сергунин<sup>2</sup> [0000-0003-4025-0023],  
Е. Ю. Зыков<sup>3</sup> [0000-0002-3014-2507], А. В. Уланов<sup>4</sup> [0000-0002-7336-5276],  
О. Д. Сергунин<sup>5</sup> [0000-0002-8134-589X], Д. Р. Габдуллина<sup>6</sup> [0000-0002-4877-924X],  
А. Ш. Гилемянов<sup>7</sup> [0000-0001-8237-7589]

<sup>1, 3, 6</sup> *Институт информационных технологий и интеллектуальных систем  
Казанского федерального университета, ул. Кремлевская, д.35, г. Казань,  
420008*

<sup>2, 4, 5, 7</sup> *АО «Аринет Спейс», ул. Шаляпина 14/83, г. Казань, 420049*

<sup>1</sup>vlada.kugurakova@gmail.com, <sup>2</sup>sergunin.igor@arinet.space, <sup>3</sup>Evgeniy.Zykov@kpfu.ru,  
<sup>4</sup>ulanov.alexey@arinet.space, <sup>5</sup>sergunin.oleg@arinet.space,  
<sup>6</sup>dinaragabdullina086@gmail.com, <sup>7</sup>gilemyanov.artem@arinet.space

### **Аннотация**

Рассмотрены различные решения, существующие в области дистанционного управления роботизированными устройствами, оснащенными манипуляторами. Представлены новые подходы к организации совместного телеуправления множеством манипуляторов, с использованием различных пользовательских входов. Проанализированы следующие сценарии использования: архитектура системы с множеством манипуляторов и пользовательские интерфейсы управления, включая такие перспективные направления, как глубокое машинное обучение и нейроинтерфейсы.

**Ключевые слова:** виртуальная реальность, телеуправление, робот, ко-бот, робототехника, совместное телеуправление, телеимпенданс, когнитивное радио.

### **ВВЕДЕНИЕ**

Навигация по станции с ядерным реактором, обезвреживание бомбы или ремонт международной космической станции снаружи – работы, очень опасные для



людей, а сохранение жизни человека и обеспечение его безопасности являются приоритетной задачей. Однако даже самые совершенные автономные роботы пока не в состоянии выполнять слишком сложные задачи, требующие захвата и манипулирования [1], поэтому дистанционное управление роботизированным устройством в реальном времени часто является практической альтернативой – человек может использовать свою ловкость и свои навыки для более точных манипуляций, без необходимости присутствовать физически. Последние технологические достижения в области робототехники и систем захвата движения значительно уменьшили классические ограничения телеуправления.

Эффективность телеуправления доказана во многих областях, например, при исследовании космоса [2–4], при проведении спасательных работ [5], [6], под водой [7] и в промышленных условиях [8].

Решая проблемы доступа к удаленной системе для работы в неизвестных, опасных или динамически изменяемых средах через телеуправление, необходимо максимально использовать когнитивные навыки человека. Наш интерес направлен на активное использование телеуправления как для навигации роботизированными устройствами, так и для удаленных операций (телеопераций, телеманипуляций).

Управление как процесс представляет собой организацию целенаправленного воздействия на *объект управления* (см. ГОСТ Р МЭК 60447-2000). По терминологии, принятой в 1960 г. на Первом конгрессе Международной федерации по автоматическому управлению, автоматизированная система с человеком-оператором в контуре управления представляет собой *эргатическую систему управления* (ЭСУ).

По классическому определению (см. [9]): «*робототехническая система (робот<sup>1</sup>)* – это совокупность механизмов и устройств (электронных, электрических, оптических, пневматических, гидравлических и т. п.), формирующих единую систему, в том числе имеющую распределенную сетевую структуру и выполняющую рабо-

---

<sup>1</sup> Робот – (от лат. *robota* «подневольный труд») автоматическое устройство, предназначенное для осуществления различного рода механических операций, которое действует по заранее заложеной программе.

---

чие операции, связанные со сложными пространственными перемещениями, способную к целесообразному поведению в условиях изменяющейся внешней обстановки без непосредственного участия человека за счет автоматизации процессов получения, преобразования, передачи и использования энергии, информации и материалов».

Там же [9]: «*манипуляционный робот* представляет собой пространственный управляемый механизм, заканчивающийся схватом или иным рабочим инструментом, с помощью которого можно перемещать объекты в рабочем пространстве, либо выполнять технологические операции».

В [9] выделены разные группы манипуляционных роботов, в частности, «*коллаборативные роботы (или ко-боты)*», представляющие собой конструкции управляемой жесткости, оснащенные системами силомоментного осязательства и управления по силе, которые в свою очередь могут иметь переизбыточные бионические манипуляторы или двурукие манипуляторы».

Не отказываясь от общеупотребительных терминов «робот» или «ко-бот», будем выделять устройства, не действующие по заранее заложенным программам, именуя их роботизированными единицами или дистанционно пилотируемые устройствами.

### **Связанные работы**

В системах телеоперации движениями ведомого робота можно управлять с помощью различных интерфейсов, таких как джойстик [10], тактильный интерфейс [11], [12], 3-D мышь [13], [14], системы захвата движения [15–18] или гарнитуры виртуальной реальности. Однако для пользователя прямое отображение движений является наиболее интуитивным и эффективным методом управления роботом [19].

**Телеимпеданс.** Успех телеопераций также зависит от способности системы адаптировать свое взаимодействие с окружающей средой. В этом контексте парадигма телеимпеданса<sup>2</sup> [20] является мощным инструментом, поскольку она предусматривает оценку сопротивления руки пользователя с помощью поверхностной

---

<sup>2</sup> Телеимпеданс – (англ. impedance от лат. impedio «препятствовать»).

электромиографии и его воспроизведение на роботизированном аналоге. Так, удаленный манипулятор имитирует динамическое поведение человека, повышая безопасность, адаптивность и эффективность. Концепция телеимпедансного управления была введена в [21], причем эффективность этой концепции управления подтверждается несколькими сценариями взаимодействия (например, см. [22]).

Телеимпеданс заключается в воспроизведении желаемых траекторий движения и соответствующих профилей жесткости оператора податливым ведомым роботом в режиме реального времени. Это позволяет оператору регулировать силы взаимодействия между роботом и окружающей средой, избегая при этом проблем с устойчивостью замкнутого контура, возникающих при использовании традиционных двусторонних интерфейсов телеоперации. В то время как движения человека можно отслеживать с помощью точных и экономически эффективных внешних устройств, в современной литературе отсутствуют подходящие и вычислительно эффективные методы оценки импеданса конечностей в реальном времени. Особенно это касается сложных манипуляционных задач, которые исследуют все возможности динамики человеческой руки, а традиционные методы оценки траекторий крутящего момента и жесткости суставов человека требуют реализации нескольких этапов, от измерения биосигналов до кинематики и динамики опорно-двигательного аппарата.

Хорошо известно, что люди по-разному модулируют вязкоупругие свойства конечности. Один из подходов к этому – коконтракция групп мышц, воздействующих на конечность [23]. Альтернативно это может осуществляться путем адаптации чувствительности рефлекторной обратной связи [24] или избирательного контроля конфигурации конечности [25]. Традиционно совместное действие этих механизмов модуляции жесткости в конечной точке руки исследуется путем приложения к руке силы и исследования реакции смещения [26, 27]. За этим обычно следует этап автономной постобработки для оценки параметров импеданса. Применение таких методов для оценки динамических профилей импеданса в многосуставных движениях руки было подробно изучено [28].

**Совместное управление.** С ростом сложности задач роботизированного манипулирования может возникнуть необходимость одновременного использова-

ния и координации нескольких роботов, поскольку они также увеличивают рабочее пространство и диапазон грузов, которыми можно манипулировать. В частности, все более популярным становится бимануальное манипулирование [29, 30], для которого требуются двурукие роботизированные системы, где манипуляторы координируются для достижения общей цели [31].

С этой целью в системах телеоперации, где человек и робот выполняют общую задачу, была введена концепция совместного управления, по которой робот работает с определенной степенью автономности, чтобы уменьшить усилия пользователя. Чтобы извлечь выгоду из совместного управления в «многоруких» системах, в [32] использован словарь бимануальных движений для объединения пользовательских входов управления переводом и вращением с известными траекториями перевода и вращения. В [33] контактные силы и ориентация объекта регулируются самостоятельно, так что оператору нужно только управлять положением трехосевого ведущего тактильного устройства (3-DoF). В [34] использован метод обнаружения объектов на основе зрения для автоматизации захвата, но захват реализован только для одной руки. В [35] описаны задачи, требующие участия двух человек, поэтому две роботизированные руки управляются через двухпедальный ножной интерфейс. В [36] показано, что поддержка и перемещение груза с помощью синергетической координации роботов особенно эффективны и подходят для симметричных задач.

В ряде работ российских исследователей (см., например, [37]) представлены математические модели формирования команд совместного управления объектом и модели управляемого движения объекта в пространстве состояний. Государственный стандарт Российской Федерации ГОСТ Р МЭК 60447-2000 устанавливает общие требования к приводимым в действие вручную органам управления электрическим оборудованием, машинами или технологическими процессами, составляющими человеко-машинные интерфейсы.

**«Многорукие» устройства.** Все упомянутые выше исследования реализуют различные стратегии телеманипуляции с совместным управлением, но всегда ограничиваются телеманипуляцией двумя руками. В литературе мало исследований, посвященных использованию более двух манипуляторов в телеоперации. В

[38] собраны демонстрации задач для имитационного обучения с тремя роботизированными руками, но управляемыми из разных мест через смартфон отдельными пользователями на каждую руку. В [39] рассмотрен сценарий манипулирования «многорукими» устройствами с использованием трех или четырех рук и предложен метод увеличения индекса манипулируемости каждой руки, но не рассмотрены ситуации приближения, захвата и отпускания объекта. В [40] описана схема телеманипуляции, расширяющая предложенную в [41] и снимающая конфигурационное ограничение, когда одна механическая рука управляет концевыми захватами, но все руки всегда обращены друг к другу, и можно работать с любым количеством манипуляторов. В названной работе предложены:

- обобщение режима координированного управления для обеспечения любого возможного относительного положения между конечными эффекторами<sup>3</sup>;
- расширение режимов как независимого, так и координированного управления для управления любым количеством роботов одной рукой;
- отказ и сохранение управления предпочтительными роботами по запросу;
- создание менеджера ресурсов (роботов), который позволяет пользователю легко выбирать и перенастраивать управляемые манипуляторы и режимы управления на лету.

Благодаря таким возможностям пользователь может воспользоваться преимуществами высокой ловкости при соединении один на один, а также комбинированным управлением несколькими роботами с интеграцией общей автономии в структуру. Такая система очень гибка за счет интуитивной реконфигурируемости.

Итак, «многорукая» телеманипуляционная система может обеспечить интуитивное управление как отдельными манипуляторами, так и различными комбинациями любого количества роботизированных рук, предлагая синхронизировать действия в режиме реального времени. С помощью представленной архитектуры человек-оператор может выбирать предлагаемые способы управления и манипу-

---

<sup>3</sup> Эффектор (от лат. effectus действие, результат, впечатление) – как правило, часть устройства, непосредственно влияющая на окружающую среду, например, руки человека; применяется также и как часть устройства, воспринимающая воздействие окружающей среды [42].

---

ляторы, которые делают задачу удобной для выполнения. Эксперименты, описанные выше, демонстрируют преимущества различных способов телеуправления (или удаленного управления несколькими роботами), так как пользователям удастся выполнить сложную задачу, которая было невозможно реализовать одной или даже двумя роботизированными руками.

В [43] предложена довольно гибкая структура, в которой любое количество роботов может одновременно управляться посредством телеоперации с возможностью выбора различных подгрупп рук, размещенных на этих удаленных роботах, в соответствии с выполняемой задачей. Этот подход обеспечивает интуитивное управление различными комбинациями любого количества роботизированных рук.

Для повышения удобства использования и улучшение пользовательского опыта при изменении конфигурации архитектуры управления устаревшие интерфейсы (типа джойстика) должны быть заменены более продвинутыми. Это позволит увеличить разнообразие пользовательских входов и поможет обогатить контроль управления удаленной системой.

**Управление через смартфон.** Еще одним способом удаленного управления роботизированной единицей является смартфон. В этом случае не требуется специфическая техника, пользователем может стать любой человек. Главной проблемой в использовании смартфона является интерфейс, ограниченный небольшими размерами экрана, а также огромным количеством кнопок и слайдеров, управляющих отдельными манипуляторами робота [44].

**Компьютерное зрение.** Алгоритмы компьютерного зрения для навигации роботизированной единицы также находят свое применение. В [45] описано использование библиотеки OpenCV для распознавания препятствий и их преодоления в городской среде, что применяется для распознавания лестниц с использованием карт глубины. Задача адаптивного контроля нескольких роботизированных единиц при помощи лидара также является проблемой при погружении робота в среду с препятствиями, для оптимальной работы сети. В [46] приведено решение данной проблемы с учетом возникающих неисправностей у агентов.

**Виртуальная реальность (VR).** VR в качестве интерфейса для телеуправления роботами дает одновременно захватывающий и детализированный пользовательский опыт. Иммерсивные среды позволяют достичь значительных результатов в любой области применения (см., например, [47] или [48]). Примеров, позволяющих удаленно управлять роботом дистанционно с помощью VR, тоже уже достаточно (см., например, [49–51]). Наш собственный опыт VR-телеуправления представлен в [52] и посвящен как вопросам обеспечения визуальной обратной связи от удаленного роботизированного устройства в режиме стереозрения, так и проблемам получения видеопотока в реальном времени независимо от местоположения устройства.

Другим примером использования виртуальной реальности для управления является проект IVRE [50]. Функционал IVRE позволяет манипулировать реальными физическими объектами, эмулировать захват различных виртуальных объектов, с возможностью редактирования их формы, размера и положения в виртуальном пространстве. Пользователю предоставляются виртуальный двойник робота и дисплей для отображения того, что происходит на небольшом пространстве стола с манипулируемыми предметами в реальном мире.

**Машинное обучение.** В работе [49] говорится об использовании виртуальной реальности для управления роботом. Подчеркнуто удобство использования контроллеров для управления манипуляторами. Действия для управления и захвата объектов являются естественными для человека, чем обусловлен низкий порог вхождения. В ходе названного исследования была реализована система телеуправления PR2 роботом при помощи шлема виртуальной реальности и контроллеров. Для более точного захвата объектов применено обучение на основе подражания (демонстраций). Такая форма обучения является новым и интересным подходом к приобретению роботами различных навыков. Однако получение демонстраций, пригодных для обучения, может оказаться сложной задачей. В анализируемой работе описано, как трекинг рук может быть использован для естественного телеуправления роботами при выполнении сложных задач. Ещё одним интересным направлением будущих исследований является масштабирование такой системы на несколько роботов для более быстрого и параллельного сбора данных.

**Нейроинтерфесы.** Виртуальная реальность позволяет эмулировать для человека те возможности, которые пока трудно достижимы в обыденной жизни, например, симуляцию новых манипуляторов, кроме обычных для человека (таких как руки, пальцы, для управления которыми существуют мышцы и чему человек учится с рождения). Первый пример такого использования, который сразу приходит в голову, – это реализация протеза руки для инвалидов [53], [54]. Другой пример – роботизированная система «увеличения» тела, которую носит один человек, но управляется более чем одним человеком, например, система Fusion [55] имеет две роботизированные конечности, управляемые телеоператором, в то время как роботизированная «голова» камеры, прикрепленная к суррогату, обеспечивает визуальную обратную связь. Таким образом, телеоператор и суррогат могут «делиться телом» и взаимодействовать удаленно.

Однако спектр возможностей можно бесконечно расширять, описывая не только третью руку, но и вторую пару рук или любые другие расширения человеческого тела – в литературе уже появились устойчивые выражения «третья рука» (third arm или imaginary third arm), «паразитическое тело» (parasitic body) [56], «лишние конечности» (supernumerary limbs, supernumerary robotic limbs или SRL) [57], дополнительные виртуальные роборуки (virtual supernumerary robotic arms), «мета-руки» (metaArms) [58]. Детальный обзор использования SRL с текущими техническими проблемами, с которыми приходится сталкиваться, а также применения ключевых технологий представлен в [59]. Но если смотреть на ситуацию несколько шире, чем применение SRL, то можно представить систему управления не только «конечностями»-руками, но и абстракциями, функционал которых позволяет захватывать или иначе воздействовать на другие абстракции. В качестве способа контроля предлагается использовать интерпретацию данных энцефалограммы (EEG), получаемых *in situ*, что уже используется, например, для управления протезом руки [60] или дистанционного управления квадрокоптером [61].

Сценарии использования третьей руки включают двойные задачи, когда пользователь должен выполнять две разные задачи одновременно. Дополнительные конечности могут помочь в выполнении двойных задач быстрее и эффектив-



нее [62], а масштаб владения дополнительными конечностями, о которых сообщают сами пользователи, увеличивается в течение сеансов, в которых пользователи их используют [63].

В [64] авторы исследовали, можно ли получить положительные результаты от владения виртуальным третьим плечом, о котором только что сообщил сам пользователь, переназначая «конечности» по необходимости.

В работе [65] основной фокус разработки находится в области реализации функционала для тренировки через интерфейс мозг–компьютер. Разработан универсальный алгоритм тренировки «призрачных конечностей», которые переключаются пользователем по необходимости на выполнение разных функций. Цель следующих исследований концепции тренировки управления несуществующими конечностями находится в области доказательства того, что ни изменение области применения, ни увеличение количества таких «призрачных конечностей» не влечет за собой экспоненциального роста сложности алгоритма, а зависит лишь от количества подходов к тренировке альтернативных средств контроля. Это утверждение спорное, требует тестирования и обработки репрезентативной выборки.

**Многозвенные манипуляторы.** В работе [66] представлен опыт разработки алгоритмов управления движущимся объектом со многими степенями свободы на примере многозвеного робота-манипулятора. Рассмотрена, в частности, задача согласованного ручного управления человеком-оператором несколькими многозвенными манипуляторами.

## ПОСТАНОВКА ЗАДАЧИ

Необходимо представить архитектуру совместного управления несколькими манипуляторами, размещенными на одном или разных удаленных дистанционно пилотируемых механических устройствах, с нативной реконфигурацией одновременно используемых манипуляторов, включая управление как через традиционные инструменты управления в виртуальной реальности, такие как трекинг рук или контроллеры, так и через тактильную связь (костюм или цифровые перчатки), нейроинтерфейсы или голосовое управление, с визуальным контролем размещения удаленных устройств в режиме стереозрения на основе общего пользования видеопотока в реальном времени, получаемого по сетям.

Минимизация задержек передачи данных является отдельной задачей.

---

## Архитектура системы

На рис. 1 представлена обобщенная схема работы системы многопользовательского управления удаленными роботизированными устройствами, оснащенными множеством манипуляторов.

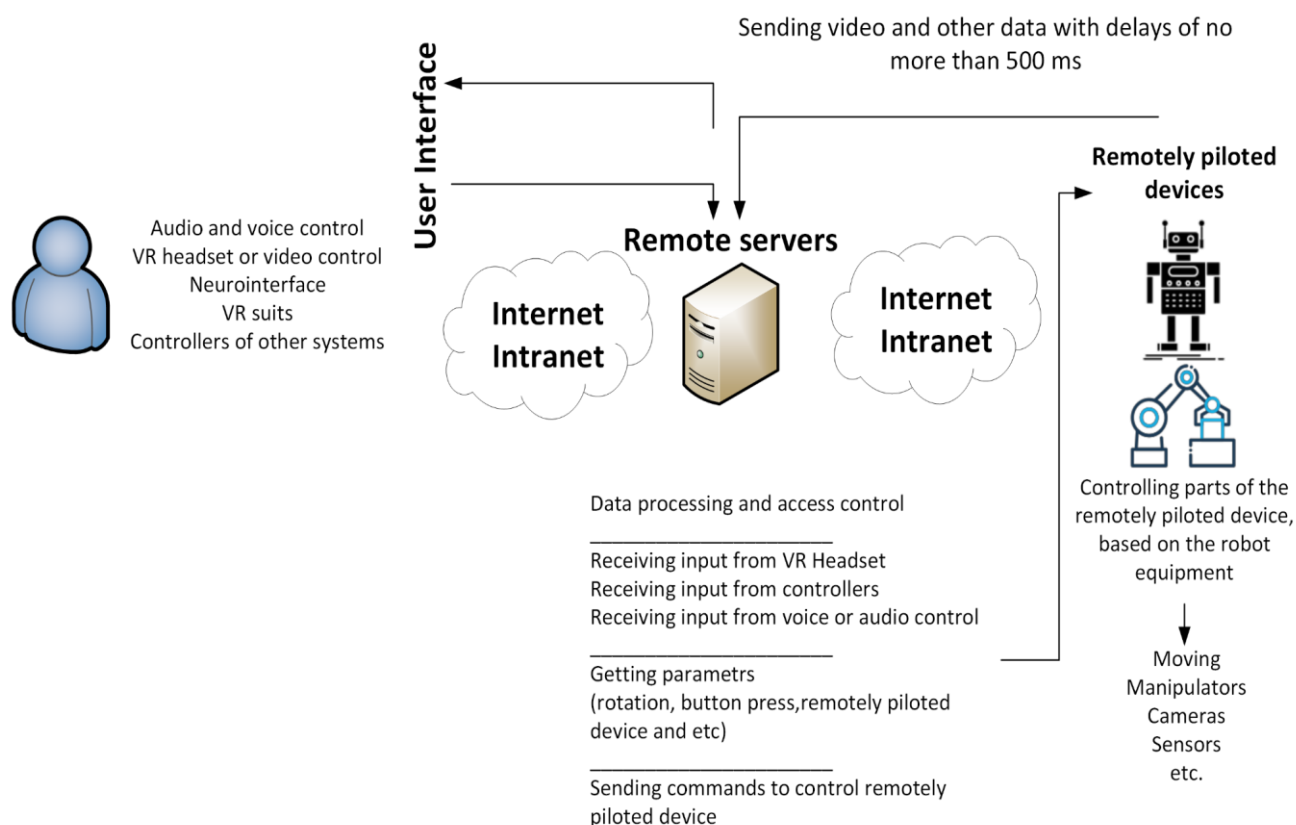


Рис. 1. Схема работы системы многопользовательского управления удаленными роботизированными устройствами

Один или более пользователей (человек-оператор), используя доступные им устройства (ПК, смартфон, гарнитуру VR, костюм VR и т. п.), через свои аккаунты Internet/intranet-платформы выбирают устройство, доступное им для управления, и начинают взаимодействие с удаленным устройством.

Серверы платформы, обрабатывая данные (видео, аудио, команды управления), полученные в режиме реального времени, формируют команды для удаленного устройства в соответствии с его функционалом. Команды поступают на исполнительные приводы удаленного устройства.

При взаимодействии оператор получает обратную связь от устройства в виде потока видео-, аудиоданных, данных с датчиков и сенсоров удаленного устройства. Эти данные позволяют пользователю (оператору) в режиме реального времени (задержка не должна превышать 500 мс) в режиме телеприсутствия управлять удаленным устройством.

В качестве пользовательского входа для управления (UI) могут быть использованы следующие типы устройств: ПК, смартфон, VR гарнитура, нейроинтерфейсные комплексы, костюмы VR и т. п. Выбор интерфейса взаимодействия с удаленным устройством определяется по следующим критериям:

- сложность удаленного устройства и его возможности;
- доступность для оператора (с учетом п. 1);
- пропускная способность сети передачи данных оператора и удаленного устройства.

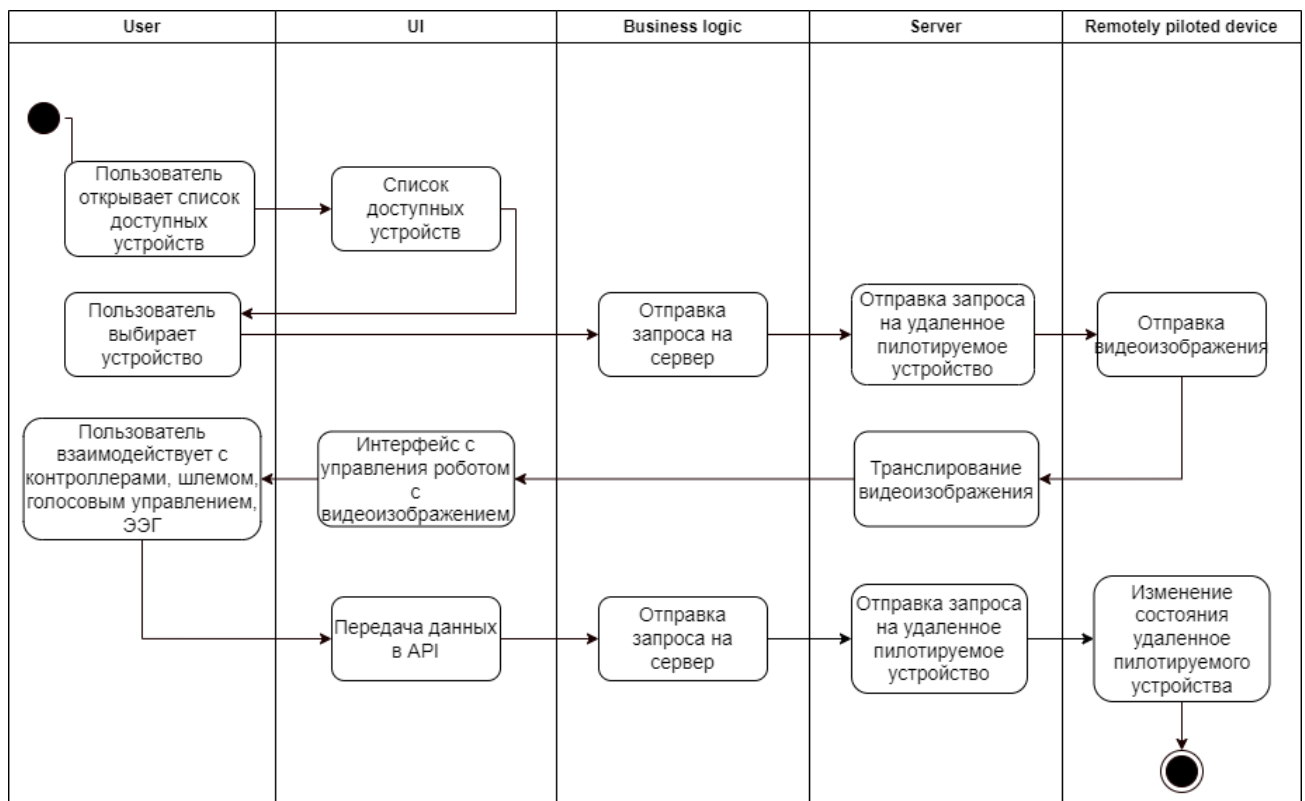


Рис. 2. UML-диаграмма активностей управления удаленным роботизированным устройством

На рис. 2 представлена схема передачи управления от одной подсистемы к другой. На схеме представлены следующие ключевые объекты: пользователь, пользовательский интерфейс, бизнес-логика, сервер и удаленное устройство.

## **БУДУЩИЕ РАБОТЫ**

Представленная система уже реализована в пилотном исполнении, при этом использованы классические протоколы для передачи информации, включая стереоизображение [67], и стоят следующие задачи для её совершенствования: разработка нового кодека для передачи видеопотока в режиме реального времени, разработка новых подходов к использованию самоорганизующихся сетей, разработка алгоритмов обработки данных энцефалограмм в режиме реального времени для телеуправления мета-конечностью, разработка алгоритмов снятия показаний тела.

Также можно предложить следующий метод управления многорукими робототехническими устройствами одним оператором: управление двумя антропоморфными манипуляторами как роботами-копирами, с возможностью подключения или смены дополнительных манипуляторов при помощи тактильных датчиков, VR-гарнитур и других устройств, например, устройств с большей степенью автономностью, которые могут работать по заранее заданным детерминированным алгоритмам.

**Когнитивное радио.** Когда речь идёт о труднодоступных для человека местах, возникает проблема с наличием возможности подключения в принципе и, в частности, покрытием сетями в таких локациях. Сотовые сети позволяют использовать множество политик энергосбережения, имеют широкий переключаемый диапазон скоростей, частот и технологий доступа. Но недостатками использования сотовых сетей являются привязка к поставщику, плохое покрытие в труднодоступных местах, низкая проникающая способность на промышленных объектах [68]. Для создания распределённого сетевого телеуправления системой «многоруких» устройств вне покрытия интернетом необходимо обеспечить самоорганизующуюся сеть intranet в парадигме «когнитивного радио», позволяющую сохранять приемлемые параметры для передачи видеопотоков в режиме реального времени.

Наиболее перспективным стандартом, по нашему мнению, для когнитивных радиосистем является IEEE 802.11af (начало разработки – январь 2010 года), основной целью является адаптация семейства стандартов IEEE 802.11 к телевизионным полосам частот. В Российской Федерации по результатам научно-исследовательской работы, проведенной ФГУП НИИР, подготовлен отчет, на основе которого Государственная комиссия по радиочастотам в 2012 г. приняла решение «О создании опытной зоны по внедрению когнитивных систем широкополосного беспроводного доступа в Российской Федерации в полосе радиочастот 470–686 МГц» с приложением «Основные технические характеристики РЭС когнитивных систем широкополосного беспроводного доступа в полосе радиочастот 470–686 МГц» [69].

### **ЗАКЛЮЧЕНИЕ**

В работе представлена обобщенная схема функционирования системы многопользовательского управления удаленными роботизированными устройствами, оснащенными множеством манипуляторов. Предложены методы управления многорукими робототехническими устройствами одним оператором. Рассмотрена возможность использования передачи данных по когнитивным сетям широкополосного беспроводного доступа по классическим протоколам и поставлены задачи для дальнейшего усовершенствования как непосредственно кодеков для передачи стереовидеосигналов, так и алгоритмов обработки данных в режиме реального времени с использованием элементов искусственного интеллекта. Поставленные задачи требуют экспериментальных исследований для получения оптимальных конструктивных аппаратных и программных схемотехнических решений, которые смогут в дальнейшем быть использованы в различных областях промышленности.

### **Благодарности**

Работа выполнена за счет средств Программы стратегического академического лидерства Казанского (Приволжского) федерального университета («ПРИОРИТЕТ-2030»).

### **СПИСОК ЛИТЕРАТУРЫ**

1. *Pratt G., Manzo J.* The darpa robotics challenge [competitions]. IEEE Robotics & Automation Magazine. 2013. V. 20(2). P. 10–12.

2. *Artigas J. et al.* Kontur-2: Force-feedback teleoperation from the international space station // Proc. IEEE Int. Conf. Robot. Automat. (ICRA). 2016. P. 1166–1173.
3. *Bluethmann W.* Robonaut: A robot designed to work with humans in space // Auton. Robots. 2003. Vol. 14, no. 2. P. 179–197.
4. *Sheridan T.* Space teleoperation through time delay: Review and prognosis // IEEE Trans. Robot. Automat. 1993. Vol. 9, no. 5. P. 592–606.
5. *Negrello F.* Humanoids at work: The walk-man robot in a postearthquake scenario // IEEE Robot. Automat. Mag. 2018. Vol. 25, no. 3. P. 8–22.
6. *Hirche S., Stanczyk B., Buss M.* Transparent exploration of remote environments by internet telepresence // Proc. Int. Workshop High-Fidelity Telepresence Teleaction jointly with Conf. HUMANOIDS. 2003. P. 1–20.
7. *Khatib O. et al.* Ocean one: A robotic avatar for oceanic discovery // IEEE Robot. Automat. Mag. 2016. Vol. 23, no. 4. P. 20–29.
8. *Shukla A., Karki H.* Application of robotics in onshore oil and gas industry-a review part I // Robot. Auton. Syst. 2016. Vol. 75. P. 490–507.
9. *Колюбин С.А.* Динамика робототехнических систем / Учебное пособие // СПб.: Университет ИТМО, 2017. 117 с.
10. *Glynn S., Fekieta R., Henning R.* Use of force-feedback joysticks to promote teamwork in virtual teleoperation // Proc. Hum. Factors Ergonom. Soc. Annu. Meeting. 2001. Vol. 45. P. 1911–1915,
11. *Martinez-Palafox O., Lee D., Spong M.W., Lopez I., Abdallah C.T.* Bilateral teleoperation of mobile robot over delayed communication network: Implementation // Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst., 2006. P. 4193–4198.
12. *Selvaggio M., Giordano P.R., Ficuciello F., Siciliano B.* Passive task- prioritized shared-control teleoperation with haptic guidance // Proc. Int. Conf. Robot. Automat. (ICRA), 2019. P. 430–436.
13. *Materna Z. et al.* Teleoperating assistive robots: A novel user interface relying on semi-autonomy and 3D environment mapping // J. Robot. Mechatronics, 2017. Vol. 29. P. 381–394.
14. *Garate V.R., Gholami S., Ajoudani A.* A scalable framework for multi-robot tele-impedance control // IEEE Trans. Robot., 2021. Vol. 37, no. 6. P. 2052–2066.

15. *Clark J.P., Lentini G., Barontini F., Catalano M.G., Bianchi M., O'Malley M.K.* On the role of wearable haptics for force feedback in teleimpedance control for dual-arm robotic teleoperation // in Proc. Int. Conf. Robot. Automat. (ICRA), 2019. P. 5187–5193.

16. *Zhang T. et al.* Deep imitation learning for complex manipulation tasks from virtual reality teleoperation // Proc. IEEE Int. Conf. Robot. Automat. (ICRA), 2018. P. 5628–5635.

17. *Lipton J.I., Fay A.J., Rus D.* Baxter's homunculus: Virtualreality spaces for teleoperation in manufacturing // IEEE Robot. Automat. Lett., 2018. Vol. 3, no. 1. P. 179–186.

18. *Koenemann J., Burget F., Bennewitz M.* Real-time imitation of human whole-body motions by humanoids // Proc. IEEE Int. Conf. Robot. Automat. (ICRA), 2014. P. 2806–2812.

19. *Rakita D., Mutlu B., Gleicher M.* A motion retargeting method for effective mimicry-based teleoperation of robot arms // Proc. IEEE/ACM 12th Int. Conf. Hum.-Robot Interaction, 2017. P. 361–370.

20. *Ajoudani A., Fang C., Tsagarakis N., Bicchi A.* Reduced-complexity representation of the human arm active endpoint stiffness for supervisory control of remote manipulation // Int. J. Robot. Res., 2018. Vol. 37, no. 1. P. 155–167.

21. *Ajoudani A., Tsagarakis N., Bicchi A.* Tele-impedance: Teleoperation with impedance regulation using a body-machine interface // International Journal of Robotics Research. 2012. Vol. 31(13). P. 1642–1655.

22. *Hocaoglu E., Patoglu V.* Tele-impedance control of a variable stiffness prosthetic hand. // IEEE international conference on robotics and biomimetics (ROBIO). Piscataway, NJ: IEEE. 2012, P. 1576–1582.

23. *Gribble P.L., Mullin L.I., Cothros N. et al.* Role of cocontraction in arm movement accuracy // Journal of Neurophysiology. 2003. Vol. 89(5). P. 2396–2405.

24. *Akazawa K., Milner T.E., Stein R.B.* Modulation of reflex EMG and stiffness in response to stretch of human finger muscle // Journal of Neurophysiology. 1983. Vol. 49(1). P. 16–27.

25. *Trumbower R.D., Krutky M.A., Yang B.S. et al.* Use of self-selected postures to regulate multi-joint stiffness during unconstrained tasks // PLoS ONE. 2009. Vol.4(5). P. e5411.

26. *Mussa-Ivaldi F.A., Hogan N., Bizzi E.* Neural, mechanical, and geometric factors subserving arm posture in humans // *Journal of Neuroscience*. 1985. Vol. 5(10). P. 2732–2743.

27. *Perreault E.J., Kirsch R.F., Crago P.E.* Voluntary control of static endpoint stiffness during force regulation tasks // *Journal of Neurophysiology*. 2002. Vol. 87(6). P. 2808–2816.

28. *Franklin DW, Burdet E, Osu R, et al.* Functional significance of stiffness in adaptation of multijoint arm movements to stable and unstable dynamics // *Experimental Brain Research*. 2003. Vol. 151(2). P. 145–157.

29. *Edsinger A., Kemp C.C.* Two arms are better than one: A behavior based control system for assistive bimanual manipulation // *Recent Progress in Robotics: Viable Robotic Service to Human*. Berlin, Germany: Springer, 2007. P. 345–355.

30. *Makris S., Tsarouchi P., Surdilovic D., Krüger J.* Intuitive dual arm robot programming for assembly operations // *CIRP Ann.*, 2014. Vol. 63, no. 1. P. 13–16.

31. *Smithetal C.* Dual arm manipulation—survey // *Robot.Auton.Syst.*, 2012. Vol. 60, no. 10. P. 1340–1353.

32. *Rakita D., Mutlu B., Gleicher M., Hiatt L.M.* Shared control-based bimanual robot manipulation // *Sci. Robot.*, 2019. Vol. 4, Art. no. eaaw0955.

33. *Sun D., Liao Q., Loutfi A.* Single master bimanual teleoperation system with efficient regulation // *IEEE Trans. Robot.*, 2020. Vol. 36, no. 4. P. 1022–1037.

34. *Lin T.-C., Unni Krishnan A., Li Z.* Shared autonomous interface for reducing physical effort in robot teleoperation via human motion mapping // *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, 2020. P. 9157–9163.

35. *Amanhoud W., Hernandez Sanchez J., Bouri M., Billard A.* Contact-initiated shared control strategies for four-arm supernumerary manipulation with foot interfaces // *Int. J. Robot. Res.*, 2021. Vol. 40, P. 1–29.

36. *Laghi M. et al.* Shared-autonomy control for intuitive bimanual telemanipulation // *Proc. IEEE-RAS 18th Int. Conf. Humanoid Robots (Humanoids)*, 2018. P. 1–9.

37. *Тырва В.О.* Совместное управление объектом в эргатической системе: модели и реализации // *Вестник Государственного университета морского и речного флота имени адмирала С.О. Макарова*. 2018. Т. 10. № 2. С. 430–443.



38. *Tung A. et al.* Learning multi-arm manipulation through collaborative teleoperation // Proc. IEEE Int. Conf. Robot. Automat. (ICRA), 2021. P. 9212–9219.

39. *Kennel-Maushart F., Poranne R., Coros S.* Manipulability optimization for multi-arm teleoperation // Proc. IEEE Int. Conf. Robot. Automat. (ICRA), 2021. P. 3956–3962.

40. *Ozdamar I., Laghi M., Grioli G., Ajoudani A., Bicchi A.* A Shared Autonomy Reconfigurable Control Framework for Telemanipulation of Multi-Arm Systems // IEEE ROBOTICS AND AUTOMATION LETTERS, 2022. Vol. 7, No. 4. P. 9937–9944.

41. *Laghi M. et al.* Shared-autonomy control for intuitive bimanual tele-manipulation // Proc. IEEE-RAS 18th Int. Conf. Humanoid Robots (Humanoids), 2018. P. 1–9.

42. *Ашкинази Л.А.* Мир Лема – словарь и путеводитель. 2004. 1068 с.

43. *Ozdamar I., Laghi M., Grioli A., Ajoudani G., Catalano M.G., Bicchi A.* A Shared Autonomy Reconfigurable Control Framework for Telemanipulation of Multi-Arm Systems // IEEE Robotics and Automation Letters, 2022. V.7(4). P. 9937–9944.

44. *Florea B.C.* Smartphone Controlled Autonomous Robotic Platform // 2018 15th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), Chiang Rai, Thailand, 2018. P. 620–623.

45. *Mustafin M., Tsoy T., Martínez-García E.A., Meshcheryakov R., Magid E.* Modelling mobile robot navigation in 3D environments: camera-based stairs recognition in Gazebo // 2022 Moscow Workshop on Electronic and Networking Technologies (MWENT), Moscow, Russian Federation, 2022. P. 1–6.

46. *Bai Y., Wang Y., Svinin M., Magid E., Sun R.* Adaptive Multi-Agent Coverage Control with Obstacle Avoidance // IEEE Control Systems Letters, 2022. Vol. 6. P. 944–949.

47. *Кугуракова В.В., Антонов И.О., Гончаренко Б.В., Чайбар А.А.* Цифровое представление в виртуальной реальности места происшествия как инструмент уголовного судопроизводства // Программные системы: теория и приложения, 2022. Т. 13, вып. 3. С. 193–223.

48. *Kugurakova, V.V., Golovanova, I.I., Shaidullina, A.R.* Digital Solutions in Educators' Training: Concept for Implementing a Virtual Reality Simulator // Eurasia Journal of Mathematics, Science and Technology Education. 2021. Vol. 17, Is. 9. P.1–10.

49. *Tianhao Zhang, Zoe McCarthy, Owen Jow Dennis, Lee Xi Chen, Ken Goldberg, Pieter Abbeel*. Deep Imitation Learning for Complex Manipulation Tasks from Virtual Reality Teleoperation // eecs.berkeley.edu.

URL: <https://www2.eecs.berkeley.edu/Pubs/TechRpts/2020/EECS-2020-190.pdf> (дата обращения: 01.08.2022).

50. IVRE – An Immersive Virtual Robotics Environment [Электронный ресурс] // [cirl.lcsr.jhu.edu](http://cirl.lcsr.jhu.edu)

URL: <https://cirl.lcsr.jhu.edu/research/human-machine-collaborativesystems/ivre/> (дата обращения: 01.08.2022).

51. Reachy by Pollen Robotics, an open source programmable humanoid robot // [pollen-robotics.com](http://pollen-robotics.com). URL: <https://www.pollen-robotics.com/> (дата обращения: 01.02.2022).

52. *Кузуракова В.В., Хафизов М.Р., Кадыров С.А., Зыков Е.Ю.*, Удаленное управление роботизированным устройством с использованием технологий виртуальной реальности // Программные продукты и системы. 2022. Т. 35. № 3. С. 348–361.

53. *Melero M., Hou A., Cheng E., Tayade A., Lee S.C., Unberath M., Navab N., Cutolo F.* Upbeat: Augmented Reality-Guided Dancing for Prosthetic Rehabilitation of Upper Limb Amputees // *Journal of Healthcare Engineering*. 2019. Art. 2163705.

54. *Nissler C., Nowak M., Connan M., Büttner S., Vogel J., Kossyk I., Márton Z.C., Castellini C.* VITA – An everyday virtual reality setup for prosthetics and upper-limb rehabilitation // *Journal of Neural Engineering*. 2019. Vol. 16(2). Art. 026039.

55. *Saraiji M.Y., Sasaki T., Matsumura R., Minamizawa K., Inami M.* Fusion: full body surrogacy for collaborative communication // *SIGGRAPH '18: ACM SIGGRAPH 2018 Emerging Technologies*. 2018. Vol. 7. P. 1–2.

56. *Takizawa R., Hivarna A., Verhulst A., Seaborn K., Fukuoka M., Kitazaki M., Inami M., Suqirnoto M.* Parasitic body: Exploring perspective dependency in a shared body with a third arm // *26th IEEE Conference on Virtual Reality and 3D User Interfaces. VR 2019 – Proceedings*. 2019. P. 1175–1176.

57. *Parietti F., Asada H.* Supernumerary Robotic Limbs for Human Body Support // *IEEE Transactions on Robotics*. 2016. Vol. 32(2). P. 301–311.

58. *Sasaki T., Saraiji Y., Minamizawa K., Inami M.* MetaArmS: Body remapping using feet-controlled artificial arms // *UIST 2018 Adjunct – Adjunct Publication of the 31st Annual ACM Symposium on User Interface Software and Technology*. 2018. P. 140–142.

59. *Tong Y., Liu J.* Review of Research and Development of Supernumerary Robotic Limbs // *IEEE/CAA Journal of Automatica Sinica*. 2021. Vol. 8. No. 5. P. 929–952.

60. *Aly H.I., Youssef S., Fathy C.* Hybrid Brain Computer Interface for Movement Control of Upper Limb Prostheses. 2018 International Conference on Biomedical Engineering Applications // *ICBEA 2018 – Proceedings*. 2018. P. 1–6.

61. *Aljalal M., Ibrahim S., Djemal R., Ko W.* Comprehensive review on brain-controlled mobile robots and robotic arms based on electroencephalography signals // *Intelligent Service Robotics*. 2020. Vol. 13 (4). P. 539–563.

62. *Abdi E., Burdet E., Bouri M., Himidan S., Bleuler H.*, In a demanding task three-handed manipulation is preferred to two-handed manipulation // *Scientific Reports*. 2016. Vol. 6. P. 1–11.

63. *Sasaki T., Saraiji Y., Minamizawa K., Inami M.* MetaArmS: Body remapping using feet-controlled artificial arms // *UIST 2018 Adjunct – Adjunct Publication of the 31st Annual ACM Symposium on User Interface Software and Technology*. 2018. P. 140–142.

64. *Drogemuller A., Verhulst A., Volmer B., Thomas B.H., Inami M., Sugimoto M.* Remapping a Third Arm in Virtual Reality // *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, Osaka, Japan, 2019. P. 898–899.

65. Виртуальный тренажер управления призрачными конечностями через интерфейс мозг–компьютер: Свидетельство о государственной регистрации программы для ЭВМ № 202211167 Российская Федерация / В.В. Кугуракова, Д.Г. Исмаилов, Д.М. Тимергалин; заявитель и правообладатель Фед. гос. автоном. образоват. учреждение высш. образ. Казанский фед. ун-т. №2022610311; заявл. 13.01.2022; зарегистрировано в реестре программ для ЭВМ 20.01.2022. – [1] с.

66. *Жильцов А.И., Жуков В.С., Рылеев Д.А.* Управление манипуляторами с числом степеней свободы более шести // *Инженерный журнал: наука и инновации*, 2013. Вып. 10. 11 с.

67. Кугуракова В.В., Хафизов М.Р., Кадыров С.А., Зыков Е.Ю. Удаленное управление роботизированным устройством с использованием технологий виртуальной реальности // Программные продукты и системы. 2022. Т. 35. № 3. С. 348–361.

68. Ушакова М. В., Ушаков Ю. А. Исследование энергосберегающей беспроводной самоорганизующейся многопротокольной сети передачи данных интернет устройств // Современные информационные технологии и ИТ-образование. 2019. Т. 15, № 3. С. 733–745.

69. Мирошникова Н.Е. Обзор систем когнитивного радио // T-Comm – Telecommunications and Transport. 2013. № 9. С. 108–111.

---

## VR-TELECONTROL OF MULTI-ARM DEVICES: PROBLEMS, HYPOTHESES, PROBLEM STATEMENT

V. V. Kugurakova<sup>1</sup> [0000-0002-1552-4910], I. D. Sergunin<sup>2</sup> [0000-0003-4025-0023],  
E. Yu. Zykov<sup>3</sup> [0000-0002-3014-2507], A. V. Ulanov<sup>4</sup> [0000-0002-7336-5276],  
O. D. Sergunin<sup>5</sup> [0000-0002-8134-589X], D. R. Gabdullina<sup>6</sup> [0000-0002-4877-924X],  
A. Sh. Gilemyanov<sup>7</sup> [0000-0001-8237-7589]

<sup>1, 3, 6</sup> Institute of Information Technologies and Intelligent Systems of Kazan Federal University, st. Kremlin, 35, Kazan, 420008

<sup>2, 4, 5, 7</sup> JSC "Arinet Space", st. Chaliapin 14/83, Kazan, 420049

<sup>1</sup>vlada.kugurakova@gmail.com, <sup>2</sup>sergunin.igor@arinet.space, <sup>3</sup>Evgeniy.Zykov@kpfu.ru,

<sup>4</sup>ulanov.alexey@arinet.space, <sup>5</sup>sergunin.oleg@arinet.space,

<sup>6</sup>dinaragabdullina086@gmail.com, <sup>7</sup>gilemyanov.artem@arinet.space

### Abstract

The article discusses various solutions that exist in the field of remote control of robotic devices equipped with manipulators. New approaches are presented for organizing joint telecontrol of multiple manipulators using various user inputs. The following usage scenarios are considered: the architecture of a system with many manipulators

---

and user control interfaces, including such promising areas as deep machine learning and neural interfaces.

**Keywords:** virtual reality, telecontrol, robot, co-bot, robotics, joint telecontrol, teleimpedance, cognitive radio.

### **Acknowledgement**

This paper has been supported by the Kazan Federal University Strategic Academic Leadership Program ("PRIORITY-2030").

### **REFERENCES**

1. *Pratt G., Manzo J.* The darpa robotics challenge [competitions]. IEEE Robotics & Automation Magazine. 2013. V. 20(2). P. 10–12.
2. *Artigas J. et al.* Kontur-2: Force-feedback teleoperation from the international space station // Proc. IEEE Int. Conf. Robot. Automat. (ICRA). 2016. P. 1166–1173.
3. *Bluethmann W.* Robonaut: A robot designed to work with humans in space // Auton. Robots. 2003. Vol. 14, no. 2. P. 179–197.
4. *Sheridan T.* Space teleoperation through time delay: Review and prognosis // IEEE Trans. Robot. Automat. 1993. Vol. 9, no. 5. P. 592–606.
5. *Negrello F.* Humanoids at work: The walk-man robot in a postearthquake scenario // IEEE Robot. Automat. Mag. 2018. Vol. 25, no. 3. P. 8–22.
6. *Hirche S., Stanczyk B., Buss M.* Transparent exploration of remote environments by internet telepresence // Proc. Int. Workshop High-Fidelity Telepresence Teleaction jointly with Conf. HUMANOIDS. 2003. P. 1–20.
7. *Khatib O. et al.* Ocean one: A robotic avatar for oceanic discovery // IEEE Robot. Automat. Mag. 2016. Vol. 23, no. 4. P. 20–29.
8. *Shukla A., Karki H.* Application of robotics in onshore oil and gas industry-a review part I // Robot. Auton. Syst. 2016. Vol. 75. P. 490–507.
9. *Kolyubin S.A.* Dinamika robototekhnicheskikh sistem / Uchebnoe posobie // SPb.: Universitet ITMO, 2017. 117 s.
10. *Glynn S., Fekieta R., Henning R.* Use of force-feedback joysticks to promote teamwork in virtual teleoperation // Proc. Hum. Factors Ergonom.Soc. Annu. Meeting. 2001. Vol. 45. P. 1911–1915,

11. *Martinez-Palafox O., Lee D., Spong M.W., Lopez I., Abdallah C.T.* Bilateral teleoperation of mobile robot over delayed communication network: Implementation // Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst., 2006. P. 4193–4198.

12. *Selvaggio M., Giordano P.R., Ficuciello F., Siciliano B.* Passive task- prioritized shared-control teleoperation with haptic guidance // Proc. Int. Conf. Robot. Automat. (ICRA), 2019. P. 430–436.

13. *Materna Z. et al.* Teleoperating assistive robots: A novel user interface relying on semi-autonomy and 3D environment mapping // J. Robot. Mechatronics, 2017. Vol. 29. P. 381–394.

14. *Garate V.R., Gholami S., Ajoudani A.* A scalable framework for multi-robot tele-impedance control // IEEE Trans. Robot., 2021. Vol. 37, no. 6. P. 2052–2066.

15. *Clark J.P., Lentini G., Barontini F., Catalano M.G., Bianchi M., O'Malley M.K.* On the role of wearable haptics for force feedback in teleimpedance control for dual-arm robotic teleoperation // in Proc. Int. Conf. Robot. Automat. (ICRA), 2019. P. 5187–5193.

16. *Zhang T. et al.* Deep imitation learning for complex manipulation tasks from virtual reality teleoperation // Proc. IEEE Int. Conf. Robot. Automat. (ICRA), 2018. P. 5628–5635.

17. *Lipton J.I., Fay A.J., Rus D.* Baxter's homunculus: Virtualreality spaces for teleoperation in manufacturing // IEEE Robot. Automat. Lett., 2018. Vol. 3, no. 1. P. 179–186.

18. *Koenemann J., Burget F., Bennewitz M.* Real-time imitation of human whole-body motions by humanoids // Proc. IEEE Int. Conf. Robot. Automat. (ICRA), 2014. P. 2806–2812.

19. *Rakita D., Mutlu B., Gleicher M.* A motion retargeting method for effective mimicry-based teleoperation of robot arms // Proc. IEEE/ACM 12th Int. Conf. Hum.-Robot Interaction, 2017. P. 361–370.

20. *Ajoudani A., Fang C., Tsagarakis N., Bicchi A.* Reduced-complexity representation of the human arm active endpoint stiffness for supervisory control of remote manipulation // Int. J. Robot. Res., 2018. Vol. 37, no. 1. P. 155–167.

21. *Ajoudani A., Tsagarakis N., Bicchi A.* Tele-impedance: Teleoperation with impedance regulation using a body-machine interface // International Journal of Robotics Research. 2012. Vol. 31(13). P. 1642–1655.

22. Hocaoglu E., Patoglu V. Tele-impedance control of a variable stiffness prosthetic hand. // IEEE international conference on robotics and biomimetics (ROBIO). Piscataway, NJ: IEEE. 2012, P. 1576–1582.

23. Gribble P.L., Mullin L.I., Cothros N. et al. Role of cocontraction in arm movement accuracy // Journal of Neurophysiology. 2003. Vol. 89(5). P. 2396–2405.

24. Akazawa K., Milner T.E., Stein R.B. Modulation of reflex EMG and stiffness in response to stretch of human finger muscle // Journal of Neurophysiology. 1983. Vol. 49(1). P. 16–27.

25. Trumbower R.D., Krutky M.A., Yang B.S. et al. Use of self-selected postures to regulate multi-joint stiffness during unconstrained tasks // PLoS ONE. 2009. Vol. 4(5). Art. e5411.

26. Mussa-Ivaldi F.A., Hogan N., Bizzi E. Neural, mechanical, and geometric factors subserving arm posture in humans // Journal of Neuroscience. 1985. Vol.5(10). P. 2732–2743.

27. Perreault E.J., Kirsch R.F., Crago P.E. Voluntary control of static endpoint stiffness during force regulation tasks // Journal of Neurophysiology. 2002. Vol. 87(6). P. 2808–2816.

28. Franklin DW, Burdet E, Osu R, et al. Functional significance of stiffness in adaptation of multijoint arm movements to stable and unstable dynamics // Experimental Brain Research. 2003. Vol. 151(2). P. 145–157.

29. Edsinger A., Kemp C.C. Two arms are better than one: A. behavior based control system for assistive bimanual manipulation // Recent Progress in Robotics: Viable Robotic Service to Human. Berlin, Germany: Springer, 2007. P. 345–355.

30. Makris S., Tsarouchi P., Surdilovic D., Krüger J. Intuitive dual arm robot programming for assembly operations // CIRP Ann., 2014. Vol. 63, no. 1. P. 13–16.

31. Smithetal C. Dual arm manipulation—survey // Robot.Auton.Syst., 2012. Vol. 60, no. 10. P. 1340–1353.

32. Rakita D., Mutlu B., Gleicher M., Hiatt L.M. Shared control-based bimanual robot manipulation // Sci. Robot., 2019. Vol. 4, Art. no. eaaw0955.

33. Sun D., Liao Q., Loutfi A. Single master bimanual teleoperation system with efficient regulation // IEEE Trans. Robot., 2020. Vol. 36, no. 4. P. 1022–1037.

34. *Lin T.-C., Unni Krishnan A., Li Z.* Shared autonomous interface for reducing physical effort in robot teleoperation via human motion mapping // Proc. IEEE Int. Conf. Robot. Automat. (ICRA), 2020. P. 9157–9163.

35. *Amanhoud W., Hernandez Sanchez J., Bouri M., Billard A.* Contact-initiated shared control strategies for four-arm supernumerary manipulation with foot interfaces // Int. J. Robot. Res., 2021. Vol. 40, P. 1–29.

36. *Laghi M. et al.* Shared-autonomy control for intuitive bimanual telemanipulation // Proc. IEEE-RAS 18th Int. Conf. Humanoid Robots (Humanoids), 2018. P. 1–9.

37. *Tyrva V.O.* Sovmestnoe upravlenie ob"ektom v ergaticheskoy sisteme: modeli i realizacii // Vestnik Gosudarstvennogo universiteta morskogo i rechnogo flota imeni admirala S.O. Makarova. 2018. T. 10. No. 2. S. 430–443.

38. *Tung A. et al.* Learning multi-arm manipulation through collaborative teleoperation // Proc. IEEE Int. Conf. Robot. Automat. (ICRA), 2021. P. 9212–9219.

39. *Kennel-Maushart F., Poranne R., Coros S.* Manipulability optimization for multi-arm teleoperation // Proc. IEEE Int. Conf. Robot. Automat. (ICRA), 2021. P. 3956–3962.

40. *Ozdamar I., Laghi M., Grioli G., Ajoudani A., Bicchi A.* A Shared Autonomy Reconfigurable Control Framework for Telemanipulation of Multi-Arm Systems // IEEE ROBOTICS AND AUTOMATION LETTERS, 2022. Vol. 7, No. 4. P. 9937–9944.

41. *Laghi M. et al.* Shared-autonomy control for intuitive bimanual tele-manipulation // Proc. IEEE-RAS 18th Int. Conf. Humanoid Robots (Humanoids), 2018. P. 1–9.

42. *Ashkinazi L.A.* Mir Lema – slovar' i putevoditel'. 2004. 1068 s.

43. *Ozdamar I., Laghi M., Grioli A., Ajoudani G., Catalano M.G., Bicchi A.* A Shared Autonomy Reconfigurable Control Framework for Telemanipulation of Multi-Arm Systems // IEEE Robotics and Automation Letters, 2022. Vol.7(4). P. 9937–9944.

44. *Florea B.C.* Smartphone Controlled Autonomous Robotic Platform // 2018 15th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), Chiang Rai, Thailand, 2018. P. 620–623.

45. *Mustafin M., Tsoy T., Martínez-García E.A., Meshcheryakov R., Magid E.* Modelling mobile robot navigation in 3D environments: camera-based stairs recognition in



Gazebo // 2022 Moscow Workshop on Electronic and Networking Technologies (MWENT), Moscow, Russian Federation, 2022. P. 1–6.

46. *Bai Y. , Wang Y., Svinin M., Magid E., Sun R.* Adaptive Multi-Agent Coverage Control With Obstacle Avoidance // *IEEE Control Systems Letters*, 2022. Vol. 6. P. 944–949.

47. *Kugurakova V.V., Antonov I.O., Goncharenko B.V., Chajbar A.A.* Cifrovoe predstavlenie v virtual'noj real'nosti mesta proisshestviya kak instrument ugovnogo sudoproizvodstva // *Programmnye sistemy: teoriya i prilozheniya*, 2022. T. 13, vyp. 3. S. 193–223.

48. *Kugurakova, V.V., Golovanova, I.I., Shaidullina, A.R.* Digital Solutions in Educators' Training: Concept for Implementing a Virtual Reality Simulator // *Eurasia Journal of Mathematics, Science and Technology Education*. 2021. Vol. 17, Is. 9. P. 1–10.

49. *Tianhao Zhang, Zoe McCarthy, Owen Jow Dennis, Lee Xi Chen, Ken Goldberg, Pieter Abbeel.* Deep Imitation Learning for Complex Manipulation Tasks from Virtual Reality Teleoperation // [eecs.berkeley.edu](https://eecs.berkeley.edu).

URL: <https://www2.eecs.berkeley.edu/Pubs/TechRpts/2020/EECS-2020-190.pdf> (access date: 01.08.2022).

50. IVRE – An Immersive Virtual Robotics Environment // [cirl.lcsr.jhu.edu](http://cirl.lcsr.jhu.edu).  
URL: <https://cirl.lcsr.jhu.edu/research/human-machine-collaborativesystems/ivre/> (access date: 01.08.2022).

51. Reachy by Pollen Robotics, an open source programmable humanoid robot // [pollen-robotics.com](http://pollen-robotics.com). URL: <https://www.pollen-robotics.com/> (access date: 01.08.2022).

52. *Kugurakova V.V., Hafizov M.R., Kadyrov S.A., Zykov E.Yu.* Udalennoe upravlenie robotizirovannym ustrojstvom s ispol'zovaniem tekhnologij virtual'noj real'nosti // *Programmnye produkty i sistemy*. 2022. T. 35. No.3. S. 348–361.

53. *Melero M., Hou A., Cheng E., Tayade A., Lee S.C., Unberath M., Navab N., Cutolo F.* Upbeat: Augmented Reality-Guided Dancing for Prosthetic Rehabilitation of Upper Limb Amputees // *Journal of Healthcare Engineering*. 2019. Art. 2163705.

54. *Nissler C., Nowak M., Connan M., Büttner S., Vogel J., Kossyk I., Márton Z.C., Castellini C.* VITA – An everyday virtual reality setup for prosthetics and upper-limb rehabilitation // *Journal of Neural Engineering*. 2019. Vol. 16(2). Art. 026039.

55. Saraiji M.Y., Sasaki T., Matsumura R., Minamizawa K., Inami M. Fusion: full body surrogacy for collaborative communication // SIGGRAPH '18: ACM SIGGRAPH 2018 Emerging Technologies. 2018. Vol. 7. P. 1–2.

56. Takizawa R., Hivarna A., Verhulst A., Seaborn K., Fukuoka M., Kitazaki M., Inami M., Sugimoto M. Parasitic body: Exploring perspective dependency in a shared body with a third arm // 26th IEEE Conference on Virtual Reality and 3D User Interfaces. VR 2019 – Proceedings. 2019. P. 1175–1176.

57. Parietti F., Asada H. Supernumerary Robotic Limbs for Human Body Support // IEEE Transactions on Robotics. 2016. Vol. 32(2). P. 301–311.

58. Sasaki T., Saraiji Y., Minamizawa K., Inami M. MetaArmS: Body remapping using feet-controlled artificial arms // UIST 2018 Adjunct – Adjunct Publication of the 31st Annual ACM Symposium on User Interface Software and Technology. 2018. P. 140–142.

59. Tong Y., Liu J. Review of Research and Development of Supernumerary Robotic Limbs // IEEE/CAA Journal of Automatica Sinica. 2021. Vol. 8. No. 5. P. 929–952.

60. Aly H.I., Youssef S., Fathy C. Hybrid Brain Computer Interface for Movement Control of Upper Limb Prostheses. 2018 International Conference on Biomedical Engineering Applications // ICBEA 2018 – Proceedings. 2018. P. 1–6.

61. Aljalal M., Ibrahim S., Djemal R., Ko W. Comprehensive review on brain-controlled mobile robots and robotic arms based on electroencephalography signals // Intelligent Service Robotics. 2020. Vol. 13 (4). P. 539–563.

62. Abdi E., Burdet E., Bouri M., Himidan S., Bleuler H., In a demanding task three-handed manipulation is preferred to two-handed manipulation // Scientific Reports. 2016. Vol. 6. P. 1–11.

63. Sasaki T., Saraiji Y., Minamizawa K., Inami M. MetaArmS: Body remapping using feet-controlled artificial arms // UIST 2018 Adjunct - Adjunct Publication of the 31st Annual ACM Symposium on User Interface Software and Technology. 2018. P. 140–142.

64. Drogemuller A., Verhulst A., Volmer B., Thomas B.H., Inami M., Sugimoto M. Remapping a Third Arm in Virtual Reality // 2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR), Osaka, Japan, 2019. P. 898–899.

65. Virtual'nyj trenazher upravleniya prizrachnymi konechnostyami cherez interfejs mozg-komp'yuter : Svidetel'stvo o gosudarstvennoj registracii programmy dlya EVM № 202211167 Rossijskaya Federaciya / V.V. Kugurakova, D.G. Ismailov, D.M. Timergalin;

заявитель' i правообладатель' Fed. gos. avtonom. obrazovat. uchrezhdenie vyssh. obraz. Kazanskij fed. un-t. №2022610311; zayavl. 13.01.2022; zaregistrovano v reestre programm dlya EVM 20.01.2022. – [1] s.

66. *Zhil'cov A.I., Zhukov V.S., Ryleev D.A.* Upravlenie manipulyatorami s chislom stepenej svobody bolee shesti // Inzhenernyj zhurnal: nauka i innovacii, 2013, vyp. 10. 11 s.

67. *Kugurakova V.V., Hafizov M.R., Kadyrov S.A., Zykov E.Yu.* Udalennoe upravlenie robotizirovannym ustrojstvom s ispol'zovaniem tekhnologij virtual'noj real'nosti // Programmnye produkty i sistemy. 2022. T. 35. No. 3. S. 348–361.

68. *Ushakova M.V., Ushakov Yu.A.* Issledovanie energosberegayushchej besprovodnoj samoorganizuyushchejsya mnogoprotokol'noj seti peredachi dannyh internet ustrojstv // Sovremennye informacionnye tekhnologii i IT-obrazovanie. 2019. T. 15, No. 3. S. 733–745.

69. *Miroshnikova N.E.* Obzor sistem kognitivnogo radio // T-Comm – Telecommunications and Transport magazine. 2013. No. 9. S. 108–111.

---

## СВЕДЕНИЯ ОБ АВТОРАХ



**КУГУРАКОВА Влада Владимировна** – к. т. н., доцент кафедры программной инженерии Института ИТИС КФУ, руководитель НИЛ разработки интеллектуальных инструментов для компьютерных игр. Сфера научных интересов – иммерсивность виртуальных сред, различные аспекты проектирования игр, AR/VR.

**Vlada Vladimirovna KUGURAKOVA**, PhD., Docent of the Institute ITIS KFU, Head of laboratory of intelligent tools design for computer games development. Research interests include immersiveness of virtual environments, problems of generating realistic visualization, various aspects of game design, AR/VR.

email: vlada.kugurakova@gmail.com

ORCID: 0000-0002-1552-4910

---



**СЕРГУНИН Игорь Дмитриевич** – технический Директор АО «Аринет Спейс». Сфера научных интересов – робототехника, сетевые технологии, ИИ, системы телеприсутствия, системы беспроводной передачи данных, системы дистанционного управления сложными манипуляторами.

**Igor Dmitrievich SERGUNIN** – Technical Director of Arinet Space JSC. Research interests - robotics, network technologies, AI, telepresence systems, wireless data transmission systems, remote control systems for complex manipulators.

email: [sergunin.igor@arinet.space](mailto:sergunin.igor@arinet.space)

ORCID: 0000-0003-4025-0023



**ЗЫКОВ Евгений Юрьевич** – доцент, к. ф.-м. н., руководитель НИЛ «Технологии расширенной реальности в промышленных процессах» («Приоритет 2030»), доцент кафедры радиоастрономии Высшей школы киберфизических систем и прикладной электроники Института физики Казанского федерального университета. Область научных интересов: радиофизика, математическое моделирование и численные методы, математическое и программное обеспечение вычислительных машин.

**Evgeniy Yurevich ZYKOV** – Docent, Candidate of Physical and Mathematical Sciences, Head of the Research Laboratory of Augmented Reality Technology in Industrial Processes («Priority 2030»), Associate Professor of the Department of Radio Astronomy of the Higher School of Cyber-Physical Systems and Applied Electronics of the Institute of Physics of Kazan Federal University. Research interests: radiophysics, mathematical modeling and numerical methods, mathematical and computer software.

email: [Evgeniy.Zykov@kpfu.ru](mailto:Evgeniy.Zykov@kpfu.ru)

ORCID: 0000-0002-3014-2507

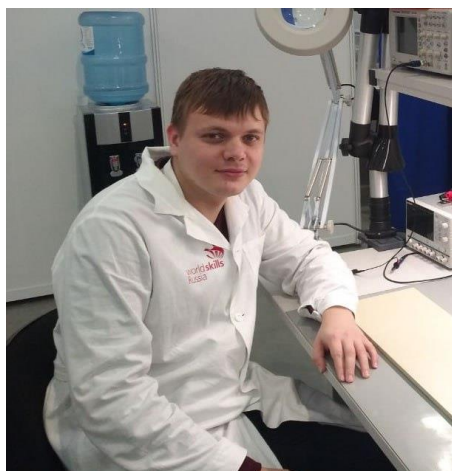


**СЕРГУНИН Олег Дмитриевич** – Генеральный Директор АО «Аринет Спейс». Сфера научных интересов – робототехника, сетевые технологии, ИИ, системы телеприсутствия, системы беспроводной передачи данных, системы дистанционного управления сложными манипуляторами.

**Oleg Dmitrievich SERGUNIN** – General Director of Arinet Space JSC. Research interests are robotics, network technologies, AI, telepresence systems, wireless data transmission systems, remote control systems for complex manipulators.

email: [sergunin.oleg@arinet.space](mailto:sergunin.oleg@arinet.space)

ORCID: 0000-0002-8134-589X



**УЛАНОВ Алексей Валерьевич** – Главный конструктор АО «Аринет Спейс». Сфера научных интересов – робототехника, цифровая обработка сигналов, инфокоммуникационные технологии и системы связи, технологии программно-конфигурируемого радио.

**Alexey Valerievich ULANOV** – Chief designer of JSC «Arintet Space». Research interests - Robotics, digital signal processing, infocommunication technologies and communication systems, software defined radio technologies.

email: [ulanov.alexey@arinet.space](mailto:ulanov.alexey@arinet.space)

ORCID: 0000-0002-7336-5276



**ГАБДУЛЛИНА Динара Рустамовна** – лаборант, НИЛ «Технологии расширенной реальности в промышленных процессах» Института информационных технологий и интеллектуальных систем Казанского федерального университета («Приоритет 2030»). Область научных интересов: технологии дополненной и виртуальной реальности, компьютерное зрение.

**Dinara Rustamovna GABDULLINA** – laboratory assistant, Research Laboratory of Augmented Reality Technologies in Industrial Processes of the Institute of Information Technologies and Intelligent Systems of Kazan Federal University («Priority 2030»). Research interests: augmented and virtual reality technologies, computer vision.

email: dinaragabdullina086@gmail.com

ORCID: 0000-0002-4877-924X



**ГИЛЕМЯНОВ Артем Шамилевич** – Руководитель отдела робототехники АО «Аринет Спейс». Сфера научных интересов – робототехника, системы управления удаленными устройствами.

**Artem Shamilevich GILEMYANOV** – Head of the Robotics Department of Arinet Space JSC. Research interests - Robotics, remote device control systems.

email: gilemyanov.artem@arinet.space

ORCID: 0000-0001-8237-7589

*Материал поступил в редакцию 12 сентября 2022 года*

## **СРАВНЕНИЕ КЛИЕНТ-СЕРВЕРНЫХ РЕШЕНИЙ ПРИ РАЗРАБОТКЕ МНОГОПОЛЬЗОВАТЕЛЬСКИХ ОНЛАЙН-ИГР НА UNITY**

**И. Р. Мухаметханов**<sup>1</sup> [0000-0002-0562-1434], **М. Р. Хафизов**<sup>2</sup> [0000-0001-7275-9102],

**А. В. Шубин**<sup>3</sup> [0000-0002-6203-3268]

<sup>1,2,3</sup>*Институт информационных технологий и интеллектуальных систем Казанского (Приволжского) федерального университета;*

<sup>1</sup>ilnurhamay@mail.ru, <sup>2</sup>mrimsh@mrimsh.ru, <sup>3</sup>shubin.aleksey.kpfu@gmail.com

### ***Аннотация***

В работе представлена критика традиционного подхода, используемого для создания многопользовательской игры в системе разработки интерактивных приложений в реальном времени Unity, особенно в случае большого числа одновременных пользователей. В качестве гипотезы предложен альтернативный вариант, не являющийся распространённым, но решающий многие проблемы предыдущего подхода. Проведено сравнение двух клиент-серверных решений при разработке в Unity многопользовательских онлайн-игр, также описаны преимущества обоих подходов для разных случаев. Предложена архитектура разработки игры при помощи более актуального метода: вместо библиотеки Mirror – стандартного инструментария для Unity-разработки – использованы микросервисы, написанные на языке Golang. Приведены весомые доказательства предпочтительности альтернативного подхода, главное преимущество которого – поддержка современной архитектуры, обеспечивающей высокоскоростную связь между микросервисами, что подкреплено тестами при передаче сообщений на разных платформах.

Полученные результаты тестирования подтверждают выдвинутую гипотезу, и можно сделать вывод, что для многопользовательских видеоигр связка Unity вместе с Golang является более эффективной.

Описаны также основные методы отладки многопоточного приложения на Golang в связке с системой разработки игровых приложений Unity и предложен

технологический прием, позволяющий получить быстрый способ передачи данных между клиентом и сервером.

**Ключевые слова:** игровой движок, микросервисы, Unity, Golang, Mirror, видеоигра, мультиплеер.

## ВВЕДЕНИЕ

Играя с другими людьми, игрок получает новые опыт и эмоции. Поэтому для реализации видеоигр очень важно использовать подходящие клиент-серверные решения, тем самым обеспечив игроку комфорт в плане корректной и бесперебойной работы для заданного количества пользователей.

Отсутствие ошибок и корректная работа – одни из важнейших аспектов игры, позволяющие удержать игрока на более длительный промежуток времени, а обеспечивает данные аспекты именно грамотный выбор используемого решения. Обсудим, какие инструменты для этого используют при разработке и в чем их принципиальные различия.

Unity – система разработки интерактивных приложений, называемая среди разработчиков игровым движком<sup>1</sup>, является бесплатным, одним из наиболее популярных вариантов для разработки видеоигр, имеющих низкий порог входа, что позволяет начать разработку в кратчайшие сроки.

При создании мультиплеерной<sup>2</sup> видеоигры разработчики имеют довольно широкий выбор сервисов для реализации. Довольно популярным среди других разработчиков [1–3] является использование Photon Unity Networking (PUN)<sup>3</sup>. Данный инструмент при всём его удобстве и популярности имеет большой недостаток – поддержка до 20 пользователей, за большее число пользователей разработчик обязан заплатить за аренду мест, так как все приложения, применяющие технологию PUN, запускаются на серверах Photon Cloud.

---

<sup>1</sup> Игровой движок (от англ. game engine) – комплексное ПО для разработки и поддержки игровых приложений.

<sup>2</sup> Мультиплеер (от англ. multiplayer) – режим компьютерной игры, когда совместно играет больше одного человека.

<sup>3</sup> Photon Unity Networking (PUN) или Photon Engine – популярный фреймворк для разработки многопользовательских игр. URL: <https://www.photonengine.com/en-US/Photon>.



Просмотрев другие варианты реализации мультимедиа, мы решили сравнить пару вариантов решений: (1) традиционное решение от Unity в виде библиотеки Mirror<sup>4</sup>, использующей архитектуру REST<sup>5</sup>, и (2) приложение на языке Golang<sup>6</sup> (или Go), реализуемое через микросервисы<sup>7</sup>.

Mirror – это высокоуровневая сетевая библиотека для Unity, совместимая с различными транспортом низкого уровня. Она используется для создания как клиентского, так и серверного приложений и поддерживает архитектуру REST для оптимизации приложений и транспортировки данных. REST – это набор правил для разработки веб-приложений. Приложение является RESTful, если удовлетворяет набору условий.

Golang – компилируемый язык программирования с открытым исходным кодом, ориентированный на избавление от избыточных данных и скорость работы кода, в том числе при помощи многопоточных вычислений. С момента своего первого выпуска в 2009 году язык программирования Golang [4] был хорошо принят сообществами разработчиков программного обеспечения. Основной причиной его успеха стала мощная поддержка разработки на основе библиотек, когда проект, реализуемый на Golang, может быть удобно построен поверх других проектов путем ссылки на них как на библиотеки, что позволяет проекту импортировать и повторно использовать функциональные возможности из другого проекта Golang, просто указав путь импорта.

### **ДОСТОИНСТВА И НЕДОСТАТКИ ВЫБРАННЫХ ВАРИАНТОВ РЕАЛИЗАЦИИ**

По нашему мнению, одна из важнейших проблем Unity и решений, реализованных при помощи этого игрового движка, это отсутствие безопасности отдельных потоков, заключающееся в невозможности проводить сверхсложные вычисле-

---

<sup>4</sup> Mirror, <https://mirror-networking.gitbook.io/docs/general/ccu>.

<sup>5</sup> REST (от англ. Representational State Transfer – «передача репрезентативного состояния» или «передача „самоописываемого“ состояния») — архитектурный стиль взаимодействия компонентов распределённого приложения в сети.

<sup>6</sup> Golang – The Go Programming Language, <https://go.dev>

<sup>7</sup> Микросервис (от англ. microservices) – вариант архитектуры программного обеспечения, состоящей из взаимозаменяемых и независимых модулей.

---

ния в другом потоке – ведь, чтобы применять результаты таких вычислений на игровой сцене<sup>8</sup>, необходимо вернуться в основной поток. При этом Golang имеет большой потенциал для разработки в многопоточном режиме и, соответственно, может задействовать больше мощностей сервера.

Вторая проблема состоит в том, что в Unity присутствует ограничение на количество игроков на одном сервере в 1500 пользователей.

Наконец, ещё одна проблема состоит в том, что библиотека Mirror, традиционно используемая для Unity, имеет недостаток в виде устаревшей системы удалённого вызова процедур (RPC<sup>9</sup>), разработка и практическая имплементация которой были начаты ещё в 1970-х годах.

В свою очередь Golang славится поддержкой gRPC – это архитектура RPC с открытым исходным кодом, разработанная Google для обеспечения *высокоскоростной* связи между микросервисами. Она позволяет интегрировать сервисы, написанные на разных языках, используя высокоэффективный формат обмена сообщениями через буферы протокола.

Существует много популярной литературы и научных исследований, посвящённых сравнению производительности веб-серверов, но редко обсуждается их сочетание, хотя Unity и Golang – популярные платформы, которые широко используются в качестве бэкендов веб- и мобильных приложений.

Итак, оценим реализацию мультиплеера на Unity при помощи как традиционного решения в виде библиотеки Mirror, так и через реализацию микросервисов на языке Golang. Выделим конкретные достоинства и недостатки каждого метода – для определения более подходящего в случае разработки многопользовательской видеоигры.

---

<sup>8</sup> Игровая сцена – общеупотребительный термин в разработке игр и приложений компьютерной графики – внутриигровое пространство с 3D и 2D объектами и другими данными, структурированными в определенной иерархии, которые необходимы для организации взаимодействий игрового процесса.

<sup>9</sup> RPC (от англ. remote procedure call) – Удалённый вызов процедур.

---

## Характеристики gRPC

### *Достоинства gRPC:*

1. Использование http/2 в качестве транспортного протокола:
  - асинхронная обработка больших наборов данных;
  - поддержка отправки нескольких запросов по одному соединению;
  - двунаправленное взаимодействие для одновременной отправки запросов от клиента и получения ответа от сервера;
  - сжатие заголовков (уменьшение загруженности сети).
2. Применение Protobuf<sup>10</sup> (вместо стандартных JSON или XML) для обработки данных:
  - сериализация сообщений в строго типизированном двоичном формате, что может легко восприниматься любым языком программирования;
  - более сжатый формат сообщений.
3. Встроенная генерация кода для создания запросов, что позволяет отказаться от стороннего инструментария.
4. Встроенный компилятор протоколов.
5. Поддержка нескольких языков программирования и отсутствие зависимости от платформы.

### *Недостатки gRPC:*

1. Отсутствие обратной совместимости с протоколом http 1.1, что ограничивает применение метода.
2. Использование Protobuf не обеспечивает удобочитаемости данных для программиста.
3. На данный момент, является менее популярным методом, что влечёт за собой отсутствие прямой совместимости с некоторыми инструментами.

Платформа gRPC была создана для ускорения передачи данных между микросервисами и другими системами, которым необходимо взаимодействовать друг с другом. Несмотря на то, что gRPC основывается на RPC, он имеет ряд существенных отличий от своего предшественника.

---

<sup>10</sup> Protobuf – технология упаковки данных от Google. Буферы протокола – это легкий и эффективный формат хранения структурированных данных, который можно использовать для сериализации или сериализации структурированных данных.

---

## Характеристики RPC

Основными положительными чертами RPC можно назвать отсутствие недостатков, названных для gRPC, а также:

1. Использование универсальных подходов и методов.
2. Высокая скорость имплементации решения.

Также данная технология не обходится и без недостатков. Недостатки RPC:

1. Необходимость в использовании стороннего инструментария для генерации кода, как, например, Swagger<sup>11</sup> или Postman<sup>12</sup>.
2. Отсутствие стандартной методологии – имеются лишь требования к архитектуре, при этом реализация может отличаться.
3. Заметно низкая скорость передачи данных, связанная с использованием нежатых форматов файлов и устаревших протоколов передачи данных.

Учитывая все достоинства и недостатки, можно сказать, что использование платформы gRPC в связке с Unity имеет больше плюсов, чем с её аналогом REST. Наиболее весомым отличием этих двух связок является ускоренная передача сообщений – по тестам Рувана Фернандо [5], при передаче сообщений на разных платформах соединения gRPC API значительно быстрее, чем соединения REST API, фактически в 7–10 раз (см. рис. 1). Этот факт, вместе с поддерживаемым многопоточным сообщением с несколькими клиентами является критически важным для массовой многопользовательской видеоигры, где количество игроков может достигать нескольких тысяч одновременно. Кроме этого, от скорости и точности обмена данными будет влиять, насколько интересным будет опыт игры.

REST API идеально подходит, когда системе требуются высокоскоростная итерация и стандартизация протокола HTTP. Благодаря универсальной поддержке сторонних инструментов, REST API должен быть первым аргументом в пользу интеграции приложений, интеграции микросервисов и разработки веб-сервисов – но не видеоигры.

Что касается gRPC, то большинство инструментов сторонних производителей до сих пор не имеют встроенных функций для совместимости с ним.

---

<sup>11</sup> Swagger, <https://swagger.io>

<sup>12</sup> Postman, <https://www.postman.com>

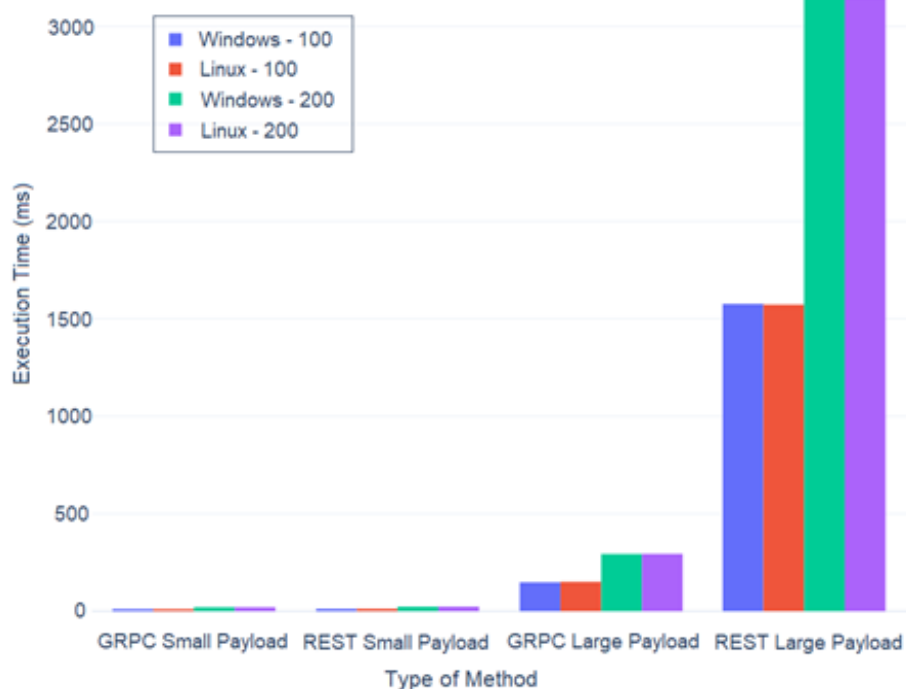


Рис. 1. Сравнение производительности REST и gRPC

Значит, gRPC используется в основном для построения внутренних систем, то есть инфраструктур, закрытых для внешних пользователей. Учитывая эту оговорку, API gRPC может быть полезен в следующих случаях:

- *Соединения легковесных микросервисов* – особенно там, где эффективность передачи сообщений имеет первостепенное значение.
- *Системы, где используется несколько языков программирования*, – благодаря поддержке генерации собственного кода для широкого спектра языков разработки.
- *Потоковая передача в реальном времени* – порождается способностью gRPC управлять двунаправленным потоком, не дожидаясь ответа от клиента.
- *Сети с низким энергопотреблением и низкой пропускной способностью* – использование сериализованных сообщений Protobuf обеспечивает простоту обмена сообщениями, большую эффективность и скорость для сетей с ограниченной пропускной способностью и низким энергопотреблением [6].

## Контейнерная платформа Docker

Также микросервисы, написанные на Golang, очень удобно развёртывать за счёт Docker – платформы для разработки, доставки и запуска контейнерных приложений и управления их жизненным циклом. Docker позволяет запускать множество контейнеров на одной хост-машине<sup>13</sup>. Контейнеры в целом упрощают работу как программистам, так и администраторам, которые разворачивают эти приложения, и позволяют упаковать в единый образ приложение и все его зависимости: библиотеки, системные утилиты и файлы настройки. Это упрощает перенос приложения на другую инфраструктуру. Контейнер – это набор процессов, изолированных от основной операционной системы. Приложения работают только внутри контейнеров и не имеют доступа к основной операционной системе (ОС). Это повышает безопасность приложений, потому что они не смогут случайно или умышленно навредить основной системе. Если приложение в контейнере завершится с ошибкой или зависнет, это никак не затронет основную ОС. Контейнеры хорошо вписываются в микросервисную архитектуру. Это подход к разработке, при котором приложение разбивается на небольшие компоненты, по возможности независимые. Обычно такой подход противопоставляется монолитной архитектуре, где все части системы сильно связаны друг с другом. Это позволяет разрабатывать новую функциональность быстрее, ведь в случае с монолитной архитектурой изменение какой-то части может затронуть всю остальную систему.

## ОБЗОР МЕТОДОВ РАЗРАБОТКИ ВИДЕОИГР

Итак, выше были рассмотрены как достоинства, так и недостатки разных подходов к обеспечению многопоточности в многопользовательских видеоиграх. Связка Unity и Golang представляется удачной для видеоигр, в которых происходит передача данных в реальном времени. Это видеоигры в жанрах ММО<sup>14</sup>,

---

<sup>13</sup> Хост-машина (от англ. host – «владелец, принимающий гостей») – любое устройство, предоставляющее сервисы формата «клиент-сервер» в режиме сервера по каким-либо интерфейсам и уникально определённое на этих интерфейсах. В более широком смысле под хостом могут понимать любой компьютер, подключённый к локальной или глобальной сети.

<sup>14</sup> ММО (сокр. от англ. Massively Multiplayer Online Game) – массовая многопользовательская онлайн игра.

MMORPG<sup>15</sup>, Shooter<sup>16</sup> – игры, в которых происходит очень много обмена данными, где даже минимальная задержка может помешать наслаждаться геймплеем<sup>17</sup>. А также «большие» игры, в которые одновременно играет много игроков, – ведь, как и в случае с очень активными играми, будет очень большая нагрузка на сервер.

Приведем пример архитектуры такой видеоигры, основанной на микросервисном подходе. Микросервисы могут хранить какие-то данные о пользователе. Если эти данные изменяются, микросервис подключается к базе данных, в которой будут храниться эти данные. Также среди микросервисов присутствует файл *go.mod*, где хранятся все файлы репозитория, из которых будет использован программный код. Но самое интересное в использовании микросервисного подхода – это proto-файлы, где хранятся grpc-методы и структуры данных, которые они получают на вход или выдают на выходе. Из этих proto-файлов компилируется код *pb.go*, который содержит весь код Protobuf для заполнения, сериализации и извлечения типов сообщений запроса и ответа. Микросервис *grpc.pb.go* содержит

- тип интерфейса (или заглушка) для вызовов клиентов с помощью определенных в нём методов,
- тип интерфейса для реализации серверами также с определенными в нём методами.

Само Unity-приложение с контентом видеоигры содержит пользовательский интерфейс, через который и отправляются запросы на сервер. Сервер их обрабатывает и отправляет ответ. Пример архитектуры Unity-игры, использующей gRPC, приведён на рис. 2.

---

<sup>15</sup> MMORPG (сокр. от англ. Massively Multiplayer Online Role-Playing Game) – массовая многопользовательская ролевая игра.

<sup>16</sup> Shooter (англ. стрелок) – игровой жанр, достижение цели которого преимущественно заключается в ликвидации противника при помощи огнестрельного оружия.

<sup>17</sup> Геймплей (от англ. gameplay) – игровой процесс, который образуется благодаря действиям игрока.

---

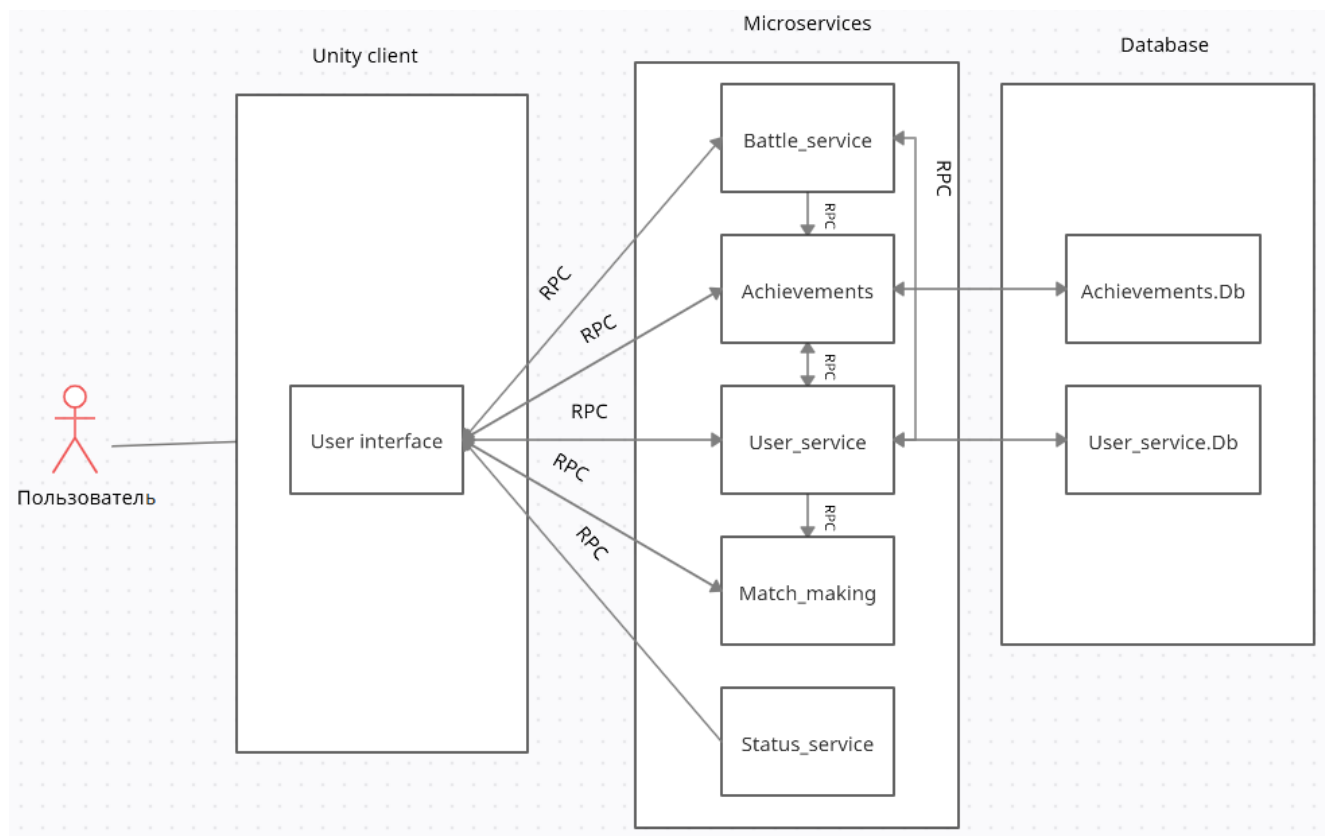


Рис. 2. Архитектура игры

Опишем, как происходит процесс создания микросервисов. Сначала создаётся proto-файл, в котором определены методы и сообщения, в частности, как на примере ниже (см. рис. 3), создаётся grpc-метод *GetUserProfile*, который принимает запрос *QueryUserProfile* и возвращает ответ *UserProfile*.

Далее на сервере создаётся метод *GetUserProfile*, который принимает в себя *QueryUserProfile* и возвращает *GetUserProfile* на основе proto. Unity же вызывает этот метод через код-прослойку на клиенте.

Использование докера позволяет легко собрать микросервисы и, в случае необходимости их изменения, легко сделать это, не задевая работу остальных компонентов системы. Для каждого микросервиса нужны свои proto-файлы, а для Unity можно создать отдельный proto-файл для удобства. Для того чтобы это всё работало, необходимо для каждого метода, в котором есть общение клиента с сервером, создать свой grpc-метод.



Докер – это тот технологический приём, который позволяет реализовать не только эффективный и быстрый, но и безопасный способ передачи данных между клиентом и сервером.

```
service UserService {
    rpc Auth(AuthRequest) returns (Session);
    rpc Register(RegisterRequest) returns (Session);
    rpc ChangeUsername(CommandChangeUsername) returns (common.Empty);
    rpc ChangeAvatar(CommandChangeAvatar) returns (common.Empty);
    rpc ChangeTable(CommandChangeTable) returns (common.Empty);
    rpc ChangeHandColor(CommandChangeColor) returns (common.Empty);
    rpc ChangeHandJewelry(CommandChangeJewelries) returns (common.Empty);
    rpc GetUserEmojies(QueryUserEmojies) returns (ProfileItemResult);
    rpc GetAvatars(QueryAvatars) returns (ProfileItemResult);
    rpc GetUserInfo(QueryUserInfo) returns (UserInfo);
    rpc GetUserProfile(QueryUserProfile) returns (UserProfile);
    rpc GetUserUpdates(QueryUserUpdates) returns (UserUpdatesResult);
    rpc GetLeaderboard(QueryLeaderboard) returns (LeaderboardResult);
    rpc GetUserLegendaryTrackReward(CommandGetUserLegendaryTrackReward) returns (RewardResult);
    rpc GetLegendaryTrack (QueryLegendaryTrack) returns (LegendaryTrackResult);
    rpc GetOpenUserAchievements (QueryUserAchievements) returns (AchievementsResult);
    rpc GetAllUserAchievements (QueryUserAchievements) returns (AchievementsResult);
    rpc GetUserAchievementReward (CommandGetUserAchievementReward) returns (UserAchievementReward);
    rpc GetUserDailyTasks (QueryUserDailyTasks) returns (DailyTasksResult);
    rpc GetUserDailyReward (CommandGetUserDailyReward) returns (DailyReward);
    rpc MarkEmojiAsActive (MarkEmojiAsActiveRequest) returns (common.Empty);
    rpc UnmarkEmojiAsActive (UnmarkEmojiAsActiveRequest) returns (common.Empty);

    rpc CleanCardsAndTutorial (common.Empty) returns (common.Empty);
    rpc GetTutorialState(QueryUserStatus) returns (StatusUserResult);
    rpc SetTutorialState(CommandSetStatus) returns (shop.PurchaseProductResult);
}
```

Рис. 3. Пример proto-файла

### Нюансы отладки многопоточных программ

Стоит отметить, что Go, являющийся многофункциональным языком программирования, требует эффективных методов отладки параллелизма. Но использование комбинированного инструмента статического и динамического тестирования и анализа параллелизма значительно облегчает процесс отладки реальных программ, предлагая автоматизированную динамическую трассировку для фикса-

ции поведения примитивов параллелизма, систематическое исследование пространства расписаний для ускорения поиска ошибок и обнаружения тупиков с дополнительными визуализациями и отчетами, набор требований к покрытию, характеризующий динамическое поведение примитивов параллелизма и метрики для измерения качества тестов. Эти подходы, безусловно, эффективны в обнаружении редких ошибок, а метод возмущения расписания позволяет быстро их обнаружить, делая «полевую» отладку программ на Go значительно проще.

Мы не рассматривали работу анализируемых продуктов с СУБД<sup>18</sup>, так как скорость работы с ней зависит непосредственно от самой СУБД. Однако нужно отметить, что в [7] проведено сравнение производительности Golang и node.js в качестве бэкенда веб-приложений в сцепке с MySQL и MongoDB в качестве баз данных и сделаны выводы о том, что комбинация Go+MySQL превосходит по использованию памяти и процессора, а Node.js+MySQL превосходит по времени отклика.

### ОБСУЖДЕНИЕ РЕЗУЛЬТАТОВ

Можно сделать вывод, что использование gRPC-архитектуры имеет больше плюсов, чем использование стандартного подхода REST. В частности, gRPC имеет передачу сообщений быстрее в 7–10 раз, чем REST. Но для утверждения выставленной гипотезы о большей эффективности предложенной связки необходимо, кроме *времени отклика*, оценить такие характеристики производительности, как *загрузка процессора* и *использование памяти*. Так как в качестве буфера протокола используется Protobuf, то размер данных сокращается – этим подтверждается меньшее использование памяти.

Многопоточность, используемая gRPC, ускоряет передачу сообщений, делает её более эффективной, но не подтверждает меньшую загрузку процессора, что ограничивает применение связки Unity+gRPC для тех многопользовательских видеоигр, которые слишком требовательны к характеристикам сервера.

Итак, Golang при производстве многопользовательских игр на Unity показывает лучшие результаты, чем стандартные решения.

---

<sup>18</sup> СУБД (аббрев.) – система управления базой данных.

## **ЗАКЛЮЧЕНИЕ**

Индустрия разработки видеоигр, или gamedev, на данный момент имеет малую степень систематизации знаний и пока находится вне фундаментальных исследований. Требуется создать научный базис, основываясь на котором, можно будет формировать гипотезы, теории, методы, в том числе, для повышения качества разработки видеоигр, для уменьшения переработок специалистов, для смещения баланса разработки видеоигр с практико-ориентированной к научно-ориентированной.

На данный момент фокусом научных интересов лаборатории разработки игр, визуализации, дополненной/виртуальной реальности (Digital Media Lab) ИТИС<sup>19</sup> КФУ<sup>20</sup> являются: нейронные сети в разработке видеоигр (см., например, [8, 9]); автоматизация и абстрагирование разработки видеоигр, включая поиск оптимальных подходов задачи автоматизации генерации прототипов видеоигр (см., например, [10]); систематизация и поиск наилучших методов вовлечения и погружения игроков видеоигр. Практические интересы включают применение современных технологий и знаний в области создания видеоигр.

В статье сравнены современные подходы к реализации многопользовательских видеоигр и обсуждена эффективность технологических приемов, позволяющих получать быстрые способы передачи данных между клиентом и сервером. Также приведены основные методы для отладки многопоточных приложений на Golang в качестве бэкенда игрового приложения в связке с системой разработки игровых приложений Unity.

Данная статья является одним из этапов к пониманию процессов эффективного создания видеоигр.

## **БЛАГОДАРНОСТИ**

Работа выполнена за счет средств Программы стратегического академического лидерства Казанского (Приволжского) федерального университета («ПРИОРИТЕТ-2030»).

---

<sup>19</sup> ИТИС – Институт информационных технологий и интеллектуальных систем.

<sup>20</sup> КФУ – Казанский федеральный университет.

---

## СПИСОК ЛИТЕРАТУРЫ

1. *Jitendra M.S.N.V., Amiripalli S.S., Surendra T., Rao R.V., Chowdary P.R.* A study on game development using unity engine // AIP Conference Proceedings. 2021. Vol. 2375. No. 1. P. 040001-1–040001-13.
  2. *Polančec D., Mekterović I.* Developing MOBA games using the Unity game engine // 40th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO). 2017. P. 1510–1515.
  3. *Sia B.N.C.H., Koh Z.W., Chung M., Chen Z., Chua S.H., Ganesan D.A.L., Kuah M.Y., Tey K.J., Yeoh W.E.* Cryptocoinopoly: A Real Time Online Multiplayer Board Game // 14th International Conference on Advanced Computer Theory and Engineering. 2021. P. 27–31.
  4. *Yasir R.M., Asad M., Galib A.H., Ganguly K.K., Siddik M.S.* Godexpo: an automated god structure detection tool for Golang // Proceedings of the 3rd International Workshop on Refactoring. 2019. P. 47–50.
  5. *Fernando R.* Evaluating Performance of REST vs gRPC. 2019.  
URL: <https://medium.com/@EmperorRXF/evaluating-performance-of-rest-vs-grpc-1b8bdf0b22da> (дата обращения: 15.07.2022).
  6. Protocol Buffers. 2019.  
URL: <https://developers.google.cn/protocol-buffers/docs/overview> (дата обращения: 15.07.2022).
  7. *Effendy F. Taufik, Adhilaksono B.* Performance comparison of web backend and database: A case study of node.js, Golang and MySQL, Mongo DB // Recent Advances in Computer Science and Communications. 2021. Vol. 14. No. 6. P. 1955–1961.
  8. *Козар Б.А., Кугуракова В.В., Сахибгареева Г.Ф.* Структуризация сущностей естественного текста с использованием нейронных сетей для генерации трехмерных сцен // Программные продукты и системы. 2022. No. 3. С. 329–339.
  9. *Кугуракова В.В., Абрамов В.Д., Костюк Д.И., Шараева Р.А., Газизов Р.Р., Хафизов М.Р.* Генерация трехмерных синтетических датасетов // Электронные библиотеки. 2021. Т. 24(4). С. 622–652.
  10. *Сахибгареева Г.Ф., Кугуракова В.В.* Практики балансирования компьютерных игр // Программные системы: теория и приложения. 2022. Т. 13. No. 3. С. 255–273.
-

## COMPARISON OF CLIENT-SERVER SOLUTIONS IN THE DEVELOPMENT OF MASSIVELY MULTIPLAYER ONLINE GAMES ON UNITY

I. R. Muhamethanov<sup>1</sup> [0000-0002-0562-1434], M. R. Khafizov<sup>2</sup> [0000-0001-7275-9102],  
A. V. Shubin<sup>3</sup> [0000-0002-6203-3268]

<sup>1,2,3</sup> *Institute of Information Technologies and Intelligent Systems, Kazan (Volga Region) Federal University, ul. Kremlyovskaya, 35, Kazan, 420008*

<sup>1</sup>ilnurhamay@mail.ru, <sup>2</sup>mrimsh@mrimsh.ru, <sup>3</sup>shubin.aleksey.kpfu@gmail.com

### Abstract

This paper presents a critique of the traditional approach used to create a multiplayer game in the Unity real-time interactive application development system, especially in the case of a large number of concurrent users. As a hypothesis, an alternative option, which is not common, but which solves many of the problems of the previous approach, is proposed. Two client-server solutions have been compared for developing multiplayer online games in Unity, and the advantages of both approaches have been described for different cases. A game development architecture using a more up-to-date method is proposed: instead of the Mirror library, a standard toolkit for Unity development, microservices written in Golang are used. We present solid proofs of the preference of the alternative approach, the main advantage of which is the support of modern architecture providing high-speed communication between microservices, supported tests on messaging on different platforms.

The test results confirm the hypothesis put forth, and we can conclude that the Unity bundle with Golang is more effective for multiplayer video games.

The article also contains basic methods for debugging multi-threaded application in Golang bundled with Unity game development system and suggests a technological method that allows to get a fast way of data transfer between the client and the server.

**Keywords:** *game engine, microservices, Unity, Golang, Mirror, videogame, multiplayer.*

## REFERENCES

1. *Jitendra M.S.N.V., Amiripalli S.S., Surendra T., Rao R.V., Chowdary P.R.* A study on game development using unity engine // AIP Conference Proceedings. 2021. Vol. 2375. No. 1. P. 040001-1–040001-13.
2. *Polančec D., Mekterović I.* Developing MOBA games using the Unity game engine // 40th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO). 2017. P. 1510–1515.
3. *Sia B.N.C.H., Koh Z.W., Chung M., Chen Z., Chua S.H., Ganesan D.A.L., Kuah M.Y., Tey K.J., Yeoh W.E.* Cryptocoinopoly: A Real Time Online Multiplayer Board Game // 14th International Conference on Advanced Computer Theory and Engineering. 2021. P. 27–31.
4. *Yasir R.M., Asad M., Galib A.H., Ganguly K.K., Siddik M.S.* Godexpo: an automated god structure detection tool for Golang // Proceedings of the 3rd International Workshop on Refactoring. 2019. P. 47–50.
5. *Fernando R.* Evaluating Performance of REST vs gRPC. 2019. URL: <https://medium.com/@EmperorRXF/evaluating-performance-of-rest-vs-grpc-1b8bdf0b22da> (Accessed 15 July 2022).
6. Protocol Buffers. 2019. URL: <https://developers.google.cn/protocol-buffers/docs/overview> (Accessed 15 July 2022).
7. *Effendy F.Taufik, Adhilaksono B.* Performance comparison of web backend and database: A case study of node.js, Golang and MySQL, Mongo DB // Recent Advances in Computer Science and Communications. 2021. Vol. 14. No. 6. P. 1955–1961.
8. *Kozar B., Kugurakova V., Sakhibgareeva G.* Structuring natural text entities using neural networks for generating 3D-scenes // Software & Systems. 2022. Vol. 35. No. 3. P. 329-339.
9. *Kugurakova V., Abramov V., Kostyuk D., Sharaeva R., Gazizov R., Khafizov M.* Generation of three-dimensional synthetic datasets // Russian Digital Libraries Journal. 2021. Vol. 24. No. 4.
10. *Sahibgareeva G., Kugurakova V.* Game Balance Practices // Program systems: theory and applications. 2022. Vol. 13. No. 3(54). P. 255–273.

## СВЕДЕНИЯ ОБ АВТОРАХ



**МУХАМЕТХАНОВ Ильнур Радикович** – лаборант кафедры программной инженерии Института ИТИС КФУ. Сфера научных интересов: разработка видеоигр, информационные технологии.

**Ilnur Radikovich MUKHAMETKHANOV** – laboratory assistant of the software engineering department of the Institute of ITIS KFU. Research interests: video games development, information technologies.

e-mail: [ilnurhamay@mail.ru](mailto:ilnurhamay@mail.ru)

ORCID: 0000-0002-0562-1434



**ХАФИЗОВ Мурад Рустэмович** – старший преподаватель Института ИТИС КФУ. Сфера научных интересов: разработка видеоигр, виртуальная реальность, синтетические данные.

**Murad Rustemovich KHAFIZOV** – senior lecturer at the Institute of Information Technologies and Intelligent Systems, Kazan Federal University. Research interests: game development, virtual reality, synth data.

e-mail: [murcorp@gmail.com](mailto:murcorp@gmail.com)

ORCID: 0000-0001-7275-9102



**ШУБИН Алексей Витальевич** – лаборант кафедры программной инженерии Института ИТИС КФУ. Сфера научных интересов – разработка видеоигр, игровой дизайн.

**Aleksey Vitalevich SHUBIN** – lab assistant at the Department of Software Engineering of the Institute of ITIS KFU. Research interest – videogame development, game design.

e-mail: [shubin.aleksey.kpfu@gmail.com](mailto:shubin.aleksey.kpfu@gmail.com)

ORCID: 0000-0002-6203-3268

*Материал поступил в редакцию 16 сентября 2022 года*

## ПОДХОДЫ К ПРОЕКТИРОВАНИЮ ВИРТУАЛЬНЫХ ТРЕНАЖЁРОВ ХИРУРГИЧЕСКИХ ОПЕРАЦИЙ

Р. А. Шараева<sup>1</sup> [0000-0002-2359-1873], В. В. Кугуракова<sup>2</sup> [0000-0002-1552-4910],

Р. Р. Галиева<sup>3</sup> [0000-0002-6468-2069], С. В. Зинченко<sup>4</sup> [0000-0002-9306-3507]

<sup>1,2,3</sup>Институт информационных технологий и интеллектуальных систем,

<sup>4</sup>Институт фундаментальной медицины и биологии

Казанского (Приволжского) федерального университета;

<sup>1</sup>r.sharaeva3496@gmail.com, <sup>2</sup>vlada.kugurakova@gmail.com,

<sup>3</sup>galievarailina@gmail.com, <sup>4</sup>zinchenkos.v@mail.ru

### **Аннотация**

Показаны достоинства симуляционного обучения в тренинге клинических навыков. Представлено, как практики, которые применяются в инвазивной хирургии, могут быть перенесены в виртуальную операционную. Рассмотрены примеры реализаций тренажёров для определения оптимальных подходов к реализации и выстраиванию оптимального процесса разработки. Как логичное развитие функционала, представлены подходы, которые смогут позволить эмулировать все практики, которые происходят при реальных операциях, в т. ч. многопользовательский режим, вариативность сценариев операций, достижение освоения мануальных техник, а также необходимость внесения инцидентов, инициирующих психологическое выгорание будущих хирургов и осознанный выбор специализации.

**Ключевые слова:** симуляционный тренажёр, виртуальная реальность, хирургия, медицинское образование, Unity, Unreal Engine.

### **ВВЕДЕНИЕ**

Преобразования в современном медицинском образовании направлены на формирование у студентов навыков врача общей практики. Безусловно, лишь небольшая часть студентов в последующем посвятит себя хирургии. Однако существует ряд вмешательств, выполнить которые в экстремальных условиях обязан



врач любой специальности: навыки квалифицированной временной и окончательной остановки кровотечения, хирургической обработки ран, трахеостомия и некоторые другие экстренные вмешательства – и это предусмотрено программами обучения в высших образовательных медицинских учреждениях.

Для сохранения преемственности в преподавании изучение соответствующего предмета должно быть построено по принципу «от простого к сложному» с постоянным повторением наиболее важных разделов хирургии и отработкой практических навыков на симуляторах разной степени сложности. *«... В условиях постоянного дефицита рабочего времени и ужесточающегося медико-юридического прессинга хирургическая симуляция позволяет сократить разрыв между теоретическими знаниями и практическими навыками у интернов. Ранее инвазивные вмешательства отрабатывали при препарировании трупов и в экспериментах на животных. Хотя эти операции дают ценный опыт и не требуют сложной технологической поддержки, трупы и лабораторные животные относятся к ограниченно доступным и дорогостоящим ресурсам ...» [1].*

Развитие технологии виртуальной реальности (VR), разработка новых гарнитур, создание новых алгоритмов компьютерной графики, а также достижения тактильной обратной связи позволили создать за два последних десятилетия ошеломляющие тренажёры для повышения квалификации как нейрохирургов [2], так и других специалистов (см., например, [3–5]).

Подготовка по общей хирургии, как и в других хирургических специальностях, традиционно являлась передачей мастерства «от учителя к ученику». Но в современную эпоху доказательной медицины и медицины, основанной на результатах, акцент на безопасности пациентов и эффективности работы хирургов [6] означает, что этот подход устаревает [7]. Инструменты симуляционного обучения в хирургии являются многообещающим дополнением к другим методам обучения, позволяющим усовершенствовать хирургические навыки, снижая при этом нагрузку на преподавателей и обучающихся [8]. Можно утверждать, что симуляционное обучение лучше, чем отсутствие обучения, и не может быть большим сюрпризом; цитируя [9]: *«Нам не нужно сравнивать что-то с ничем ... Мы примем без доказательств, что некоторое образование лучше, чем его отсутствие».*

Например, в [10] показано, что примерно 4 часа обучения на VR-симуляторе значительно эффективнее, чем полдня обучения на пациентах под наблюдением врача. Вопрос уже не в том, нужно ли заниматься на тренажёрах, а в том, как проводить симуляционное обучение и как разрабатывать эффективные симуляторы.

Безусловно, эти инструменты требуют научной оценки для определения их образовательных преимуществ. Многие симуляторы были исследованы на предмет их валидности, но использование различных рамок валидности в разных исследованиях привело к неубедительным результатам, поэтому показать полную валидность во всех областях ещё предстоит.

Создание профессиональных симуляционных продуктов, которые используются как медицинскими вузами, так и медицинскими учреждениями (больницами, клиниками), является эффективным способом решения потенциальных проблем, связанных с затратами, лицензиями, логистикой и т. д. Однако нужно подчеркнуть, что *«симуляция не должна быть одноразовым учебным упражнением»* [11]. Отмечается, что обучение, когда тренировочные сессии распределены на несколько дней, более эффективно, чем массовая практика в обучающем центре. Практично, что несколько обучающихся могут использовать для обучения один тренажёр или одну виртуальную гарнитуру, а занятому (и дорогостоящему) консультанту не нужно постоянно присутствовать [12]. Направленное, саморегулируемое обучение, когда обучаемым разрешается получить свой собственный опыт и обучаться на своих ошибках, может улучшить запоминание материала по сравнению с обучением под руководством инструктора [13].

Итак, уже существует большое количество доказательств эффективности симуляционного обучения в тренинге клинических навыков. Симуляция должна быть включена в учебную программу в виде распределённых учебных занятий с возможностью направленного, саморегулируемого обучения в профессиональных учебных заведениях. Обучение на основе симуляции до получения навыков должно быть обязательным, прежде чем обучающимся будет разрешено выполнять процедуры на пациентах.

Несмотря на то, что рынок тренажёров активно развивается, и это показывают периодические обзоры [2–5], их всё ещё недостаточное количество при возрастающей актуальности их использования. Поэтому цель нашей работы состоит в

обосновании оптимальных подходов к технической реализации – для ускорения или даже автоматизации процесса разработки VR-тренажёров по обучению целого ряда важных операций.

## **1. СВЯЗАННЫЕ РАБОТЫ**

В медицинском образовании существуют тренажёры для различных узких конкретных сфер хирургии: ортопедическая хирургия, лапароскопия, нейрохирургия и т. д. Отдельную группу образуют тренажёры по открытым операциям на внутренних органах – инвазивной хирургии.

Практики, которые применяются в инвазивной хирургии, должны быть перенесены в виртуальную операционную и реализованы как игровые (в широком понимании этого термина) механики, например:

- разрезание мягких тканей (например, кожи виртуальным скальпелем);
- расширение раны виртуальными крючками или расширителями;
- извлечение внутренних органов (например, кишечника наружу);
- пережимание тканей пациента для гемостаза;
- вправление внутренних органов внутрь;
- ушивание краев раны кетгутовой или шелковой нитью [14].

Рассмотрим, как эти и другие практики-механики реализуются в разных тренажёрах.

В рамках работ по программе повышения конкурентоспособности вузов (5/100) сотрудники научной лаборатории SIM (в фокусе практической реализации которой находится разработка симуляционных методов в биомедицине) Казанского федерального университета (КФУ), продолжив серию работ, посвященных практическому применению виртуальной реальности в образовании [15–17], в 2018–2020 годах разработали [18, 19] и в 2020 году зарегистрировали тренажёр обучения аппендэктомии [20] (VR Аппендэктомия), в 2022 году – зарегистрировали тренажёр обучения трахеостомии [21] (VR Трахеостомия), кроме того, в лаборатории заканчивается работа над VR-тренажёром обучения декомпрессионной краниотомии (VR Краниотомия). Ещё на этапе проектирования эти тренажёры проходят строгую валидацию на соответствие этапности проведения операций и применяемых техник на основании реальной медицинской практики в России (под научным руководством д. м. н. С. В. Зинченко).

---

Довольно большой спектр образовательных медицинских VR-тренажеров приведен в табл. 1 [22]. Упомянем некоторые из них и приведем другие решения для обучения проведения операций инвазивной хирургии, в основном, без использования гарнитур, эмулирующих тактильную отдачу:

1. **Трахеостомия** (кроме разрабатываемой в КФУ) – Самарский государственный медицинский университет (СамГМУ) в 2018 году разработал и получил свидетельство о государственной регистрации SurgeraVR – программы-симулятора открытой хирургии трахеостомии с применением технологии виртуальной реальности. С реализацией можно познакомиться на видео в их магазине SimMed (<https://sim-med.ru/simulator/vr/tracheostomy-and-conicotomy/>). В качестве удачных механик, реализованных в продукте, стоит отметить:

- реалистичное отображение рук пользователя, демонстрирующее верный хват инструмента;
- для визуализации расширения ран используются манипуляции с процедурной полигональной сеткой в реальном времени, что несомненно является плюсом (однако из-за недостаточной графической детализации теряется иммерсивность).

2. **Краниотомия** (кроме разрабатываемой в КФУ) – СамГМУ представляет Аппаратно-программный комплекс для 3D-симулятора в виртуальной реальности «Трепанация черепа – доступы к основанию черепа», также нужно отметить уникальные тренажеры для отработки техники выполнения хирургических вмешательств на мозговом отделе черепа и головном мозге, имитирующие реальную топографо-анатомическую среду:

- Санкт-Петербургский медицинский университет им. И.И. Мечникова (СЗГМУ) разработал симуляционный комплекс для освоения мануальных хирургических навыков на мозговом отделе черепа [23].
- Пермский государственный медицинский университет курирует разработку тренажера, на котором врачи смогут отрабатывать технику трепанации черепа [24]. Эта работа интересна указанием основных этапов, которые легко перенести в виртуальную реальность.

### 3. **Ортопедия.**

- Представлена американскими образовательными VR-тренажерами Osso

VR (<https://www.ossovr.com/>) для изучения этапов сложных ортопедических процедур, требующих множества инструментов, в том числе точной установки штифтов/винтов, необходимых для многих ортопедических процедур. Важность точного чрескожного введения штифта особенно актуальна при фиксации соскальзывающего капитального эпифиза бедра. Этот симулятор отличается хорошей графикой, реалистичными моделями врачей и различными механиками – например, вкручиванием хирургических инструментов. Удачно выглядит возможность сравнения снимков костей пациента, обеспечивающая большее вовлечение в процесс. В [25] проведено сравнение производительности начинающих хирургов-стажёров, которые обучались с использованием симуляции в VR, и тех, кто не имел доступа к VR-тренингу и изучал только стандартное руководство по технике проведения.

- Fundamental Surgery (<https://fundamentalsurgery.com/>), кроме ортопедии, представляет целый комплекс VR-тренажёров обучения проведения различным операциям (например, в офтальмологии). Их решения особо интересны нам тем, что предлагают удалённый многопользовательский режим, позволяющий отработку ассистирования и совместного принятия решения – мультиплеер (см. п. 4.1), организованный с помощью CollaborationVR.

- СамГМУ также анонсировал продолжение собственных разработок в направлении симуляции проведения хирургического лечения переломов и заболеваний костей (остеосинтеза и резекции остеосаркомы).

- Реалистичный опыт того, с чем сталкиваются хирурги-ортопеды в реальной операционной, представляют VR-тренажёры канадской компании PrecisionOS (<https://www.precisionostech.com>) [22].

4. **Неотложная помощь.** Специалисты РНИМУ им. Н.И. Пирогова представили несколько VR-тренажёров: (1) по оказанию скорой и неотложной помощи в санитарном автомобиле скорой медицинской помощи (6 клинических ситуаций) и (2) по отработке алгоритма оказания неотложной хирургической помощи [26].

5. **Эндоскопия.** СамГМУ разработал VR-тренажёр для проведения эндоскопических операций [27]. Фокус этого тренажёра: реализация тактильной обратной связи, чем и обусловлено сотрудничество с Самарским электромеханическим заводом корпорации РосТех. Стоит подчеркнуть интересную практику использования имитаторов реальных рентгенографических инструментов.

6. **Сердечно-легочная реанимация и экстренная рестернотомия** после кардиохирургических операций – рассмотрены два клинических сценария (шоковая и нешоковая остановки сердца) [28].

## **2. ВИРТУАЛЬНАЯ ОПЕРАЦИОННАЯ**

Данные объективных показателей эффективности указывают на то, что VR-тренажёры могут улучшить освоение хирургических навыков, особенно по сравнению с тренировками без использования VR. Например, в предварительных исследованиях [29–33] такой VR-тренинг привел к значительному улучшению времени до завершения и эффективности движений. В качестве наиболее часто обсуждаемого недостатка VR-тренажёров называют (см., например, [29, 31, 33, 35–38]) отсутствие реалистичной тактильной обратной связи, что может ограничить передачу навыков пациентам. Однако реалистичные слуховые стимулы могут дополнить такое мультисенсорное обучение в специальностях, связанных со сверлением кости (например, в ортопедической хирургии, нейрохирургии и отоларингологии). В [39] показано, что слуховые стимулы могут внести такой же вклад, как и тактильные, в контроль экспертами-хирургами глубины погружения при бикортикальном сверлении кости. Рассмотрение возможности улучшения тактильной и слуховой обратной связи может повысить эффективность VR-тренинга. Более того, создание гибкости в симуляциях, включая вариации сложности или интраоперационные осложнения, может тренировать другие ключевые навыки в хирургии, такие как способность принимать решения [32, 35].

### **2.1. Постановка задачи от практикующего врача-хирурга и лектора**

Чтобы виртуальный тренажёр соответствовал корректным оперативным действиям, сценарий должен быть основан на протоколах и стандартах проведения операций. Несмотря на чёткий регламент, у каждого хирурга или медицинского учреждения по мере приобретения опыта возникают свои подходы для улучшения процесса операции. Таким образом, наше решение основано на операционной практике, выработанной в Университетской клинике КФУ.

По нашему мнению, Виртуальная операционная должна включать в себя следующий комплекс виртуальных операций, необходимых для освоения студентами

медицинских высших учебных заведений, как обязательная практика, так как зачастую они не получают такого опыта во время очного обучения по учебной программе: *клиника перфоративной язвы желудка, холецистэктомия, кесарево сечение, катетеризация мочевыводящих путей у мужчин, коникотомия, аппендэктомия, трахеостомия, трепанация черепа, в частности, декомпрессионная краниотомия*. Три из последних упомянутых операций уже реализованы и представлены ниже. Как дополнительные главы комплекса могут быть разработаны также другие операции, которые необходимо освоить всем врачам-практикам хотя бы поверхностно, например, *проведение подкожных и внутривенных инъекций, ушивание сложных ран, остановка кровотечения* и т. д.

### **2.1.1. Аппендэктомия: фокус обучения**

Операция аппендэктомии заключается в отсечении воспаленного аппендикса. В реализованном нами решении VR Аппендэктомия [20] операция выполняется разрезом через точку Мак Бурнея по следующему сценарию:

1. Обработка антисептиком операционной зоны: в качестве антисептика используется йодный раствор.
2. Рассечение кожи и подкожной клетчатки через точку Мак Бурнея.
3. Рассечение апоневроза.
4. Расслоение мышц живота.
5. Рассечение поперечной фасции и брюшины.
6. Извлечение аппендикса: аппендикс допускается извлекать окончательным зажимом во избежание повреждения кишечника.
7. Отсечение брыжейки аппендикса.
8. Перевязка основания аппендикса: зажать основание аппендикса кровоостанавливающим зажимом, чтобы не было рецидива аппендицита.
9. Перевязка основания брыжейки с прошиванием: обязательно нужно закрепить нить, чтобы снизить риск рецидива.
10. Наложение серозно-мышечного кисетного шва, отступив от основания аппендикса 1,5 см.
11. Удаление аппендикса: возрастает риск нагноения послеоперационной раны и перитонита в случае, когда выше места перевязки аппендикса не был наложен зажим.

12. Обработка йодом культи аппендикса: если пропустить этот шаг, возможен культевой абсцесс и перитонит.

13. Вправление отростка внутрь слепой кишки.

14. Наложение Z-образного шва.

15. Ревизия 80 см тонкой кишки на предмет Меккелева дивертикула.

16. Вправление слепой кишки.

17. Ушивание фасции живота.

18. Ушивание мышц.

19. Ушивание апоневроза.

20. Ушивание кожи и подкожной клетчатки.

Во время выполнения операции необходимо проводить гемостаз, чтобы не допустить удлинения продолжительности операции из-за кровотечения и отсутствия обзора, а также нагноения послеоперационной раны.

### **2.1.2. Трахеостомия: фокус обучения**

Трахеостомия — это экстренная операция, выполняемая на шее и заключающаяся во введении в трахею специальной трубки, облегчающей дыхание пациента. В реализованном нами решении VR Трахеостомия [21] операция выполняется пошагово, как и в предыдущем случае (см. п. 2.1.1) и состоит из следующих шагов:

1. Обработка антисептиком: выполняется классическая обработка йодным раствором.

2. Подготовка пациента: лицо пациента необходимо накрыть стерильной салфеткой, поверх которой нужно покрыть всё тело пациента покрывалом с вырезом для операционной зоны.

3. Местная анестезия области шеи: в качестве анестезии используется концентрированный 2% раствор лидокаина.

4. Оперативный доступ: нужно сделать продольный разрез от уровня перстневидного хряща до яремной вырезки.

5. Мобилизация передних мышц шеи с обнажением щитовидной железы.

6. Обработка перешейка щитовидной железы и сосудов.

7. Местная анестезия слизистой трахеи.

8. Вскрытие трахеи с резекцией полукольца: отсекается треть четвертого полукольца трахеи.



9. Санация катетером для бронхиального дерева 3 мм.
10. Фиксация кожи к трахее, ушивание и герметизация раны.
11. Установка трахеостомической трубки и перевязка нитей.

Так как операция выполняется на шейном отделе, важно добавить звуки откашливания пациента и визуальные эффекты выделения слизи.

### **2.1.3. Декомпрессионная краниотомия: фокус обучения**

В случаях, когда у пациента в мозговом отделе возникают гематома или опухоль, декомпрессионная краниотомия позволяет снизить давление и ослабить боли пациента. В реализованном нами решении VR Краниотомия (находится на стадии регистрации) операция включает в себя следующие этапы:

1. Укладка пациента: пациент должен лежать на боку, чтобы хирургу была доступна височная часть головы.
2. Обработка антисептиком операционного поля: в качестве антисептика используется классический йодный раствор.
3. Подготовка пациента: фломастером обозначается линия разреза на височной части пациента, самого пациента накрывают стерильным покрывалом.
4. Рассечение кожно-апоневротического лоскута.
5. Гемостаз краев раны клипсами.
6. Рассечение височной мышцы по ходу волокон и ее оттягивание кпереди и книзу.
7. Инфильтрация и отслаивание надкостницы от кости.
8. Наложение фрезевых отверстий: выполнение возможно и перфоратором, и коловоротом.
9. Резекция костного лоскута колостомом или пилой Джильи.
10. Резекция твердой мозговой оболочки.
11. Ушивание надкостницы, височной мышцы и кожи головы.

Наше решение рассматривает непосредственно краниотомию, однако операция может включать в себя и шаги, связанные с удалением новообразований или гематом.

### **3. Наш опыт: проблемы и решения**

Для реализации наших тренажёров, разрабатываемых на протяжении некоторого времени, были использованы как разные технологии, в частности, разные системы разработки интерактивных приложений, или *игровые движки* (от англ. game engine), так и разные подходы, которые позволили выстроить финальный процесс разработки, или *пайплайн* (от англ. pipeline), который мы планируем применять при создании следующих тренажёров из комплекта обязательных операций для ознакомления студентов-медиков. Во время выработки этого пайплайна перед нами встали следующие вопросы.

#### **3.1. Можно ли создать универсальный конструктор?**

Изначально был выработан подход для реализации виртуальных операций, представляющий собой универсальный конструктор, который содержит в себе различные уникальные механики. Среди них можно выделить разрезание тканей тела в произвольных местах, ушивание рассечённых тканей, обработку поверхностей антисептиком. Но такое решение требует обработки в реальном времени, что вызывает чрезмерную нагрузку на аппаратные мощности [40].

Так как человеческое тело содержит много слоёв плоти (кожа, подкожная клетчатка, апоневроз, мышцы и др.) и внутренних органов, реализация решения с возможностью проведения операции на всём теле в реальном времени достаточно трудоёмка. Поэтому акцент в обучении был смещён в сторону пошагового выполнения действий на определённом участке тела по строгому линейному сценарию.

#### **3.2. Влияние игровых движков на пайплайн разработки**

Большинство источников (см., например, [41, 42]) утверждает, что в разработке приложений расширенной реальности (XR) предпочтительнее использовать Unity (<http://www.unity.com/>), отмечая при этом, что выбор игрового движка – это лишь первый шаг. Поэтому первые наработки в реализации виртуальных операций (см. п. 2.1.1) были сделаны именно на этом движке [19]. Так как качество графики для нас было неудовлетворительным, было принято решение следующую операцию (см. п. 2.1.2) реализовать на альтернативном движке. Наш выбор был сделан

в пользу движка Unreal Engine (<https://www.unrealengine.com/>), который отличается огромным преимуществом в работе с новейшими графическими технологиями, предоставляя высокий уровень реализма без дополнительных работ по настройке и наладке процесса графического рендеринга, в то время как в стандартной поставке Unity имеет весьма скучную графику, а чтобы сделать очень оптимальную и реалистичную 3D-картину, необходимо приложить достаточно много усилий. Кроме того, последние версии движка Unity стали отличаться низкой стабильностью ключевых компонентов [43].

Также преимуществом движка Unreal Engine является встроенная система визуального скриптинга Blueprints. Этот инструмент позволяет в короткие сроки вносить поправки и редактировать классы объектов.

### **3.3. Оптимальный пайплайн, выработанный нами**

Апробация движков показала очевидные преимущества движка Unreal Engine (см. п. 3.2), поэтому последующие операции, входящие в пакет обязательных операций, например, п. 2.1.3, были реализованы именно на этом движке.

Сборка приложений принимает шаблонный характер. Созданная архитектура тренажёра, трехмерные модели окружения, тела пациента, его внутренних органов, дублирующиеся анимации тканей пациента, пользовательский интерфейс могут быть использованы повторно. Однако уникальные разрезы и манипуляции должны быть интегрированы отдельно. Таким образом, наработанные механики и артефакты образуют некую библиотеку данных, к которым можно обратиться отдельно, не создавая большую нагрузку на аппаратные мощности, как в случае п. 3.1. Так как разработка моделей и трехмерных анимаций ведется отдельно, доступно лёгкое портирование<sup>1</sup> на другие гарнитуры виртуальной реальности, что решает проблему устаревания аппаратного и программного обеспечения.

Каждая виртуальная операция представляет собой отдельное приложение. Сценарий виртуальной операции выполняется пошагово и дополняется подсказками: для каждого шага подсвечивается инструмент, который необходимо взять;

---

<sup>1</sup> Портирование (англ. porting) или портация — адаптация программного обеспечения для работы в другой среде, отличающейся от той, под которую ПО была изначально написано, с максимальным сохранением её пользовательских свойств.

---

после взятия инструмента «в руку» появляется полупрозрачный силуэт инструмента над телом пациента, показывающий место для установки инструмента; имеются визуальная подсказка линии разреза; отображение названия инструмента при наведении на объекты операционной. Для повышения иммерсивности [15] в тренажёры добавляются звуковые эффекты откашливания пациента, открывания/закрывания инструментов, звуки оборудования и фоновый шум.

Тело пациента сохраняет реальные размеры, однако внутренние органы могут быть слишком мелкими для освоения правильности манипуляций. Для решения этой проблемы в приложение интегрировано дополнительное окно, отражающее ход операции в увеличенном виде (см. рис. 1).



Рис. 1. Увеличенное изображение операционного поля

#### **3.4. Симуляция разрезания тканей и осмотра внутренних органов**

В зависимости от проводимой инвазивной операции изменяется расположение разрезов. В реализованном решении выполняется симуляция определенного разреза, через который пользователь получает доступ к другим слоям тела пациента и его внутренним органам. Такой подход позволяет повысить иммерсивность

обучения, готовя будущего хирурга к поведению мышечной, жировой ткани и других слоев человеческой плоти (далее – ткань), к появлению кровянистых выделений и слизи.

Разрезание ткани реализовано с помощью блендшейпов – 3D-моделей идентичной топологии [44]. Для этого сначала проводятся подготовительные операции по созданию блендшейпов: создается дубликат модели ткани, затем вершины дублированной модели переносятся для создания необходимой итоговой модели, например, разрезанной скальпелем. После этого 3D-редактор предлагает использовать управляющие ползунки для каждого из блендшейпов и установить ключевые кадры, зафиксировав значения блендшейпа в нужный момент времени – так создаются анимации.

### **3.5. Реализация механик операций**

Система визуального скриптинга Blueprints позволяет создать множество разных объектов, которые наследуются от одного класса-родителя. Кроме того, движок включает в себя шаблон для VR, содержащий базовые функции телепортации, функции захвата и прикрепления предметов к руке и другие. Базовой отправной точкой программы стал такой шаблонный проект, из которого были удалены все механики, ненужные для проекта [45].

Основными объектами сцены являются хирургические инструменты. Класс каждого такого инструмента наследуется от базового класса инструмента *TMedicalTool* и содержит в себе компонент *TAnimationComponent*, который управляет анимациями: хранит в себе необходимые ссылки на анимационные ассеты, проигрывает с нужной скоростью, подписывается на события объектов класса *UAnimNotify* [46]. Также каждый класс инструмента реализует интерфейс *PickupActor*, ответственный за взятие в руки и опускание из рук.

Скелетом программы, отвечающим непосредственно за ход сценария, является класс *TScenarioManager*. Он содержит в себе компонент *TSequenceComponent*, который хранит последовательность инструментов: какие типы хирургического прибора за какими должны следовать – и дополнительные сведения, касающиеся непосредственно инструмента: кто его должен взять (хирург или ассистент), зависят ли данные типы инструмента от какого-то другого (например, иглодержатель и нить), номер шага.

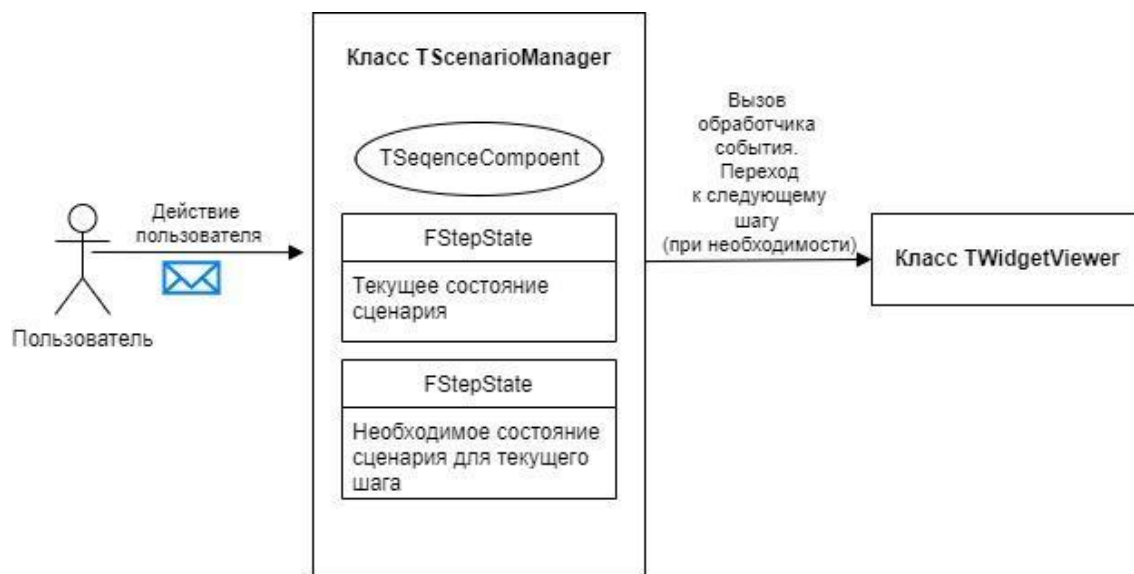


Рис. 2. Общая схема взаимодействия класса сценария с доской

Сам класс *TScenarioManager* содержит в себе массив структур *FToolstepInfo*, которые хранят состояния и действия для каждого шага. При переходе к каждому новому шагу из последовательности извлекается структура *FToolstepInfo* для следующего шага. *TScenarioManager* реагирует на все действия и события, происходящие в сценарии: поднятие инструмента, выбрасывание инструмента в урну, конец проигрывания анимации, – и в ответ на них вызывает соответствующие функции: проигрывание анимации, создание инструментов-подсказок (если это требуется в структуре *FToolstepInfo* для данного шага) и т. д.

Кроме того, класс сценария хранит в себе состояние сценария, представленное в виде структуры *FStepState*: информацию о том, какие действия уже были совершены на текущем этапе. Если состояние сценария соответствует конечному состоянию данного шага (конечное состояние сценария тоже хранится в структуре *FToolstepInfo*, т. е. структура *FToolstepInfo* содержит в себе структуру *FStepState*), то данный этап считается завершенным, и сценарий переходит к следующему шагу (см. рис. 2), оповещая об этом пользовательский интерфейс со сценарием (объект класса *TWidgetViewer*).

Такая тактика позволяет в короткие сроки добавить новые шаги в сценарий или изменить последовательность шагов сценария.

### 3.6. UX/UI

Создание интерфейса пользователя в VR имеет серьезные трудности. Реализованные традиционные виртуальные пользовательские интерфейсы, «привязанные» к голове (когда голова перемещается, интерфейс следует за ней), могут вызвать дискомфорт, поскольку у пользователя возникают проблемы со зрительным восприятием. Более того, элементы интерфейса, расположенные по краям поля зрения, менее читаемы, чем те, что расположены в центре, поскольку самое четкое изображение находится в центре взгляда пользователя [47].

В связи с рядом трудностей реализации традиционного пользовательского интерфейса в настоящем проекте интерфейс сгенерирован на трехмерном объекте, помещённом в игровом мире. Трехмерный объект, играющий роль пользовательского интерфейса, представляет собой доску (см. рис. 3), на которой расположены разные элементы интерфейса: (1) полоса прокрутки, содержащая шаги операции, (2) кнопки, позволяющие листать этапы сценария операции, а также выйти из приложения. Взаимодействие с интерфейсом происходит посредством (3) VR-указки: пользователь наводит указку на нужный элемент и подтверждает нажатие кнопкой контроллера.

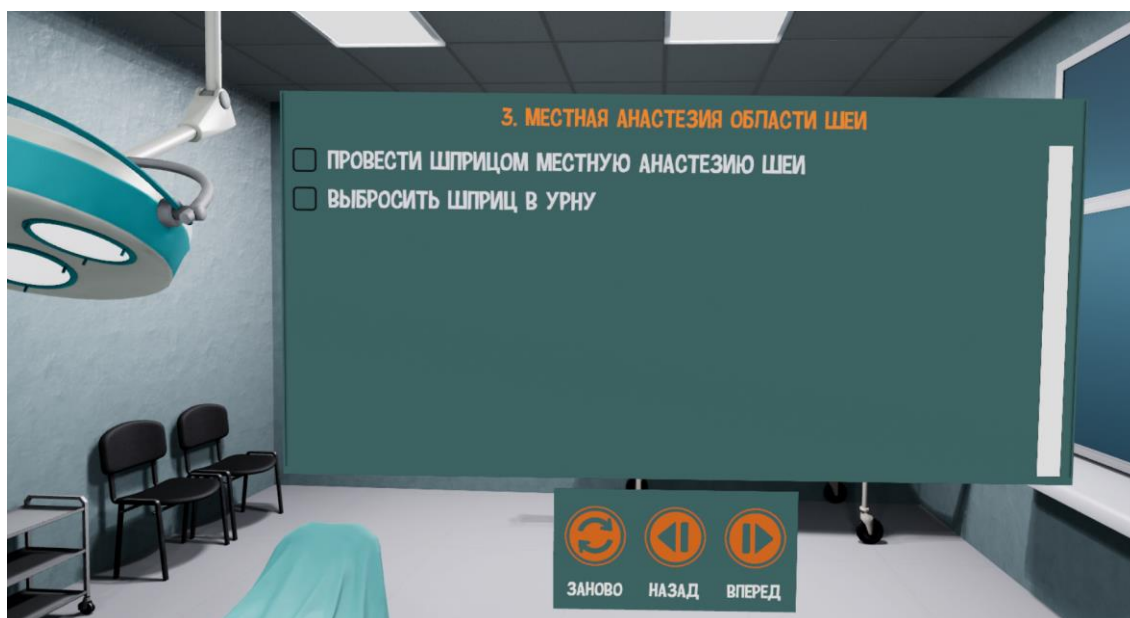


Рис. 3. Доска, играющая роль пользовательского интерфейса с кнопками

Для создания такой модели взаимодействия были использованы встроенные средства игрового движка Unreal Engine:

---

- *UMG* – инструмент, позволяющий создавать иерархические пользовательские интерфейсы и определять их функционал,
- *3D Widget Component* – инструмент, позволяющий показать *UMG*-виджет в 3D пространстве,
- *Widget Interaction Component* – инструмент, позволяющий игроку взаимодействовать с виджетами в 3D пространстве.

Подсказки на инструментах, появляющиеся в процессе операции, представляют собой тот же самый *UMG*-виджет, отображенный на сцене с помощью *3D Widget Component*, но пользователь никак не взаимодействует с ними напрямую: он лишь может влиять на их появление и исчезновение посредством наведения контроллера на нужный инструмент (см. рис. 4).

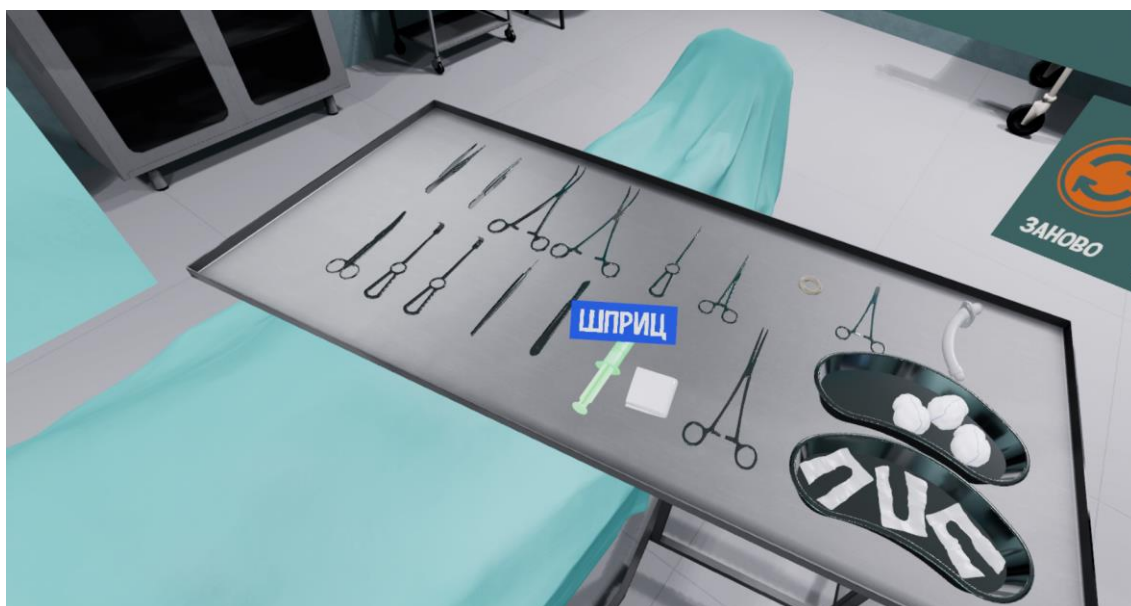


Рис. 4. Подсветка инструмента, появляющаяся в ходе сценария

### 3.7. Ассистирование

Не все шаги операции возможно выполнить в одиночку, поэтому в предложенном нами решении роль ассистентов выполняют виртуальные руки, которые появляются в ситуациях, когда нужно придержать инструменты. Несмотря на то, что определенные шаги в реальных операциях выполняются только ассистентами, наш подход знакомит будущего хирурга и с этими шагами для полноценного освоения тонкостей проведения операции [48].



На рис. 5 представлен момент участия при проведении операции в тренажёре VR Аппендэктомия нескольких виртуальных ассистентов на этапе вырезания аппендикса (*скальпель* держится контролером активного пользователя-студента, четыре *зажима Микулича* фиксируют фасцию и брюшину с четырех краёв): четыре руки отображают участие двух других ассистентов. Первый ассистент *окончатыми зажимами* придерживает правой рукой аппендикс, а левой рукой – слепую кишку, второй ассистент держит *кровоостанавливающие зажимы* правой рукой на основании брыжейки, левой рукой – над местом перевязки.

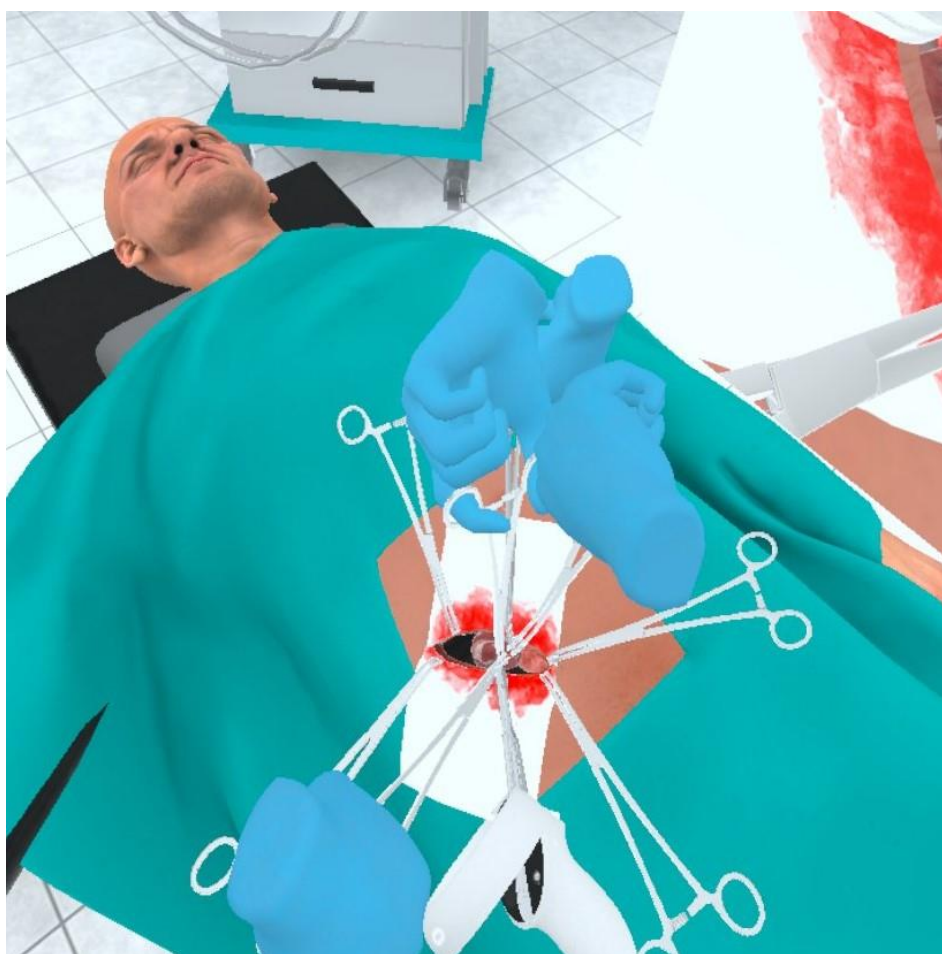


Рис. 5. Несколько виртуальных ассистентов помогают провести резекцию аппендикса

На рис. 6 представлен момент участия двух виртуальных ассистентов при проведении операции в тренажёре VR Трахеостомия, на других этапах проведения операции они помогают при мобилизации передних мышц шеи или перешейка от трахеи, в наложении кровоостанавливающих зажимов и т. п.

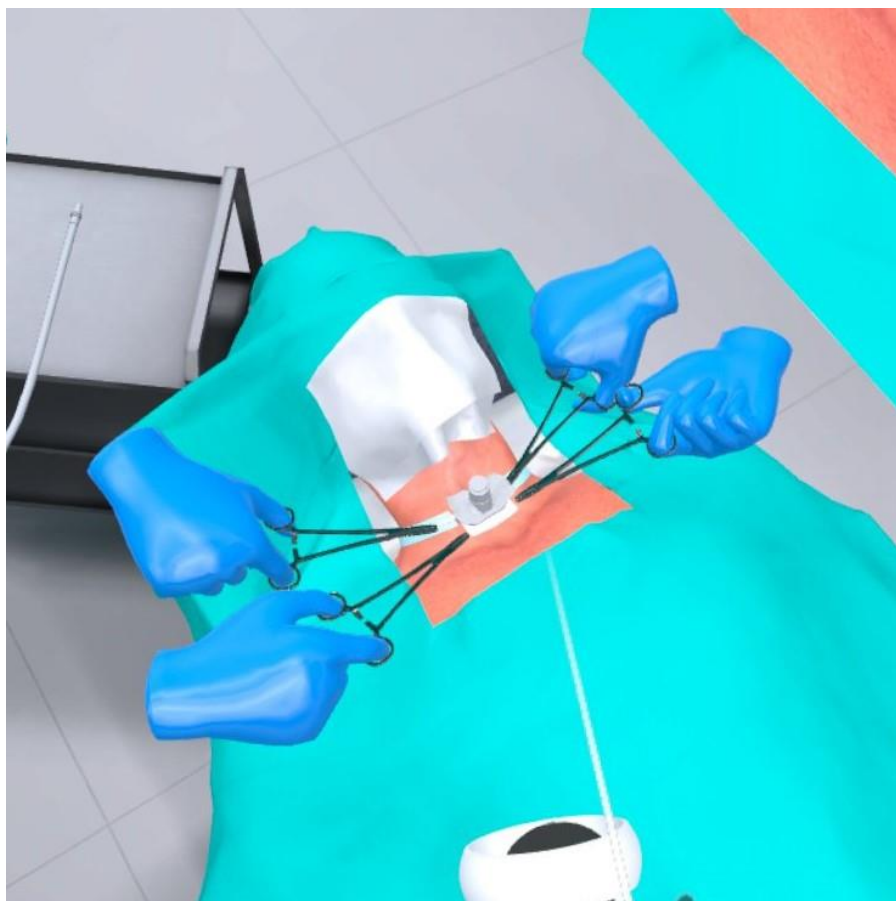


Рис. 6. Несколько виртуальных ассистентов помогают при установке трубки

### **3.8. Управление проектом и автоматизация рутин**

Разработка виртуальных тренажеров относится к сфере разработки компьютерных игр, для которой характерны разделение команды по специализациям, распределение проектных задач по исполнителям в зависимости от их навыка, сильная связность между задачами. За разработку трехмерных моделей в проекте отвечали 3D-дизайнеры, за создание анимаций — аниматоры, а интеграцией моделей, анимаций и сборкой сценариев занимались программисты. Так образовалась цепочка событий: разработка модели→анимация модели→интеграция.

С учетом этих аспектов необходима автоматизация распределения задач по ролям с учетом последовательности выполнения для сокращения рутины менеджмента. Поэтому в управлении проектом по разработке виртуальных хирургических анимаций была использована система таск-трекинга для узкоспециализированных сфер разработки [49, 50]. При создании задач в таск-трекере проектным менеджером указывались метки, определяющие исполнителя из проектной команды, а

настройка зависимостей меток автоматизировала последовательное выполнение задач.

При имеющемся выстроенном процессе реализации проекта автоматизация передачи артефактов и дополнительных комментариев между группами специалистов становится важным условием корректной интеграции. Функции сохранения результатов и автоматического сбора цепочки артефактов, реализованные в таск-трекере, снизили риск неверной интеграции выработанных артефактов.

#### **4. Развитие продукта**

Тренажёр, выработанный благодаря оптимальному пайплайну работы, достаточен для приобретения первичных навыков проведения операции, так как разработан как доступный симулятор высокой достоверности. Таким образом, он является сильным кандидатом для выполнения требований институционального симуляционного обучения в качестве дополнения к учебной программе без исчерпания финансовых ресурсов. Но для повышения эффективности обучения и улучшения продукта необходимо повторить, эмулировать все практики, которые происходят при реальной операции, и для этого мы видим следующее логичное развитие функционала.

##### **4.1. Реализация мультиплеера**

Для совершенствования непрерывного медицинского образования необходимо расширить функционал многопользовательским режимом [51], позволяющим: (1) полноценно отрабатывать процедуры ассистирования; (2) участвовать в удаленных тренингах консилиумов для принятия решения на основе доступных медицинских изображений и свободного обсуждения; (3) отрабатывать совместные действия хирургических бригад вне зависимости от места нахождения обучающихся.

Для разработки многопользовательского режима или *мультиплеера* (от англ. multiplayer) можно применять в случае использования игрового движка Unity связку реализации микросервисов на Golang [52], которая представляется удачной для решений, в которых происходит передача данных в реальном времени, когда даже минимальная задержка может помешать погрузиться в процесс обучения, а

для игрового движка Unreal Engine стоит использовать подход использования одного из клиентов как сервера.

#### **4.2. Режим экзамена и вариативность сценариев операций**

Наличие готового сценария операции позволяет внести изменения в сценарий с целью тестирования знаний будущего хирурга. Разработанный «эталон» пошагового выполнения предполагает использование определенных инструментов. Однако одни и те же шаги операции могут выполняться несколькими допустимыми и недопустимыми инструментами. К примеру, наложение зажима Микулича вместо кровоостанавливающего зажима на кишечник может привести к его повреждению, а отрезание нитей допустимо любыми ножницами.

Порядок действий операций также может быть вариативным. При послойном разрезании слоев тела пациента такая вариативность мало реализуема, но пропуск определенных действий, например, гемостаза, ушивания тканей может привести к нагноению послеоперационной раны, в худшем случае — к смерти пациента.

Таким образом, во время режима экзамена запланирован контроль использования инструментов, пропуска шагов, а также критичности действий пользователя, отклоняющихся от традиционного подхода.

Итак, необходимо использовать потенциал Виртуальной операционной для применения в качестве инструмента объективной оценки для специальностей, основанных на конкретных процедурах, возможно, для дополнения сертификационных экзаменов хирургических советов. Симуляторы должны работать в контексте целостной учебной программы, тогда при долгосрочной интеграции в реальное обучение можно оценить их истинную эффективность [53].

##### **4.2.1. Контроль процесса обучения**

Моделируемая медицинская или хирургическая задача генерирует большое количество данных о том, как пользователь взаимодействует с моделируемым сценарием. Эти данные могут включать как базовые измерения, такие как, например, позиционирование хирургического инструмента, и более сложные измерения, такие как вращение инструмента, силы, приложенные к анатомическим структурам, или объем удаленных анатомических структур.

В [54] представлена прозрачная структура машинного обучения для создания образовательной платформы для сложных психомоторных задач. Для выбора метрик, отражающих текущие показатели опыта в области хирургии, необходимо опираться на опыт практикующих хирургов, чтобы выбрать то, что отражает эффективность работы хирурга при выполнении имитационной задачи. Полезно интегрировать чужие наработки в более ранних симуляционных решениях или попытаться разработать уникальные и ранее неизвестные метрики, способные дифференцировать такой опыт. В хирургическом контексте применение силы инструмента обучающимся трудно оценить инструкторам-хирургам. Компьютерные платформы способны извлекать как позиционные, так и качественные компоненты применения силы обучаемыми [55, 56], что позволяет количественно оценить эти новые метрики хирургической деятельности. Механизм создания метрик может дать новое представление об основополагающих конструкциях опыта для используемого хирургического сценария. Необходимо приложить усилия для создания как можно большего количества различных метрик, чтобы попытаться отразить многочисленные аспекты данного технического навыка.

Затем набор метрик должен быть сужен до тех, которые являются значимыми для алгоритма машинного обучения [57]. Для отбора метрик можно использовать несколько методов (см., например, [58, 59]). В [54] описана нейрохирургическая модель, применяющая 4 метрики (2 метрики безопасности и 2 метрики движения), веса которых несут информацию об относительной важности каждой метрики для процесса принятия решений модели [60, 61]. Метрика, соответствующий вес которой больше, будет играть большую роль в процессе принятия решений алгоритмом. Хотя алгоритм оценил метрику «расстояние до наконечника инструмента» как наиболее важную, авторы решили проконсультироваться с опытными нейрохирургами, чтобы определить, какие метрики будут наиболее важными для обучения в первую очередь. После консультации метрики безопасности были признаны более важными, чем метрики движения [62].

Следуя принципам теории когнитивной нагрузки [63], необходимо ограничить объем информации, предоставляемой пользователю, чтобы облегчить обучение, что достигается путем разделения метрик на группы [64], при этом обратная связь предоставляется поэтапно.

Цифровые платформы становятся всё более распространенными компонентами образовательных парадигм [65]. Однако исследования, связанные с использованием технологий в высшем образовании, выявили, что основными негативными эмоциями, испытываемыми студентами, являются пренебрежение, разочарование, неуверенность, потребность в подтверждении и дискомфорт [66]. Считается, что эти проблемы вызваны «отсутствием обратной связи или неправильным взаимодействием». А концепция *прозрачности оценивания* является центральной для решения этих проблем.

#### **4.3. Достижение иммерсивности в интерактивном взаимодействии**

В [67] проведены эксперименты в VR, доказывающие субъективный характер иммерсивности виртуальной среды, предложены подходы к оценке её уровня и связанных с ним степени вовлеченности пользователя и оценки эффективности его обучения. В применении этих подходов к комплексу Виртуальной операционной среды стоит выделить необходимость высокой реалистичности отображения процессов проведения хирургической операции: симуляции поведения слоев человеческой плоти при разрезании, отодвигании, сшивании и т. п. и поведения жидкостей организма, например, истечения крови.

Пока почти нигде (см. в п. 1 симулятор трахеостомия (СамГМУ) и симуляторы лапароскопии (Эйдос), где реализуются подхватывание манипуляторами плоти, надрезание, отодвигание в режиме реального времени) не используется полная эмуляция физических законов (Гука, тяготения и др.), и это следующий этап для достижения иммерсивности в интерактивном взаимодействии. Разрезание меша [68] в режиме реального времени – в общем случае уже решённая задача.

#### **4.4. Дооперационное прогнозирование осложнений**

Неизбежным спутником хирургии являются различные осложнения, возникающие в послеоперационный период. Они значительно ухудшают результаты хирургического лечения, увеличивают летальность, приводят к существенному увеличению сроков госпитализации пациентов и общих затрат на лечение. Как одна из безусловных практик в предоперационном периоде хирург (и анестезиолог), несмотря на возможный жесткий цейтнот, обязаны детально ознакомиться с состоя-

нием больного и провести его подготовку, направленную если не на полную нормализацию всех функций, то хотя бы на устранение наиболее опасных нарушений деятельности жизненно важных органов и систем. Одним из вариантов обучения в виртуальной операционной должна стать отработка такого сценария предоперационного анализа в режиме решения *случайным образом генерируемых проблем*. Это задача отдельного этапа VR-тренажёра для каждой из операций.

#### **4.5. Проработка осложнений во время операции**

Но возникают осложнения и во время операции, которые не были предусмотрены стандартным протоколом её проведения. Для выработки у студента быстрого принятия решений при возникающих вызовах во время прохождения им этапов операции, возможно уже отточенных в совершенстве, нужно включить режим проработки осложнений. Педагогически верно, чтобы они тоже были случайными и о них не было специального оповещения. Особо интересно реализовать каскадное наращивание проблем, если не было вовремя применено нужное ключевое решение, вплоть до летального исхода виртуального пациента. Казалось бы, точечными сценариями сложно описать всю возможную проблематику, но и проработка в виртуальной реальности вчерне важнейших осложнений уже будет стоить спасения чьих-то реальных жизней.

#### **4.6. Достижение освоения техник разрезания плоти и наложения швов**

Реализованные сценарии операций показывают корректное место рассечения и технику разрезания тканей и наложения швов. При использовании гарнитуры виртуальной реальности в стандартной поставке и применении контроллеров нельзя предложить контроль отработки мануальных техник. Использование цифровых перчаток позволяет эмулировать обратную тактильную связь – как от тяжести инструментов, так и от ощущения плоти пациента.

В КФУ существуют собственные подходы (см., например, [69–71]) к созданию цифровых перчаток и использующих их приложений. Такие перчатки ещё не позволяют получить полноценную обратную связь, но развитие этого направления разработки дополнительной периферии виртуальной реальности перспективно для усиления освоения соответствующих техник хирургии на этапе дооперацион-

ного обучения. Например, экзоскелетные перчатки Dexmo<sup>2</sup> [72] китайской компании Dextra Robotics с тактильной обратной связью (по силе хвата) способны передавать физические ощущения от взаимодействия пользователя с виртуальными объектами, несуществующими в реальности.

#### 4.7. Психологическая подготовка студента

Медицинское образование эволюционировало со времен традиционного метода обучения «*See One, Do One, Teach One*» [7]. Возрастающая роль симуляции должна помочь повысить эффективность приобретения навыков [73]. Хотя отчасти эта тенденция может быть обусловлена сокращением сроков обучения и рабочего времени, в конечном итоге речь идет о пациентах и их жизнях. Структурированные подходы к обучению оперативным навыкам и тщательный контроль со стороны более старших коллег по мере того, как стажер переходит от новичка к эксперту, должны позволить снизить частоту осложнений, когда стажеры оперируют как в качестве основного хирурга, так и в качестве ассистента. Однако, несмотря на хорошую подготовку, когда стажер участвует в реальном уходе за пациентом, неизбежно возникает эмоциональная нагрузка, когда всё идёт не так, как надо. Осложнения неизбежны вне зависимости от опыта, поскольку «хирургия – суровый мастер». Эта прописная истина часто применяется к любой дисциплине, требующей сложных знаний, навыков и принятия решений, что, безусловно, включает все хирургические специальности.

Авторы [74] обнаружили, что инциденты, связанные с безопасностью пациентов, «могут иметь глубокие и длительные последствия для вовлеченных в них медицинских работников», и обсудили концепцию «феномена второй жертвы», когда специалисты испытывают психологический ущерб после инцидентов, связанных с безопасностью пациентов («врачебных ошибок»).

В последние годы наблюдается интерес к концепции «выгорания» среди хирургов, а также к роли нетехнических навыков и командной работы в оказании сложной медицинской помощи. Необходимость хирурга уметь справляться с эмоциональным бременем работы и проявлять «стойкость», чтобы избежать выгора-

---

<sup>2</sup> Dexmo. VR High-risk Industry Training, demo: <https://youtu.be/8cylQigg2GE>



ния, является характеристикой, которую следует учитывать на этапе выбора профессии и которая имеет отношение к обучению в бакалавриате и аспирантуре, а также к выбору специальности.

Хирургия в этом не уникальна – все медицинские профессии связаны со стрессом и способностью к выгоранию. В постпандемическом мире нельзя позволить себе терять врачей, на подготовку которых были затрачены огромные средства. Это будет трагедией, если произойдет в результате предотвратимой эмоциональной травмы, ведущей к выгоранию и ещё более преждевременному раннему выходу на пенсию, чем это происходит в настоящее время [75].

Итак, выявлена необходимость серьезного отбора медиков на ранних этапах их подготовки, с возможной сменой специализации [76–78] во время учёбы, что в свою очередь оправдывает необходимость внесения в виртуальные тренажёры инцидентов, инициирующих выгорание. Такие шоковые процедуры дадут возможность точно взвесить характеристики студентов и обосновать выбор специализации.

## **ЗАКЛЮЧЕНИЕ**

Предложен апробированный процесс разработки виртуальных хирургических тренажёров без использования дополнительных аппаратных устройств, кроме стандартной поставки гарнитур виртуальной реальности. Показаны пути развития комплекса Виртуальная операционная: многопользовательский режим, вариативность сценариев операций, контроль процесса обучения, дооперационное прогнозирование осложнений, проработка осложнений во время операции. Представлена возможность расширения сферы применения комплекса Виртуальная операционная в образовательном процессе при использовании цифровых перчаток, позволяющих обрабатывать в виртуальной реальности разнообразные мануальные техники освоения хирургических инструментов. Отдельное внимание обращено на отработку инцидентов, инициирующих психологическое выгорание будущих хирургов и осознанный выбор специализации.

Необходимо отметить, что доступные симуляторы высокой достоверности для приобретения первичных навыков проведения операций являются в настоя-

щее время необходимым средством институционального симуляционного обучения в качестве дополнения к учебной программе без исчерпания финансовых ресурсов высших учебных медицинских заведений.

### **БЛАГОДАРНОСТИ**

Работа выполнена за счет средств Программы стратегического академического лидерства Казанского (Приволжского) федерального университета («ПРИОРИТЕТ-2030»).

### **СПИСОК ЛИТЕРАТУРЫ**

1. *Спиотта А.М., Шленк Р.П.* Симуляционное обучение нейрохирургических интернов: новая парадигма // Медицинское образование и профессиональное развитие. 2017. № 1 (27). С. 24–30.
2.  *Davids J., Manivannan S., Darzi A., Giannarou S., Ashrafian H. & Marcus H.J.* Simulation for skills training in neurosurgery: a systematic review, meta-analysis, and analysis of progressive scholarly acceptance // Neurosurgical review. 2022. Vol. 44. No. 4. P. 1853–1867.
3. *Haiser A., Aydin A., Kunduzi B., Ahmed K. & Dasgupta P.* A Systematic Review of Simulation-Based Training in Vascular Surgery // Journal of Surgical Research. 2022. Vol. 279. P. 409–419.
4. *James J., Irace A.L., Gudis D.A. & Overdevest J.B.* Simulation training in endoscopic skull base surgery: A scoping review // World Journal of Otorhinolaryngology – Head and Neck Surgery. 2022. Vol. 8. No. 1. P. 73–81.
5. *Saleem H.Y., Kaplan J.L., Torres-Guzman R.A., Avila F.R. & Forte A.J.* Simulation in Hand Surgery: A Literature Review // World Journal of Surgery. 2022. Vol. 46. No. 3. P. 718–724.
6. *Rohrich R.J.* See one, do one, teach one: an old adage with a new twist // Plast. Reconstr. Surg. 2006. Vol. 118. P. 257–258.
7. *Kotsis S.V., Chung K.C.* Application of the «see one, do one, teach one» concept in surgical training // Plast. Reconstr. Surg. 2013. Vol. 131. P. 1194–1201.
8. *Konge L., Lonn L.* Simulation-based training of surgical skills // Perspect. Med. Educ. 2016. Vol. 5. P. 3–4.

9. *Norman G.* Data dredging, salami-slicing, and other successful strategies to ensure rejection: twelve tips on how to not get your paper published // *Adv. Health Sci. Educ. Theory Pract.* 2014. Vol. 19. P. 1–5.

10. *Konge L., Clementsen P.F., Ringsted C., Minddal V., Larsen K.R., Annema J.T.* Simulator training for endobronchial ultrasound: a randomised controlled trial // *Eur. Respir. Journal.* 2015. Vol. 46. P. 1140–1149.

11. *Andersen S.A., Konge L., Caye-Thomasen P., Sorensen M.S.* Learning Curves of Virtual Mastoidectomy in Distributed and Massed Practice // *JAMA Otolaryngol. Head Neck Surg.* 2015. P. 1–6.

12. *Räder S.B., Henriksen A.H., Butrymovich V., et al.* A study of the effect of dyad practice versus that of individual practice on simulation-based complex skills learning and of students' perceptions of how and why dyad practice contributes to learning // *Acad. Med.* 2014. Vol. 89. P. 1287–1294.

13. *Brydges R., Nair P., Ma I., Shanks D., Hatala R.* Directed self-regulated learning versus instructor-regulated learning in simulation training // *Med. Educ.* 2012. Vol. 46. P. 648–656.

14. *Газизов P.P.* Физика веревки для реализации кетгутовой нити в виртуальной операционной // *Ученые записки института социальных и гуманитарных знаний КФУ.* 2019. Т. 17. № 1. С. 572–578.

15. *Kugurakova V.V., Elizarov A.M., Khafizov M.R. et al.* Towards the immersive VR: measuring and assessing realism of user experience // *ICAROB 2018: Proceedings of the 2018 International Conference on Artificial Life and Robotics.* 2018. P. 146–152.

16. *Abramov V.D., Kugurakova V.V., Rizvanov A.A. et al.* Virtual Biotechnological Lab Development // *BioNanoScience.* 2017. Vol. 7. Is. 2. P. 363–365.

17. *Lushnikov A., Kugurakova V., Nizamutdinov A.* Development of VR system to enhance understanding process of robot mechanisms // *ICAROB 2018: Proceedings of the 2018 International Conference on Artificial Life and Robotics.* 2018. P. 140–145.

18. *Kugurakova V.V., Abramov V.D., Sultanova R.R. et al.* Virtual Reality-Based Immersive Simulation for Invasive Surgery Training // *European Journal of Clinical Investigation.* 2018. Vol. 48. P. 224–225.

19. *Sultanova R., Sharaeva R.* Virtual reality-based immersive simulation mechanics for invasive surgery training // Proceedings of International Conference on Developments in eSystems Engineering, DeSE. 2019. Vol. October 2019. P. 924–928.

20. Программа для обучения аппендэктомии в виртуальной реальности: Свидетельство о государственной регистрации программы для ЭВМ №2020613665 Российская Федерация / В.В. Кугуракова, М.Р. Хафизов, В.Д. Абрамов, Р.А. Шараева, Р.Р. Газизов, Т.М. Зиннуров, С.В. Зинченко; заявитель и правообладатель Фед. гос. автоном. образоват. учреждение высш. образ. Казанский фед. ун-т. №2020612602; заявл. 10.03.2020; зарегистрировано в реестре программ для ЭВМ 19.03.2020. 1 с.

21. Программа для обучения трахеостомии в виртуальной реальности: Свидетельство о государственной регистрации программы для ЭВМ № 2022661512 Российская Федерация / В.В. Кугуракова, А.П. Киясов, Р.И. Файзуллин, С.В. Зинченко, Р.Р. Галиева, Р.Р. Газизов, Н.Р. Курбангалиева, Р.А. Шараева, Н.Э. Романчук, М.Р. Хафизов; заявитель и правообладатель Фед. гос. автоном. образоват. учреждение высш. образ. Казанский фед. ун-т. № 2022660504; заявл. от 08.06.2022; зарег. в реестре программ для ЭВМ 22.06.2022. 1 с.

22. *Mao R.Q., Lan L., Kay J., Lohre R., Ayeni O.R., Goel D.P. & SA D.D.* Immersive Virtual Reality for Surgical Training: A Systematic Review // Journal of Surgical Research. 2021. Vol. 268. P. 40–58.

23. *Смирнов А.А., Татаркин В.В., Захматов И.Г., Марьянович А.Т., Андреевская М.В., Назмиев А.И., Кудлахмедов Ш.Ш., Рыбаков В.А.* Тренажёр для освоения мануальных навыков хирургических вмешательств на мозговом отделе черепа // Креативная хирургия и онкология. 2017. № 7(1). С. 63–68.

24. Аналогов в России нет: пермяки придумали тренажёр, на котором врачи научатся делать трепанацию черепа // 59.PY. 2017. URL: <https://59.ru/text/gorod/2017/08/14/50888691/> (дата обращения: 01.09.2022).

25. *Cevallos N., Zukotyński B., Greig D., Silva M. and Thompson R.M.* The Utility of Virtual Reality in Orthopedic Surgical Training // Journal of Surgical Education. 2022. Vol. 79(6). P. 1516–1525.

26. Новый интерактивный образовательный модуль с VR-тренажёром по отработке алгоритма оказания неотложной хирургической помощи доступен на Портале // Портал непрерывного медицинского и фармацевтического образования

---

Минздрава России, 2021. URL: <https://edu.rosminzdrav.ru/anonsy/anonsy/news/novyi-interaktivnyi-obrazovatelnyi-modul-s-vr-trenazh/> (дата обращения: 01.09.2022).

27. Ростех разработал виртуальный тренажёр для обучения хирургов // Ростех, 2022. URL: <https://rostec.ru/news/rostekh-razrabotal-virtualnyy-trenazher-dlya-obucheniya-khirurgov> (дата обращения: 01.09.2022).

28. *Sadeghi A.H., Peek J.J., Max S.A., Smit L.L., Martina B.G., Rosalia R.A., Bakhuis W., Bogers A.J., Mahtab E.A.* Virtual Reality Simulation Training for Cardiopulmonary Resuscitation After Cardiac Surgery: Face and Content Validity Study // *JMIR Serious Games*. 2022. Vol. 10. No. 1. P. e30456.

29. *Atli K., Selman W., Ray A.* A comprehensive multicomponent neurosurgical course with use of virtual reality: modernizing the medical classroom // *Journal of Surgical Education*. 2020. Vol.78(4). P. 1350–1356.

30. *Bing E.G., Parham G.P., Cuevas A., et al.* Using low-cost virtual reality simulation to build surgical capacity for cervical cancer treatment // *Journal Glob. Oncol.* 2019. P. 1–7.

31. *Chaudhary A.H., Bukhari F., Iqbal W., Nawaz Z., Malik M.K.* Laparoscopic training exercises using HTC VIVE // *Intell. Autom. Soft Co.* 2020. Vol. 26. P. 53–59.

32. *Logishetty K., Gofton W.T., Rudran B., Beaulé P.E., Cobb J.P.* Fully immersive virtual reality for total hip arthroplasty: objective measurement of skills and transfer of visuospatial performance after a competency-based simulation curriculum // *Journal Bone Joint Surg. Am.* 2020. P. e27.

33. *Luca A., Giorgino R., Gesualdo L., Peretti G.M., Belkhou A., Banfi G., et al.* Innovative educational pathways in spine surgery: advanced virtual reality-based training // *World Neurosurg.* 2020. P. 674–680.

34. *Arroyo-Berezowsky C., Jorba-Elguero P., Altamirano-Cruz M.A., Quinzanos-Fresnedo J.* Usefulness of immersive virtual reality simulation during femoral nail application in an orthopedic fracture skills course // *Journal Musculoskelet. Surg. Res.* 2019. P. 326–333.

35. *Hooper J., Tsiridis E., Feng J.E., et al.* Virtual reality simulation facilitates resident training in total hip arthroplasty: a randomized controlled trial // *Journal Arthroplasty.* 2019. P. 2278–2283.

36. *Logishetty K., Rudran B., Cobb J.P.* Virtual reality training improves trainee performance in total hip arthroplasty: a randomized controlled trial // *Bone Joint Journal*. 2019. P. 1585–1592.

37. *Lohre R., Bois A.J., Athwal G.S., Goel D.P.* Canadian Shoulder and Elbow Society (CSES). Improved complex skill acquisition by immersive virtual reality training: a randomized controlled trial // *Journal Bone Joint Surg. Am.* 2020. P. e26.

38. *Orland M.D., Patetta M.J., Wieser M., Kayupov E., Gonzalez M.H.* Does virtual reality improve procedural completion and accuracy in an intramedullary tibial nail procedure? A randomized control trial // *Clin. Orthop. Relat. Res.* 2020. P. 2170–2177.

39. *Praamsma M., Carnahan H., Backstein D., Veillette C.J., Gonzalez D., Dubrowski A.* Drilling sounds are used by surgeons and intermediate residents, but not novice orthopedic trainees, to guide drilling motions // *Can. Journal Surg.* 2008. P. 442–446.

40. *Sabbagh A.J., Bajunaid K.M., Alarifi N., et al.* Roadmap for developing complex virtual reality simulation scenarios: subpial neurosurgical tumor resection model // *World Neurosurg.* 2020. P. e220–e229.

41. Comparing Unity vs Unreal for VR, MR or AR Development Projects // *XR Bootcamp*, 2022.

URL: <https://xrbootcamp.com/unity-vs-unreal-engine-for-xr-development/> (дата обращения: 01.09.2022).

42. *Tsarouva M.* Choosing an engine for VR: Unity vs Unreal // *iTechArt*, 2022. URL: <https://www.itechart.com/blog/unity-vs-unreal-virtual-reality/> (дата обращения: 01.09.2022).

43. *Диденко Г.* Орёл или решка: сравнение Unity и Unreal Engine // *DTF*, 2017. URL: <https://dtf.ru/gamedev/7227-orel-ili-reshka-sravnenie-unity-i-unreal-engine> (дата обращения: 01.09.2022).

44. Блендшейпы (Blend Shapes), морфы (Morph), Shape key | Словарь // *ShutterLine*. URL: <https://3dyuriki.com/2010/05/02/blendshejpy-blend-shapes-i-morfy-morph-slovar/> (дата обращения: 01.09.2022).

45. VR Template // *Epic Games*. URL: <https://docs.unrealengine.com/4.27/en-US/Resources/Templates/VRTemplate/> (дата обращения: 01.09.2022).

46. Animation Notifications (Notifies) // Epic Games. URL: <https://docs.unrealengine.com/4.27/en-US/AnimatingObjects/SkeletalMeshAnimation/Sequences/Notifies/> (дата обращения: 01.09.2022).

47. Маккеффи М. Unreal Engine VR для разработчиков. Изд-во Бомбора, 2019. 256 с.

48. Хай Г.А. Ассистирование при хирургических операциях. СПб, 1998. 382 с.

49. Шараева Р.А., Кузурасова В.В., Селезнева Н.Э. Методика упрощения таск-трекинга в проектах игровой индустрии // Программные продукты и системы. 2022. Т. 35. No 3. С. 374–383.

50. Шараева Р.А., Кузурасова В.В. Оценка сокращения времени при использовании модифицированной методики таск-трекинга в управлении ИТ-проектами // Программные системы: теория и приложения. 2022. Т. 13. №3(54). С. 307–324.

51. Bowyer M.W., Streete K.A., Muniz G.M., Liu A.V. Immersive virtual environments for medical training // Semin. Colon. Rectal. Surg. 2008. P. 90–97.

52. Мухаметханов И.Р., Хафизов М.Р., Шубин А.В. Сравнение клиент-серверных решений при разработке многопользовательских онлайн-игр на Unity // Электронные библиотеки. 2022. №5.

53. Carter F.J., Schijven M.P., Aggarwal R., et al. Consensus guidelines for validation of virtual reality surgical simulators // Simul. Healthc. 2006. P. 171–179.

54. Mirchi N., Bissonnette V., Yilmaz R., Ledwos N., Winkler-Schwartz A., Del Maestro R.F. The Virtual Operative Assistant: An explainable artificial intelligence tool for simulation-based training in surgery and medicine // PLoS ONE. 2020. Vol. 15(2): e0229596. P. 1–15.

55. Sawaya R., Bugdadi A., Azarnoush H., Winkler-Schwartz A., Alotaibi F.E., Bajunaid K., et al. Virtual Reality Tumor Resection: The Force Pyramid Approach // Operative Neurosurgery. 2018. P. 686–696.

56. Wagner C.R., Stylopoulos N., Jackson P.G., Howe R.D. The benefit of force feedback in surgery: Examination of blunt dissection // Presence: teleoperators and virtual environments. 2007. P. 252–262.

57. Ladha L., Deepa T. Feature selection methods and algorithms // International journal on computer science and engineering. 2011. P. 1787–1797.

58. Lli J., Cheng K., Wang S., Morstatter F., Trevino R.P., Tang J., et al. Feature selection: A data perspective // ACM Computing Surveys (CSUR). 2018. P. 1–45.

59. Winkler-Schwartz A., Yilmaz R., Mirchi N., Bissonnette V., Ledwos N., Siyar S., et al. Assessment of Machine Learning Identification of Surgical Operative Factors Associated with Surgical Expertise in Virtual Reality Simulation // JAMA Network Open. 2019. P. e198363–e198363.

60. Yu L., Liu H. Efficient feature selection via analysis of relevance and redundancy // Journal of Machine Learning Research. 2004. P. 1205–1224.

61. Wang X., Wang Y., Wang L. Improving fuzzy c-means clustering based on feature-weight learning // Pattern Recognition Letters. 2004. P. 1123–1132.

62. Ryu W.H.A., Chan S., Sutherland G.R. Supplementary educational models in Canadian neurosurgery residency programs // Canadian Journal of Neurological Sciences. 2017. P. 177–183.

63. Sweller J. Cognitive load theory, learning difficulty, and instructional design // Learning and Instruction. 1994. P. 295–312.

64. Gobet F., Lane P.C., Croker S., Cheng P.C., Jones G., Oliver I., et al. Chunking mechanisms in human learning // Trends in Cognitive Sciences. 2001. P. 236–243.

65. Williamson B. Digital education governance: data visualization, predictive analytics, and ‘real-time’ policy instruments // Journal of Education Policy. 2016. P. 123–141.

66. Saplacan D., Herstad J., Pajalic Z. Feedback from Digital Systems Used in Higher Education: An Inquiry into Triggered Emotions—Two Universal Design Oriented Solutions for a Better User Experience // Studies in Health Technology and Informatics. 2018. P. 421–430.

67. Кузурасова В.В. Математическое и программное обеспечение многопользовательских тренажёров с погружением в иммерсивные виртуальные среды: дис. канд. техн. наук: 05.13.11. Казанский федеральный университет, 2018. 187 с.

68. Akhmetsharipov R.D., Khafizov M.R., Lushnikov A.Yu., Zigantdinov Sh.Ya. The soft tissue implementation with triangulated mesh for virtual surgery system // Smart Innovation, Systems and Technologies. 2018. P. 163–167.



69. *Shigapov M.I., Kugurakova V.V.* Design and development of a hardware and software system for simulation of feedback tactility // Proceedings SIBCON. 2021. P. 1–6.

70. *Shigapov M.I., Kugurakova V.V., Zykov E.Yu.* Design of digital gloves with feedback for VR // Proceedings IEEE EWDTs. 2018. P. 1–5.

71. *Аглямов Ф.Р., Кузуратов В.С.* Тренажёр для реабилитации пациентов с проблемами мобильности руки, построенный с использованием технологий виртуальной реальности // Программные продукты и системы. 2022. Т. 35. No 3. С. 285–292.

72. *Friston S., Griffith E., Swapp D., Marshall A., Steed A.* Position-based control of under-constrained haptics: A system for the Dexmo glove // IEEE Robot Autom. [Internet]. 2019. P. 3497–3504.

73. *Bhalla S., Beegun I., Awad Z., Tolley N.* Simulation-based ENT induction: validation of a novel mannequin training model // Journal Laryngol. Otol. 2020. P. 74–80.

74. *McLaren O., Perkins C., Alderson D.* The effect of surgical complications on ENT trainees // Journal Laryngol. Otol. 2021. P. 293–296.

75. *Cherry J., Weir R.* Medicolegal and ethical aspects of ORL-HNS in the new millennium // Journal Laryngol. Otol. 2006. P. 737–740.

76. *Mayer A.W., Smith K.A., Carrie S.* A systematic review of factors affecting choice of otolaryngology as a career in medical students and junior doctors // Journal Laryngol. Otol. 2019. P. 836–842.

77. *Bhutta M., Mandavia R., Syed I., Qureshi A., Hettige R., Wong B.Y.W. et al.* A survey of how and why medical students and junior doctors choose a career in ENT surgery // Journal Laryngol. Otol. 2016. P. 1054–1058.

---

## APPROACHES TO THE DEVELOPMENT OF VIRTUAL SURGICAL TRAINING

R. A. Sharaeva<sup>1</sup> [0000-0002-2359-1873], V. V. Kugurakova<sup>2</sup> [0000-0002-1552-4910],  
R. R. Galieva<sup>3</sup> [0000-0002-6468-2069], S. V. Zinchenko<sup>4</sup> [0000-0002-9306-3507]

<sup>1,2,3</sup> *Institute of Information Technologies and Intelligent Systems,*

<sup>4</sup> *Institute of Fundamental Medicine and Biology*

*Kazan (Volga Region) Federal University, ul. Kremlyovskaya, 35, Kazan, 420008;*

<sup>1</sup>r.sharaeva3496@gmail.com, <sup>2</sup>vlada.kugurakova@gmail.com,

<sup>3</sup>galievarailina@gmail.com, <sup>4</sup>zinchenkos.v@mail.ru

### Abstract

The advantages of simulation training in clinical skills training are shown. Examples of simulators for building an optimal development process are reviewed. A way of transferring invasive surgery practices to the virtual operating room is presented.

As a logical development of functionality, approaches are presented that will be able to emulate all the practices that take place during real operations, including multi-player mode, variability in surgery scenarios, achieving mastery of manual techniques, and the need to introduce incidents that initiate psychological burnout of future surgeons and a conscious choice of specialization.

**Keywords:** *simulator, virtual reality, VR, surgical education, medical education, Unity, Unreal Engine.*

### REFERENCES

1. *Spiotta A.M., SHlenk R.P.* Simulyacionnoe obuchenie nejrohirurgicheskikh inter-nov: novaya paradigma // *Medicinskoe obrazovanie i professional'noe razvitie.* 2017. № 1 (27). S. 24–30.
2.  *Davids J., Manivannan S., Darzi A., Giannarou S., Ashrafian H. & Marcus H.J.* Simulation for skills training in neurosurgery: a systematic review, meta-analysis, and analysis of progressive scholarly acceptance // *Neurosurgical Review.* 2022. Vol. 44. No. 4. P. 1853–1867.
3. *Haiser A., Aydin A., Kunduzi B., Ahmed K. & Dasgupta P.* A Systematic Review of Simulation-Based Training in Vascular Surgery // *Journal of Surgical Research.* 2022. Vol. 279. P. 409–419.

4. *James J., Irace A.L., Gudis D.A. & Overdevest J.B.* Simulation training in endoscopic skull base surgery: A scoping review // *World Journal of Otorhinolaryngology – Head and Neck Surgery.* 2022. Vol. 8. No. 1. P. 73–81.

5. *Saleem H.Y., Kaplan J.L., Torres-Guzman R.A., Avila F.R. & Forte A.J.* Simulation in Hand Surgery: A Literature Review // *World Journal of Surgery.* 2022. Vol. 46. No. 3. P. 718–724.

6. *Rohrich R.J.* See one, do one, teach one: an old adage with a new twist // *Plast. Reconstr. Surg.* 2006. Vol. 118. P. 257–258.

7. *Kotsis S.V., Chung K.C.* Application of the «see one, do one, teach one» concept in surgical training // *Plast. Reconstr. Surg.* 2013. Vol. 131. P. 1194–1201.

8. *Konge L., Lonn L.* Simulation-based training of surgical skills // *Perspect. Med. Educ.* 2016. Vol. 5. P. 3–4.

9. *Norman G.* Data dredging, salami-slicing, and other successful strategies to ensure rejection: twelve tips on how to not get your paper published // *Adv. Health Sci. Educ. Theory Pract.* 2014. Vol. 19. P. 1–5.

10. *Konge L., Clementsen P.F., Ringsted C., Minddal V., Larsen K.R., Annema J.T.* Simulator training for endobronchial ultrasound: a randomised controlled trial // *Eur. Respir. Journal.* 2015. Vol. 46. P. 1140–1149.

11. *Andersen S.A., Konge L., Caye-Thomasen P., Sorensen M.S.* Learning Curves of Virtual Mastoidectomy in Distributed and Massed Practice // *JAMA Otolaryngol. Head Neck Surg.* 2015. P. 1–6.

12. *Räder S.B., Henriksen A.H., Butrymovich V., et al.* A study of the effect of dyad practice versus that of individual practice on simulation-based complex skills learning and of students' perceptions of how and why dyad practice contributes to learning // *Acad. Med.* 2014. Vol. 89. P. 1287–1294.

13. *Brydges R., Nair P., Ma I., Shanks D., Hatala R.* Directed self-regulated learning versus instructor-regulated learning in simulation training // *Med. Educ.* 2012. Vol. 46. P. 648–656.

14. *Gazizov R.R.* Fizika verevki dlya realizacii ketgutovoj niti v virtual'-noj operacii // *Uchenye zapiski instituta social'nyh i gumanitarnyh znanij KFU.* 2019. T. 17. № 1. С. 572–578.

15. *Kugurakova V.V., Elizarov A.M., Khafizov M.R. et al.* Towards the immersive VR: measuring and assessing realism of user experience // ICAROB 2018: Proceedings of the 2018 International Conference on Artificial Life and Robotics. 2018. P. 146–152.

16. *Abramov V.D., Kugurakova V.V., Rizvanov A.A. et al.* Virtual Biotechnological Lab Development // *BioNanoScience*. 2017. Vol. 7. Is. 2. P. 363–365.

17. *Lushnikov A., Kugurakova V., Nizamutdinov A.* Development of VR system to enhance understanding process of robot mechanisms // ICAROB 2018: Proceedings of the 2018 International Conference on Artificial Life and Robotics. 2018. P. 140–145.

18. *Kugurakova V.V., Abramov V.D., Sultanova R.R. et al.* Virtual Reality-Based Immersive Simulation for Invasive Surgery Training // *European Journal of Clinical Investigation*. 2018. Vol. 48. P. 224–225.

19. *Sultanova R., Sharaeva R.* Virtual reality-based immersive simulation mechanics for invasive surgery training // *Proceedings of International Conference on Developments in eSystems Engineering, DeSE*. 2019. Vol. October–2019. P. 924–928.

20. Программа для обучения аппендектонии в виртуальной реальности: Свидетельство о государственной регистрации программы для ЭВМ №2020613665 Российская Федерация / V.V. Kugurakova, M.R. Hafizov, V.D. Abramov, R.A. Sharaeva, R.R. Gazizov, T.M. Zinnurov, S.V. Zinchenko; заявитель и правообладатель Fed. gos. avtonom. obrazovat. uchrezhdenie vyssh. obraz. Kazanskij fed. un-t. №2020612602; заявл. 10.03.2020; зарегистрировано в реестре программ для ЭВМ 19.03.2020. 1 с.

21. Программа для обучения трахеостомии в виртуальной реальности: Свидетельство о государственной регистрации программы для ЭВМ № 2022661512 Российская Федерация / V.V. Kugurakova, A.P. Kiyasov, R.I. Fajzullin, S.V. Zinchenko, R.R. Galieva, R.R. Gazizov, N.R. Kurbangalieva, R.A. Sharaeva, N.E. Romanchuk, M.R. Hafizov; заявитель и правообладатель Fed. gos. avtonom. obrazovat. uchrezhdenie vyssh. obraz. Kazanskij fed. un-t. № 2022660504; заявл. от 08.06.2022; зарег. в реестре программ для ЭВМ 22.06.2022. 1 с.

22. *Mao R.Q., Lan L., Kay J., Lohre R., Ayeni O.R., Goel D.P. & SA D.D.* Immersive Virtual Reality for Surgical Training: A Systematic Review // *Journal of Surgical Research*. 2021. Vol. 268. C. 40–58.

23. *Smirnov A.A., Tatarkin V.V., Zahmatov I.G., Mar'yanovich A.T., Andreevskaya M.V., Nazmiev A.I., Kudlahmedov Sh.Sh., Rybakov V.A.* Trenazhyor dlya

osvoeniya manual'nyh navykov hirurgicheskikh vmeshatel'stv na mozgovom otdele cherepa // Kreativnaya hirurgiya i onkologiya. 2017. № 7(1). С. 63–68.

24. Analogov v Rossii net: permyaki pridumali trenazhyor, na kotorom vrachi nauchatsya delat' trepanaciyu cherepa // 59.RU. 2017.

URL: <https://59.ru/text/gorod/2017/08/14/50888691/> (access date: 01.09.2022).

25. *Cevallos N., Zukotyński B., GReig D., Silva M. and Thompson R.M.* The Utility of Virtual Reality in Orthopedic Surgical Training // Journal of Surgical Education. 2022. Vol. 79(6). P. 1516–1525.

26. Novyj interaktivnyj obrazovatel'nyj modul' s VR-trenazhyorom po otrabotke algoritma okazaniya neotlozhnoj hirurgicheskoy pomoshchi dostupen na Portale // Portal nepreryvnogo medicinskogo i farmacevticheskogo obrazovaniya Minzdrava Rossii, 2021. URL: <https://edu.rosminzdrav.ru/anonsy/anonsy/news/novyi-interaktivnyi-obrazovatelnyi-modul-s-vr-trenazh/> (access date: 01.09.2022).

27. Rostekh razrabotal virtual'nyj trenazhyor dlya obucheniya hirurgov // Rostekh, 2022. URL: <https://rostec.ru/news/rostekh-razrabotal-virtualnyy-trenazher-dlya-obucheniya-khirurgov/> (access date: 01.09.2022).

28. *Sadeghi A.H., Peek J.J., Max S.A., Smit L.L., Martina B.G., Rosalia R.A., Bakhuis W., Bogers A.J., Mahtab E.A.* Virtual Reality Simulation Training for Cardiopulmonary Resuscitation After Cardiac Surgery: Face and Content Validity Study // JMIR Serious Games. 2022. Vol. 10. No. 1. P. e30456.

29. *Atli K., Selman W., Ray A.* A comprehensive multicomponent neurosurgical course with use of virtual reality: modernizing the medical classroom // Journal of Surgical Education. 2020. Vol.78(4). P. 1350–1356.

30. *Bing E.G., Parham G.P., Cuevas A., et al.* Using low-cost virtual reality simulation to build surgical capacity for cervical cancer treatment // Journal Glob. Oncol. 2019. P. 1–7.

31. *Chaudhary A.H., Bukhari F., Iqbal W., Nawaz Z., Malik M.K.* Laparoscopic training exercises using HTC VIVE // Intell. Autom. Soft Co. 2020. Vol. 26. P. 53–59.

32. *Logishetty K., Gofton W.T., Rudran B., Beaulé P.E., Cobb J.P.* Fully immersive virtual reality for total hip arthroplasty: objective measurement of skills and transfer of visuospatial performance after a competency-based simulation curriculum // Journal Bone Joint Surg. Am. 2020. P. e27.

33. Luca A., Giorgino R., Gesualdo L., Peretti G.M., Belkhou A., Banfi G., et al. Innovative educational pathways in spine surgery: advanced virtual reality-based training // *World Neurosurg.* 2020. P. 674–680.

34. Arroyo-Berezowsky C., Jorba-Elguero P., Altamirano-Cruz M.A., Quinzaños-Fresnedo J. Usefulness of immersive virtual reality simulation during femoral nail application in an orthopedic fracture skills course // *Journal Musculoskelet. Surg. Res.* 2019. P. 326–333.

35. Hooper J., Tsiridis E., Feng J.E., et al. Virtual reality simulation facilitates resident training in total hip arthroplasty: a randomized controlled trial // *Journal Arthroplasty.* 2019. P. 2278–2283.

36. Logishetty K., Rudran B., Cobb J.P. Virtual reality training improves trainee performance in total hip arthroplasty: a randomized controlled trial // *Bone Joint Journal.* 2019. P. 1585–1592.

37. Lohre R., Bois A.J., Athwal G.S., Goel D.P. Canadian Shoulder and Elbow Society (CSES). Improved complex skill acquisition by immersive virtual reality training: a randomized controlled trial // *Journal Bone Joint Surg. Am.* 2020. P. e26.

38. Orland M.D., Patetta M.J., Wieser M., Kayupov E., Gonzalez M.H. Does virtual reality improve procedural completion and accuracy in an intramedullary tibial nail procedure? A randomized control trial // *Clin. Orthop. Relat. Res.* 2020. P. 2170–2177.

39. Praamsma M., Carnahan H., Backstein D., Veillette C.J., Gonzalez D., Dubrowski A. Drilling sounds are used by surgeons and intermediate residents, but not novice orthopedic trainees, to guide drilling motions // *Can. Journal Surg.* 2008. P. 442–446.

40. Sabbagh A.J., Bajunaid K.M., Alarifi N., et al. Roadmap for developing complex virtual reality simulation scenarios: subpial neurosurgical tumor resection model // *World Neurosurg.* 2020. P. e220–e229.

41. Comparing Unity vs Unreal for VR, MR or AR Development Projects // *XR Bootcamp*, 2022.

URL: <https://xrbootcamp.com/unity-vs-unreal-engine-for-xr-development/> (дата обращения: 01.09.2022).

42. Tsarouva M. Choosing an engine for VR: Unity vs Unreal // *iTechArt*, 2022. URL: <https://www.itechart.com/blog/unity-vs-unreal-virtual-reality/> (access date: 01.09.2022).

43. *Didenko G.* Oryol ili reshka: sravnenie Unity i Unreal Engine // DTF, 2017. URL: <https://dtf.ru/gamedev/7227-orel-ili-reshka-sravnenie-unity-i-unreal-engine> (access date: 01.09.2022).

44. Blend Shapes, Morph, Shape key // ShutterLine. URL: <https://3dyuriki.com/2010/05/02/blendshejpy-blend-shapes-i-morfy-morph-slovar/> (access date: 01.09.2022).

45. VR Template // Epic Games. URL: <https://docs.unrealengine.com/4.27/en-US/Resources/Templates/VRTemplate/> (access date: 01.09.2022).

46. Animation Notifications (Notifies) // Epic Games. URL: <https://docs.unrealengine.com/4.27/en-US/AnimatingObjects/SkeletalMeshAnimation/Sequences/Notifies/> (access date: 01.09.2022).

47. *Makkefri M.* Unreal Engine VR dlya razrabotchikov. Bombora, 2019. 256 p.

48. *Haj G.A.* *Assistirovanie pri hirurgicheskikh operaciyah* // Russia, Sankt-Petersburg, 1998. 382 p.

49. *Sharaeva R.A., Kugurakova V.V., Selezneva N.E.* Metodika uproshcheniya task-trekinga v proektah igrovoj industrii // Programmnye produkty i sistemy. 2022. T. 35. No. 3. S. 374–383.

50. *Sharaeva R.A., Kugurakova V.V.* Ocenka sokrashcheniya vremeni pri ispol'zovanii modificirovannoj metodiki task-trekinga v upravlenii IT-proektami // Programmnye sistemy: teoriya i prilozheniya. 2022. T. 13. №3(54). S. 307–324.

51. *Bowyer M.W., Streete K.A., Muniz G.M., Liu A.V.* Immersive virtual environments for medical training // Semin. Colon. Rectal. Surg. 2008. P. 90–97.

52. *Muhamethanov I.R., Hafizov M.R., Shubin A.V.* Sravnenie klient-servernyh reshenij pri razrabotke mnogopol'zovatel'skih onlajn-igr na Unity // Elektronnye biblioteki. 2022. No. 5.

53. *Carter F.J., Schijven M.P., Aggarwal R., et al.* Consensus guidelines for validation of virtual reality surgical simulators // Simul. Healthc. 2006. P. 171–179.

54. *Mirchi N., Bissonnette V., Yilmaz R., Ledwos N., Winkler-Schwartz A., Del Maestro R.F.* The Virtual Operative Assistant: An explainable artificial intelligence tool for simulation-based training in surgery and medicine // PLoS ONE. 2020. Vol. 15(2): e0229596. P. 1–15.

55. *Sawaya R., Bugdadi A., Azarnoush H., Winkler-Schwartz A., Alotaibi F.E., Bajunaid K., et al.* Virtual Reality Tumor Resection: The Force Pyramid Approach // *Operative Neurosurgery*. 2018. P. 686–696.

56. *Wagner C.R., Stylopoulos N., Jackson P.G., Howe R.D.* The benefit of force feedback in surgery: Examination of blunt dissection // *Presence: teleoperators and virtual environments*. 2007. P. 252–262.

57. *Ladha L., Deepa T.* Feature selection methods and algorithms // *International journal on computer science and engineering*. 2011. P. 1787–1797.

58. *Lli J., Cheng K., Wang S., Morstatter F., Trevino R.P., Tang J., et al.* Feature selection: A data perspective // *ACM Computing Surveys (CSUR)*. 2018. P. 1–45.

59. *Winkler-Schwartz A., Yilmaz R., Mirchi N., Bissonnette V., Ledwos N., Siyar S., et al.* Assessment of Machine Learning Identification of Surgical Operative Factors Associated with Surgical Expertise in Virtual Reality Simulation // *JAMA Network Open*. 2019. P. e198363–e198363.

60. *Yu L., Liu H.* Efficient feature selection via analysis of relevance and redundancy // *Journal of Machine Learning Research*. 2004. P. 1205–1224.

61. *Wang X., Wang Y., Wang L.* Improving fuzzy c-means clustering based on feature-weight learning // *Pattern Recognition Letters*. 2004. P. 1123–1132.

62. *Ryu W.H.A., Chan S., Sutherland G.R.* Supplementary educational models in Canadian neurosurgery residency programs // *Canadian Journal of Neurological Sciences*. 2017. P. 177–183.

63. *Sweller J.* Cognitive load theory, learning difficulty, and instructional design // *Learning and Instruction*. 1994. P. 295–312.

64. *Gobet F., Lane P.C., Croker S., Cheng M.P.C., Jones G., Oliver I., et al.* Chunking mechanisms in human learning // *Trends in Cognitive Sciences*. 2001. P. 236–243.

65. *Williamson B.* Digital education governance: data visualization, predictive analytics, and ‘real-time’ policy instruments // *Journal of Education Policy*. 2016. P. 123–141.

66. *Saplacan D., Herstad J., Pajalic Z.* Feedback from Digital Systems Used in Higher Education: An Inquiry into Triggered Emotions—Two Universal Design Oriented Solutions for a Better User Experience // *Studies in Health Technology and Informatics*. 2018. P. 421–430.



67. *Kugurakova V.V.* Matematicheskoe i programmnoe obespechenie mnogopol'zovatel'skih trenazhyorov s pogruzheniem v immersivnye virtual'nye sredy: dis. kand. tekhn. nauk: 05.13.11. Kazanskij federal'nyj universitet, 2018. 187 p.

68. *Akhmetsharipov R.D., Khafizov M.R., Lushnikov A.Yu., Zigantdinov Sh.Ya.* The soft tissue implementation with triangulated mesh for virtual surgery system // *Smart Innovation, Systems and Technologies*. 2018. P. 163–167.

69. *Shigapov M.I., Kugurakova V.V.* Design and development of a hardware and software system for simulation of feedback tactility // *Proceedings SIBCON*. 2021. P. 1–6.

70. *Shigapov M.I., Kugurakova V.V., Zykov E.Yu.* Design of digital gloves with feedback for VR // *Proceedings IEEE EWDTs*. 2018. P. 1–5.

71. *Aglyamov F.R., Kugurakov V.S.* Trenazhyor dlya reabilitacii pacientov s problemami mobil'nosti ruki, postroennyj s ispol'zovaniem tekhnologij virtual'noj real'nosti // *Programmnye produkty i sistemy*. 2022. T. 35. No. 3. S. 285–292.

72. *Friston S., Griffith E., Swapp D., Marshall A., Steed A.* Position-based control of under-constrained haptics: A system for the Dexmo glove // *IEEE Robot Autom.* [Internet]. 2019. P. 3497–3504.

73. *Bhalla S., Beegun I., Awad Z., Tolley N.* Simulation-based ENT induction: validation of a novel mannequin training model // *Journal Laryngol. Otol.* 2020. P. 74–80.

74. *McLaren O., Perkins C., Alderson D.* The effect of surgical complications on ENT trainees // *Journal Laryngol. Otol.* 2021. P. 293–296.

75. *Cherry J., Weir R.* Medicolegal and ethical aspects of ORL-HNS in the new millennium // *Journal Laryngol. Otol.* 2006. P. 737–740.

76. *Mayer A.W., Smith K.A., Carrie S.* A systematic review of factors affecting choice of otolaryngology as a career in medical students and junior doctors // *Journal Laryngol. Otol.* 2019. P. 836–842.

77. *Bhutta M., Mandavia R., Syed I., Qureshi A., Hettige R., Wong B.Y.W. et al.* A survey of how and why medical students and junior doctors choose a career in ENT surgery // *Journal Laryngol. Otol.* 2016. P. 1054–1058.

## СВЕДЕНИЯ ОБ АВТОРАХ



**ШАРАЕВА Регина Айратовна** – инженер НИЛ Разработка интеллектуальных инструментов для компьютерных игр Института ИТИС КФУ. Сфера научных интересов – менеджмент проектов игровой индустрии.

**Regina Airatovna SHARAEVA** – Engineer of Laboratory for Development of Intelligent Tools for Computer Games, Institute ITIS KFU. Research interests include game industry project management.

e-mail: r.sharaeva3496@gmail.com

ORCID: 0000-0002-2359-1873



**КУГУРАКОВА Влада Владимировна** – к. т. н., доцент кафедры программной инженерии Института ИТИС КФУ, руководитель НИЛ разработки AR/VR приложений и компьютерных игр. Сфера научных интересов – иммерсивность виртуальных сред, проблемы генерации реалистичной визуализации, различные аспекты проектирования игр, AR/VR, подходы к интерпретации UX.

**Vlada Vladimirovna KUGURAKOVA**, PhD., Docent of the Institute ITIS KFU, Head of Laboratory «AR/VR applications and Gamedev». Research interests include immersiveness of virtual environments, problems of generating realistic visualization, various aspects of game design, AR/VR, approaches to UX interpretation.

email: vlada.kugurakova@gmail.com

ORCID: 0000-0002-1552-4910



**ГАЛИЕВА Раилина Рафаэлевна** – лаборант-исследователь НИЛ SIM – Лаборатории симуляционных технологий в биомедицине Института ИТИС КФУ. Сфера научных интересов – разработка AR/VR проектов на движке Unreal Engine.

**Railina Rafaelevna GALIEVA** – research assistant of SIM – Lab. of simulation technologies in biomedicine, Institute ITIS KFU. Research interests include development of AR/VR projects on the Unreal Engine.

email: galievarailina@gmail.com

ORCID: 0000-0002-6468-2069



***ЗИНЧЕНКО Сергей Викторович*** – д. м. н., профессор, заведующий кафедрой хирургии, акушерства и гинекологии ИФМиБ КФУ, врач-онколог Медико-санитарной части Научно-клинического центра прецизионной и регенеративной медицины КФУ, врач-хирург Медико-санитарной части КФУ, заместитель главного врача по медицинской части, руководитель хирургического направления.

***Sergey Viktorovich ZINCHENKO*** – MD, Professor, Head of the Department of Surgery, Obstetrics and Gynecology KFU, Medical Oncologist at the KFU Scientific and Clinical Center for Precision and Regenerative Medicine, Surgeon at KFU Medical and Clinical Center, Deputy Head Physician for Medical Services, Head of the Surgical Department.

email: zinchenkos.v@mail.ru

ORCID: 0000-0002-9306-3507

*Материал поступил в редакцию 12 сентября 2022 года*

## ОБЗОР ПРАКТИК УПРАВЛЕНИЯ ПРОЕКТАМИ В ИГРОВОЙ РАЗРАБОТКЕ

А. В. Шубин<sup>1</sup> [0000-0002-6203-3268], Г. Ф. Сахибгареева<sup>2</sup> [0000-0003-4673-3253],

В. В. Кугуракова<sup>3</sup> [0000-0002-1552-4910]

<sup>1,2,3</sup>*Институт информационных технологий и интеллектуальных систем  
Казанского (Приволжского) федерального университета;*

<sup>1</sup>shubin.aleksey.kpfu@gmail.com, <sup>2</sup>gulnara.sahibgareeva42@gmail.com,

<sup>3</sup>vlada.kugurakova@gmail.com

### ***Аннотация***

Опыт игровых студий показывает, что классические методологии разработки программного обеспечения (ПО) плохо имплементируются в видеоигровую разработку из-за ее интерактивной составляющей, связанной с правильным выстраиванием обратной связи между игрой и пользователем. Кроме того, разработка видеоигр объединяет большое число разработчиков различных сфер, деятельность которых обязана быть согласована в проекте. Несмотря на эти отличия, видеоигры, как и любое другое разрабатываемое ПО, нуждаются в процессе организации команды разработки.

В статье приведён обзор традиционных методологий разработки программного обеспечения, а также модификаций, специализирующихся конкретно на разработке видеоигр. Проведено сравнение наиболее популярных методологий и определено качество их имплементации в студиях разработки видеоигр.

***Ключевые слова:*** видеоигра, программная инженерия, игровой дизайн.

### **ВВЕДЕНИЕ**

Развлечения – одна из важнейших частей жизни любого человека: каждый находит свой способ разнообразить свой досуг, отдохнуть, получить новый опыт. Вместе с технологическим прогрессом у человечества появляются новые виды развлечений, такие как книги, кино, аттракционы, видеоигры. На данный момент насчитывается более трёх миллиардов людей, когда-либо приобретших видеоигру, что может говорить об огромной популярности данного вида развлечений,

особенно учитывая, насколько мало времени прошло с момента распространения видеоигр в массы [1].

Видеоигра – это программа, обеспечивающая взаимодействие игроков с игрой и между собой по определенным правилам, а также способное выводить информацию о состоянии игры и реагировать в ответ на действия игроков, которые осуществляются с помощью контроллера.

Видеоигры с самого своего появления не прекращают развиваться, причём происходит это не только в технической части (технологии, компьютерная графика, внешний вид и количество объектов в игре), но и в аспекте игрового процесса (уровень погружения, интерактивность, достоверность). Современные игры давно отошли от концепции «одна игра – одно действие» (Tetris, Pong, Space Invaders), когда весь игровой процесс можно было описать двумя-тремя правилами. Вместо этого игры стали более комплексными, совмещают в себе множество механик или различные типы игрового процесса, более грамотно и осознанно вовлекая игрока.

Созданием игрового процесса в современных играх занимаются такие специалисты, как геймдизайнеры. Они создают правила игры и продумывают логику создания внутриигрового контента. Зачастую в больших компаниях работают целые команды игровых дизайнеров, которые отличаются различной спецификой работы: повествовательная часть игры, внутриигровые локации, правила и системы видеоигры и другие.

Наиболее значимый вопрос при планировании разработки игр – это составление представления о том, какой объем составляет будущий проект. Для того чтобы ответить на этот вопрос, необходимо обратиться к специалистам в сфере разработки игр.

Джесси Шелл, автор «библии» игровой разработки, выделил четыре основных аспекта видеоигры [2]:

Технология – игровой движок, технические характеристики ПО.

История – сюжет, персонажи, цель.

Механика – игровые правила, геймплей.

Эстетика – внешний вид, общее настроение.

Каждый из данных аспектов является ключевым при создании хорошей видеоигры, поэтому при ее разработке геймдизайнер обязан уделить достаточно

внимания каждому из них. Поэтому стоит рассмотреть эти компоненты более подробно.

*Технологии* – «кости» любой видеоигры. Чаще всего выбор движка будет зависеть непосредственно от типа игрового процесса и характеристик игровых устройств целевой аудитории. Несмотря на это, можно найти и универсальные решения для создания игровых миров, совместимых с большинством игровых устройств, к тому же такие решения будут поддерживать более простой перенос между различными игровыми платформами.

*История* – это связь между всеми вышеперечисленными аспектами видеоигры. История помогает развивать персонаж игры, даёт ему конкретную, обоснованную цель действий, позволяет игроку понять тему игры. В свою очередь хорошая тема игры вызывает у игрока более сильные чувства, позволяет проникнуться игрой и способствует погружению.

*Игровые механики* – это «мясо», которое наращивается в движке. Основой любой игры являются правила и возможности игрока в виртуальном мире, и именно это, в свою очередь, определяется реализованными механиками.

*Эстетика* – это «кожа», внешний вид игры. Благодаря однородной, грамотно выстроенной эстетике игроку можно передать нужное настроение, атмосферу игры и повествования.

Каждый из аспектов, перечисленных выше, хоть и важен для любой видеоигры, но, как показывает опыт, не важно глубоко прорабатывать одновременно все аспекты разом. К примеру, всем известный Tetris – довольно простая по своей сути игра, она не обладает глубокой историей, перипетиями, сюжетными поворотами, в ранних версиях не имела продвинутой графики. При этом игра понравилась многим за счет вовлекающего и репетитивного геймплея, который ставит перед игроком непрекращающийся вызов.

Вовлечение – это состояние игрока, при котором он остаётся заинтересованным продолжать процесс игры на долгое время. Данное понятие больше относится непосредственно к игровому процессу, так как требует активных действий со стороны игрока. Из наиболее известных примеров вовлекающих игр можно назвать игры жанра «три в ряд», медитативные игры вроде Journey, а из актуальных на данный момент – Vampire Survivor. И всё это благодаря репетитивному, несложному

игровому процессу в связке с получением награды в виде нового контента или части истории.

Погружение – это, в свою очередь, состояние человека, когда он обособлен от физического мира и полностью погружается в выдуманный мир или историю. Поэтому погружение скорее относится к сюжету, передаваемому игрой. Данное состояние можно зачастую наблюдать у людей, читающих захватывающую книгу или смотрящих сериал. В таком состоянии человек попросту не может оторваться от процесса раскрытия сюжета и, сделав перерыв, продолжает думать и планировать что-то в рамках этой книги или сериала.

Названные два понятия имеют различную природу и позволяют игроку задержаться в игре на более долгий промежуток времени, а также способствуют тому, что игрок возвращается и продолжает получать удовольствие от процесса игры.

### **АНАЛИЗ СУЩЕСТВУЮЩИХ ИГРОВЫХ ПРОДУКТОВ**

Для разработки сценария существуют инструменты, призванные облегчить структурирование сюжета. Но средств, способных автоматизировать процесс создания структуры сценария, еще нет. Раннее прототипирование возможно только после создания структуры вручную в программах, которые представляют сюжет в виде связанных карточек. Прототипирование предполагает быстрый результат без лишних затрат.

Так как современная наука создания видеоигр до сих пор не может похвастаться чёткими правилами создания идеальной игры, то лучшим вариантом на данный момент является использование лучших практик – правил, открытых опытным путём предыдущими геймдизайнерами. Эти правила сегодня являются неполными, вероятно, даже ненаучными, хотя и помогают молодым разработчикам видеоигр добиться успеха. Такой подход сравнивают с примитивными методами алхимиков до появления Менделеева и, следом, современной химии [2].

Ниже для подбора хороших игровых решений проведен анализ нескольких популярных игр в жанре «приключения». По результатам выделены ключевые аспекты игры, которые делают её более привлекательной для игроков.

Видеоигры категории AAA (например, *God of War*, *Detroit: Become Human*, *Dying Light 2* и другие) созданы в крупных именитых студиях по разработке игр и

---

имеют солидный бюджет. Такие игры могут похвастаться крупной командой разработки, проработанным сюжетом, качественными графикой и анимацией, а также выдержанным геймплеем.

Для всех высокобюджетных игр характерна одна общая деталь – современная реалистичная графика (см., например, рис. 1). Зачастую эти игры отличаются соотношением геймплей–сюжет, то есть тем, на какой аспект был сделан больший упор: отзывчивость на действия игрока и свободу действий или на кинематографичность, глубину сюжета и подачу истории. И если одни игры могут похвастаться интересной и разветвлённой историей, но при этом оставляют игроку лишь выбор реплик, то другие имеют тривиальный сюжет, но предоставляют игроку огромную свободу действий, а также содержат вызов игроку, принятие которого требует понимания игры.

Ни одна игра не может похвастаться глубокой проработкой всех ее аспектов, так как в таком случае объём работы будет возрастать в геометрической прогрессии, и велик шанс несогласованности отдельных элементов игры либо вовсе растяжения ее разработки по времени на долгие годы. Именно поэтому зачастую игровые студии стараются найти баланс в этом вопросе и отказываются от проработки отдельных компонентов игры [3].

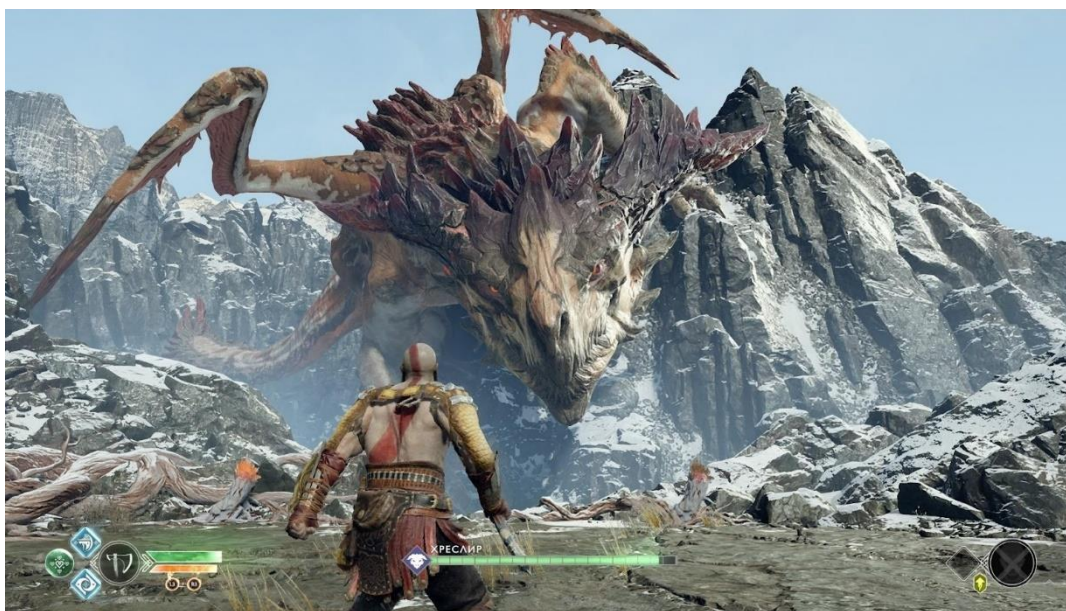


Рис. 1. Изображение игрового процесса видеоигры God of War (2018)



Взглянув на список известных игр, кроме высокобюджетных, можно заметить, что часто на первых строчках списков популярных игр стоят игры жанра «песочница» (Valheim, Raft, Rust, Project Zomboid и другие). Зачастую разработка таких видеоигр начиналась в маленьких командах либо осуществлялась разработчиками-одиночками, но после обретения популярности команда разработки выросла до крупной студии.

Инди-игры зачастую строятся на одной геймплейной механике, которая развивается по мере прохождения игры, поэтому остальные аспекты, такие как графика, музыка, сюжет, менее развиты (см., например, рис. 2) либо отброшены вовсе в пользу отполированного игрового процесса.



Рис. 2. Изображение игрового процесса игры Project Zomboid

Если сильнее углубляться в жанр «песочница», то можно заметить, что большая часть игр данного жанра, как и говорилось выше, не имеет глубокого сюжета (Valheim, Raft) или не имеет его вообще (Project Zomboid, Rust), в отличие от упомянутых AAA-игр, но это не мешает им быть популярными среди игроков.

Взглянув на визуальную составляющую, можно заметить, что названные игры не могут похвастаться реалистичной и современной графикой (Project Zomboid), но имеют довольно большую постоянную аудиторию игроков. Так

можно убедиться, что современная графика не является обязательным фактором при создании хорошей видеоигры. Дело в том, что хотя игры названного типа не могут похвастаться большими бюджетами, но при этом, уделив достаточное внимание лишь проработке игрового процесса, малые команды разработки также способны создать игры, не уступающие по популярности играм больших игровых студий.

Так как картинкой и сюжетом данные игры похвастаться не могут, то остаётся последний вариант – непосредственно игровой процесс, и есть то, что цепляет игрока. Но, что самое важное, – это поддержка игрой создания собственной мотивации у игрока, когда цели себе ставит он сам. Такая опция появляется лишь при большой степени свободы действий у игрока, когда у него появляется возможность принимать решения самостоятельно, а не просто идти по указке игры. В данном случае игрок начинает самостоятельно ставить себе цели для исследования мира, составляет план действий, решает какие-либо вопросы, распределяет ресурсы, в конечном итоге достигает максимального комфорта при нахождении в мире игры и, следовательно, самостоятельно создает себе приключение под свой вкус.

Такого игрового процесса сложно добиться простыми планированием и аналитикой. Именно поэтому разработка хорошей видеоигры всегда подразумевает большое количество игровых тестов, корректировок и повторной саморефлексии, а разработчики создают собственные инструменты для оптимизации и балансировки игрового процесса [4, 5].

### **ОБЗОР МЕТОДОВ РАЗРАБОТКИ ВИДЕОИГР**

Разработка игр, как и любое другое производство программных продуктов, обязано использовать современные методологии и практики для эффективной организации труда и сокращения рисков. Крупные компании, большие проекты и сжатые сроки – всё это также актуально для современной разработки игр.

Несмотря на то, что использование современных методологий в большей степени актуально для больших команд, разработчики-одиночки и маленькие инди-студии также используют различные методы и инструменты для протоколирования, прототипирования и разработки. При этом использование методов про-

граммной инженерии в маленьких командах позволяет отслеживать и оптимизировать деятельность, что помогает правильно распределять ресурсы, избавиться от переработок, вести команду разработки в одном направлении, что в конечном итоге позволит создать более качественный продукт.

### **Каскадная модель**

Эта модель развития программного обеспечения использовалась в производстве ранних видеоигр, когда вся разработка была распланирована заранее и каждый этап строго следовал за предыдущим.

Как было сказано ранее, такая модель подразумевает, что каждый этап разработки должен быть полностью закончен, после чего можно переходить к следующему шагу. Данная модель подходит в том случае, если ваша игра распланирована до мельчайших деталей, и по мере разработки её содержание не будет изменяться. Но это и является основным минусом, так как зачастую некоторые решения относительно наполнения игры могут быть спорными или вовсе не подходить игре, а глобальные изменения будут подразумевать полный перезапуск проекта, что невозможно в современных реалиях.

### **Спиральная модель**

Эта модель, в отличие от каскадной, является более подходящей при разработке проектов с изменяющимися на ходу или нечёткими требованиями, так как в отличие от предыдущей модели здесь разработка происходит через несколько повторяющихся этапов (итераций), в конечном итоге дополняя дизайн приложения для конечной разработки (см. рис. 3).

Данная модель, как говорилось ранее, способна легко адаптироваться к изменениям, тем самым она уменьшает риски полной перезагрузки проекта. Также одним из плюсов будет более раннее создание рабочего прототипа, что позволяет видеть прогресс наглядно, а также проверять будущие изменения на основе уже созданного прототипа. К тому же данная модель позволяет подключать всю команду разработки на ранних этапах, а не только лишь по завершению концептуальной разработки. Из минусов можно назвать значительные переработки по сравнению с каскадной моделью, так как в конечном итоге те же самые этапы приходится проходить ни один раз по мере разработки проекта.

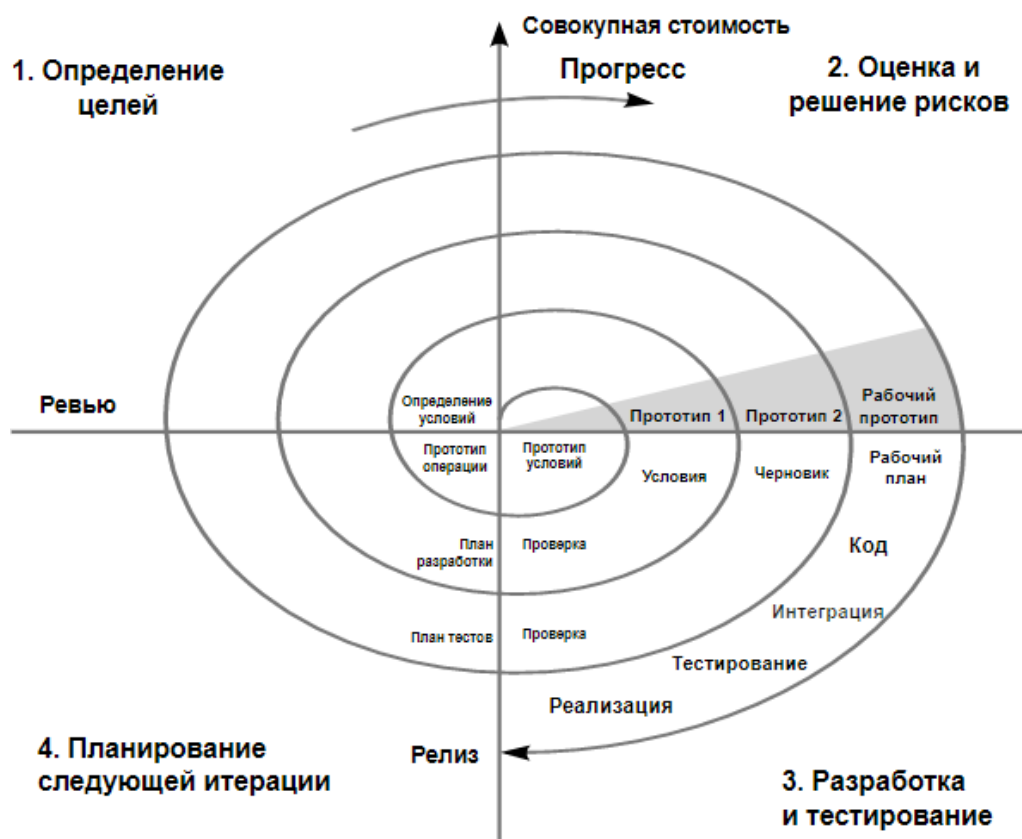


Рис. 3. Схема работы спиральной модели

### Итеративная модель

Итеративная модель – другая итерационная модель, но в отличие от спиральной, где большинство требований уже известно и происходит их уточнение, здесь основной упор делается на разработке усовершенствованной версии продукта, после чего подбираются новые требования к следующей итерации.

Данную модель целесообразно использовать в больших проектах, где большие переработки могут значительно замедлить выпуск конечного продукта. Также эта модель не требует полной спецификации требований – они могут уточняться по мере разработки новой версии продукта. Также такая модель позволяет дополнять концепт новыми требованиями, а конфликты между требованиями выявляются раньше.

### V-модель

Эта модель является дополнением каскадной модели и отличается от нее наличием параллельного тестирования каждого этапа разработки (см. рис. 4). V-

модель так же, как водопад, является довольно простой и требует полного выполнения предыдущего этапа, но при этом фаза тестирования позволяет скорректировать выполнение каждого этапа. Но, как и в случае с каскадной моделью, остаётся вероятным наличие ошибок на первых этапах разработки. Таким образом, при наличии несоответствия требований их проверка будет происходить слишком поздно, когда сам проект будет уже реализован, что может привести к снижению качества продукта либо, подобно каскадной модели, к перезапуску проекта.

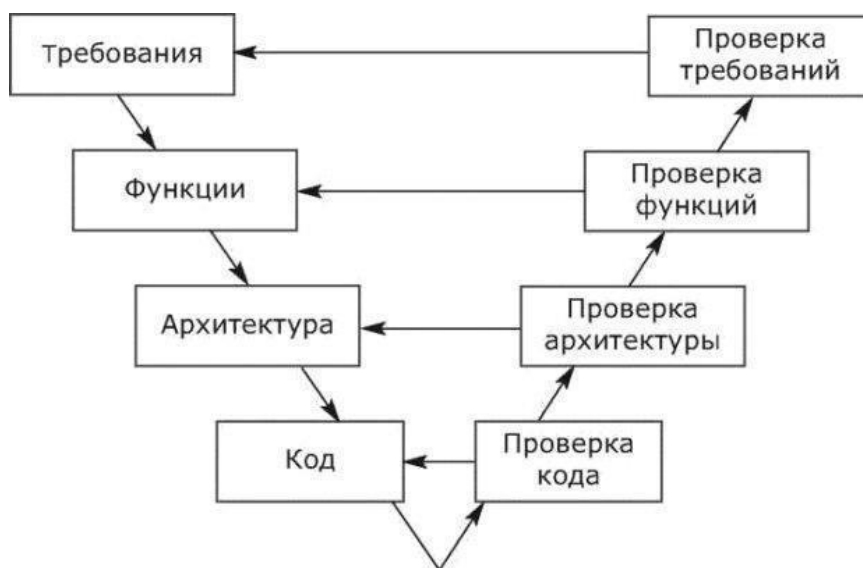


Рис. 4. Схема работы V-модели

### Модель большого взрыва

Эту модель сложно назвать методологией как таковой, так как она подразумевает естественную эволюцию проекта, без использования каких-либо практик [6]. Большая часть ресурсов здесь направляется на развитие. Эта методология обычно используется для небольших проектов, когда над разработкой игры работают всего два-три разработчика. При этом такая модель разработки не требует особых знаний и комфортна для малой команды.

Ключевой проблемой данной модели является отсутствие чёткого плана, что может привести к растянутым срокам разработки и долгому поиску решений появляющихся проблем.

### **Адаптации итерационных моделей**

Итерационные модели являются наиболее популярным выбором среди компаний – игровых разработчиков. Используемые ими методологии являются адаптацией классических моделей, модифицированных с целью достижения качественного продукта за счёт более глубокого анализа и увеличения сроков разработки и тестирования видеоигры [7]. Примером можно назвать тестирование потенциальными пользователями или так называемый «ранний доступ», когда игроки могут приобрести доступ к незаконченной версии игры, которая будет исправляться и дополняться в зависимости от мнения этих пользователей. Также применяется такая практика, как этапы Альфа- и Бета-тестирования, когда на первом этапе доступ к ранней версии игры предоставляется закрытому кругу пользователей, заинтересованных в этом продукте и его качестве, а на втором этапе – всем желающим. Следовательно, компании, практикующие данные практики, добавляют такие этапы в свои методологии. Примером служит Blitz Games Studio, где использованы:

Pitch – этап инициации разработки и появления игровой концепции;

Pre-production – создание документации, концепт-артов и развитие игровой концепции;

Main production – непосредственная реализация проекта на игровом движке;

Alpha – тестирование продукта ограниченной группой игроков и исправление наиболее критических ошибок;

Beta – предоставление доступа всем желающим пользователям и итоговая полировка качества продукта;

Master – выпуск игры в полноценную эксплуатацию, конец разработки.

Судя по большому числу вариаций, подобные итерационные модели являются одним из наиболее актуальных выборов при планировании и организации разработки в больших видеоигровых студиях.

### **Agile-методология**

Agile-практика пришла на смену более популярным в своё время методам, таким, например, как водопад или V-модель, и описывает скорее принципы орга-

низации работы команды разработки и практику тесного общения между разработчиками друг с другом и с заказчиком; приоритеты нацелены на удовлетворение клиента, использование методологий с возможностью изменения требований (итерационные подходы), а также саморефлексию [8] – поиск способов быть более эффективными для определения своего поведения в будущем.

На основе принципов Agile были созданы такие методологии, как экстремальное программирование, DSDM, Scrum, FDD и другие [9, 10]. Именно гибкие методологии разработки на данный момент являются наиболее популярными в компаниях при создании любых программных продуктов, так как они позволяют минимизировать риски, вовремя выявлять конфликт требований, оптимизировать работу команды разработки и в перспективе наращивать эффективность их деятельности.

### Адаптация Agile под игровую разработку

Одной из интерпретаций Agile-методологии является sdPP (Software Development Project Pattern), адаптированная под создание GDD и разработку видеоигр в целом [11].

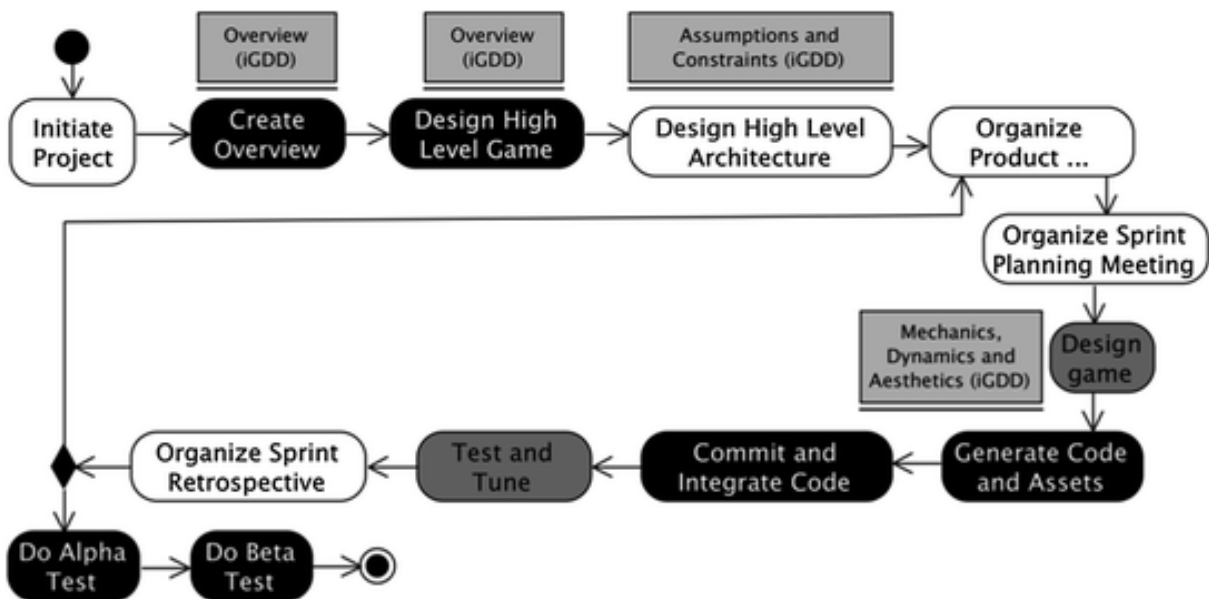


Рис. 5. Схема работы sdPP-модели, адаптированной под разработку видеоигр

На рис. 5 чёрными блоками обозначены новые блоки, относящиеся к разработке видеоигр, а тёмно-серыми – преобразованные блоки стандартной sdPP-модели, также светло-серыми обозначены описательные блоки для некоторых этапов. Ниже представлено описание каждого этапа данной методологии.

CREATE OVERVIEW – этап описания основных аспектов игры: цели, жанра, ответы на различные вопросы (стоит ли она того, какова целевая аудитория, основные механики).

DESIGN HIGH LEVEL GAME – определение главных особенностей игры: game modalities (одиночная игра, многопользовательская игра, игра через интернет, аркадный режим, сюжетный режим), поддерживаемые платформы, тема игры, история и размер планируемого проекта.

DESIGN HIGH LEVEL GAME ARCHITECTURE – описание технической составляющей для определения допущений, ограничений, которые могут появиться в игре. Техническая составляющая содержит: стандарты, соглашения, технологии, ресурсы и архитектуру, выбранную для игры.

DESIGN GAME – разработка главного героя, его действий, эстетики игры, а также включенных игровых механик.

GENERATE CODE AND ASSETS – написание кода и подготовка ассетов (музыка, внутриигровые скрипты, трёхмерные модели, анимации и т. д.)

COMMIT AND INTEGRATE CODE – разработка игры по принципу «получить в любой момент» до версии, которую уже можно было предоставить игрокам.

DO TEST AND TUNE – проверка результатов спринта. Маленькие корректировки могут отполировать игру, но большие изменения остаются на следующий спринт. В конечном итоге команда должна прийти до продукта, который можно будет продать.

DO ALPHA AND BETA TESTS – устранение багов на альфа-тесте, чтобы на бета-тесте игра была проходимой для отслеживания опыта игроков.

Данная модификация была специально спроектирована, чтобы конкретно использовать её при разработке видеоигр с целью уменьшения количества переделок в команде.



В статье [11] описан эксперимент, в котором две сбалансированные команды молодых разработчиков использовали различные подходы при создании видеоигры:

Группа А – использование адаптированной sdPP-модели.

Группа Б – применение стандартной Agile-методологии.

Разработка проектов проходила в течение трёх месяцев, где команды обязаны были реализовать от 10 до 15 функций за 4–5 итераций.

Описанные результаты эксперимента подтверждают, что среднее значение переработок в группе Б превышало более чем в четыре раза среднее количество переработок в группе А.

### **ЗАКЛЮЧЕНИЕ**

Разработка видеоигр – молодая индустрия, основой которой является использование так называемых «лучших практик», когда новые проекты зачастую делают на примере более удачных аналогов. Использование современных методологий разработки каноничного программного обеспечения будет менее эффективным, так как специфика разработки видеоигр делает упор именно на интерактивную составляющую, которую сложно описать и полностью промоделировать на этапе предпродакшена. Сложность в прогнозировании качества принятого решения диктует разработчикам, что необходимо сначала реализовать идею, а затем делать выводы о том, насколько удачно та или иная механика передаёт смысл и синергирует с другими, уже реализованными аспектами игры. Хотя в разработке инди-проектов нередко бывают случаи, когда разработчики внедряют совершенно экспериментальные решения, которые впоследствии становятся удачными среди игроков, такой вариант невозможен для большой команды разработчиков, нуждающейся в стабильном успехе.

Отмеченный аспект обусловлен по большому счёту комплексностью интерактивной составляющей игры: связь игровых механик с повествовательной частью, планомерное развитие механик по мере прохождения, учёт начального уровня подготовки игрока, а также неоднократные тесты жизнеспособности выбранного решения (из-за отсутствия универсального решения). Кроме того, важно уделить должное внимание каждому из четырёх аспектов игры (технология, история, меха-

ника, эстетика), так как недостаточное развитие какого-либо из этих аспектов может привести игрока к диссонансу. Также важен анализ большого количества конкурентных проектов, чтобы на их примере реализовывать собственные идеи.

В заключение можно сказать, что даже каноничные методологии разработки программного обеспечения «в чистом виде» хотя и помогают намного лучше организовать работу, но из-за специфики предметной области требуют дополнительной детализации при разработке. Используя методологии и инструменты, адаптированные под игровую индустрию [12], команда сокращает количество переработок, увеличивает детализацию и качество дизайна проекта, следовательно, качество итогового продукта.

### **БЛАГОДАРНОСТИ**

Работа выполнена за счет средств Программы стратегического академического лидерства Казанского (Приволжского) федерального университета («ПРИОРИТЕТ-2030»).

### **СПИСОК ЛИТЕРАТУРЫ**

1. Global video game consumer population passes 3 billion. URL: <https://www.dfcint.com/dossier/global-video-game-consumer-population/> (дата обращения: 20.04.2022).
2. *Шелл Дж.* Геймдизайн. Как создать игру, в которую будут играть все. М.: Альпина Паблишер, 2022. 640 с.
3. Game Project Management and Business Context. URL: <https://www.studytonight.com/3d-game-engineering-with-unity/business-context-of-game> (дата обращения: 23.05.2022).
4. *Сахибгареева Г.Ф., Кугуракова В.В.* Практики балансирования компьютерных игр // Программные системы: теория и приложения. 2022. Т. 13. №3(54). С. 255–273.
5. *Сахибгареева Г.Ф., Кугуракова В.В., Большаков Э.С.* Генерация и балансирование игровых механик видеоигр // Научный сервис в сети Интернет: труды XXIV Всероссийской научной конференции (19–22 сентября 2022 г., онлайн). М.: ИПМ им. М.В. Келдыша, 2022. С. 455–485.
6. Game Development Lifecycle Models.

URL: <https://www.studytonight.com/3d-game-engineering-with-unity/game-development-models> (дата обращения: 12.05.2022).

7. *Aleem S., Capretz L.F., Ahmed F.* Game development software engineering process life cycle: a systematic review // *Journal of Software Engineering Research and Development*. 2016. Vol. 4, No. 6. <https://doi.org/10.1186/s40411-016-0032-7>.

8. Manifesto for Agile Software Development. URL: <http://agilemanifesto.org> (дата обращения: 12.05.2022).

9. Ещё раз про семь основных методологий разработки.  
URL: <https://habr.com/ru/company/edison/blog/269789/> (дата обращения: 15.05.2022).

10. Популярные фреймворки для гибкого управления проектами. URL: <https://www.wrike.com/ru/project-management-guide/agile-frejmvorki-v-upravlenii-proektami/> (дата обращения: 20.05.2022).

11. *Mitre-Hernández H.A., Lara-Alvarez C., González-Salazar M., Martín D.* Decreasing Rework in Video Games Development from a Software Engineering Perspective // *Trends and Applications in Software Engineering. Advances in Intelligent Systems and Computing*, 2016. Vol. 405. P. 295–304.  
[https://doi.org/10.1007/978-3-319-26285-7\\_25](https://doi.org/10.1007/978-3-319-26285-7_25).

12. *Шараева Р.А., Кузуракова В.В., Селезнева Н.Э.* Методика упрощения таск-трекинга в проектах игровой индустрии // *Программные продукты и системы*. 2022. Т. 35. № 3. С. 374–383. <https://doi.org/10.15827/0236-235X.139.374-383>.

## THE CONCEPT OF AUTOMATIC CREATION TOOL FOR COMPUTER GAME SCENARIO PROTOTYPE

A. V. Shubin<sup>[0000-0002-6203-3268]</sup>, G. F. Sahibgareeva<sup>2 [0000-0003-4673-3253]</sup>,

V. V. Kugurakova<sup>3 [0000-0002-1552-4910]</sup>

<sup>1,2,3</sup> *Institute of Information Technologies and Intelligent Systems, Kazan (Volga Region) Federal University, ul. Kremlyovskaya, 35, Kazan, 420008*

<sup>1</sup>shubin.aleksey.kpfu@gmail.com, <sup>2</sup>gulnara.sahibgareeva42@gmail.com,

<sup>3</sup>vlada.kugurakova@gmail.com

### Abstract

The experience of game studios shows that classical methodologies of software development are poorly implemented in video game development because of the interactive component of this area, related to the correct creation of feedback between the game and the user. In addition, video game development involves a large number of developers from different areas, whose activities must be coordinated in the project.

Despite these differences, video games, like any other developed software, need a development team organization process. In this article we reviewed traditional software development methodologies, as well as modifications specializing specifically in video game development. The most popular methodologies were compared and the quality of their implementation in video game development studios was determined.

**Keywords:** *video game, software engineering, game design.*

### REFERENCES

1. Global video game consumer population passes 3 billion.  
URL: <https://www.dfcint.com/dossier/global-video-game-consumer-population/>, last accessed 2022/04/20.
2. Schell J. N. The Art of Game Design: A Book of Lenses, Second Edition. CRC Press. 2014. 600 p.
3. Game Project Management and Business Context. URL: <https://www.studytonight.com/3d-game-engineering-with-unity/business-context-of-game>, last accessed 2022/05/23.

4. *Sahibgareeva G.F., Kugurakova V.V.* Praktiki balansirovanija komp'juternyh igr // Programmnye sistemy: teorija i prilozhenija. 2022. V. 13. №3(54). P. 255–273.

5. *Sahibgareeva G.F., Kugurakova V.V., Bolshakov E.S.* Generacija i balansirovanie igrovyh mehanik videoigr // Nauchnyj servis v seti Internet: trudy XXIV Vserossijskoj nauchnoj konferencii (19–22 of September 2022, on-line). 2022. P. 455–485.

6. Game Development Lifecycle Models. URL: <https://www.studytonight.com/3d-game-engineering-with-unity/game-development-models>, last accessed 2022/05/12.

7. *Aleem S., Capretz L.F., Ahmed F.* Game development software engineering process life cycle: a systematic review // Journal of Software Engineering Research and Development. 2016/ Vol. 4, No. 6. <https://doi.org/10.1186/s40411-016-0032-7>.

8. Manifesto for Agile Software Development. URL: <http://agilemanifesto.org>, last accessed 2022/05/12.

9. Eshhjo raz pro sem osnovnyh metodologij razrabotki. URL: <https://habr.com/ru/company/edison/blog/269789/>, last accessed 2022/05/15.

10. Populjarnye frejmvorki dlja gibkogo upravljenija proektami. URL: <https://www.wrike.com/ru/project-management-guide/agile-frejmvorki-v-upravlении-proektami/>, last accessed 2022/05/20.

11. *Mitre-Hernández H.A., Lara-Alvarez C., González-Salazar M., Martín D.* Decreasing Rework in Video Games Development from a Software Engineering Perspective // Trends and Applications in Software Engineering. Advances in Intelligent Systems and Computing. 2016. Vol. 405. P. 295–304. [https://doi.org/10.1007/978-3-319-26285-7\\_25](https://doi.org/10.1007/978-3-319-26285-7_25).

12. *Sharaeva R.A., Kugurakova V.V., Selezneva N.E.* Metodika uproshhenija tasktrekinga v proektah igrovoj industrii // Programmnye produkty i sistemy. 2022. V. 35. № 3. P. 374–383. <https://doi.org/10.15827/0236-235X.139.374-383>.

## СВЕДЕНИЯ ОБ АВТОРАХ



**ШУБИН Алексей Витальевич** – лаборант кафедры программной инженерии Института ИТИС КФУ. Сфера научных интересов – разработка видеоигр, игровой дизайн.

**Aleksey Vitalevich SHUBIN** – lab assistant at the Department of Software Engineering of the Institute of ITIS KFU. Research interest – videogame development, game design.

e-mail: shubin.aleksey.kpfu@gmail.com

ORCID: 0000-0002-6203-3268



**САХИБГАРЕЕВА Гульнара Фаритовна** – ассистент кафедры программной инженерии Института ИТИС КФУ. Сфера научных интересов – игровая сценаристика, нарративный дизайн, изучение вопроса эффективности создания сценарного прототипа и возможности автоматизации данного процесса.

**Gulnara Faritovna SAHIBGAREEVA** – assistant of the Department of Software Engineering of the Institute ITIS KFU. Research interests – game scripting, narrative design, studying the issue of the effectiveness of creating a scenario prototype and the possibility of automating this process.

email: gulnara.sahibgareeva42@gmail.com

ORCID: 0000-0003-4673-3253



**КУГУРАКОВА Влада Владимировна** – к. т. н., доцент кафедры программной инженерии Института ИТИС КФУ, руководитель НИЛ разработки AR/VR приложений и компьютерных игр. Сфера научных интересов – иммерсивность виртуальных сред, проблемы генерации реалистичной визуализации, различные аспекты проектирования игр, AR/VR, подходы к интерпретации UX.

**Vlada Vladimirovna KUGURAKOVA**, PhD., Docent of the Institute ITIS KFU, Head of Laboratory «AR/VR applications and Gamedev». Research interests include immersiveness of virtual environments, problems of generating realistic visualization, various aspects of game design, AR/VR, approaches to UX interpretation.

email: vlada.kugurakova@gmail.com

ORCID: 0000-0002-1552-4910

## **СПИСОК УПОМЯНУТЫХ ВИДЕОИГР**

1. Tetris [Электронный ресурс] – Режим доступа: <https://ru.wikipedia.org/wiki/Тетрис> (дата обращения: 20.05.2022).
2. Pong (игра) [Электронный ресурс] – Режим доступа: [https://ru.wikipedia.org/wiki/Pong\\_\(игра\)](https://ru.wikipedia.org/wiki/Pong_(игра)) (дата обращения: 20.05.2022).
3. Space Invaders [Электронный ресурс] – Режим доступа: [https://ru.wikipedia.org/wiki/Space\\_Invaders](https://ru.wikipedia.org/wiki/Space_Invaders) (дата обращения: 20.05.2022).
4. Journey on Steam [Электронный ресурс] – Режим доступа: <https://store.steampowered.com/app/638230/Journey/> (дата обращения: 20.05.2022).
5. Vampire Survivors on Steam [Электронный ресурс] – Режим доступа: [https://store.steampowered.com/app/1794680/Vampire\\_Survivors/](https://store.steampowered.com/app/1794680/Vampire_Survivors/) (дата обращения: 20.05.2022).
6. God of War (игра, 2018) [Электронный ресурс] – Режим доступа: [https://ru.wikipedia.org/wiki/God\\_of\\_War\\_\(игра,\\_2018\)](https://ru.wikipedia.org/wiki/God_of_War_(игра,_2018)) (дата обращения: 20.05.2022).
7. Detroit: Become Human on Steam [Электронный ресурс] – Режим доступа: [https://store.steampowered.com/app/1222140/Detroit\\_Become\\_Human/](https://store.steampowered.com/app/1222140/Detroit_Become_Human/) (дата обращения: 20.05.2022).
8. Dying Light 2: Stay Human [Электронный ресурс] – Режим доступа: [https://ru.wikipedia.org/wiki/Dying\\_Light\\_2:\\_Stay\\_Human](https://ru.wikipedia.org/wiki/Dying_Light_2:_Stay_Human) (дата обращения: 20.05.2022).
9. Rust on Steam [Электронный ресурс] – Режим доступа: <https://store.steampowered.com/app/252490/Rust/> (дата обращения: 20.05.2022).
10. Raft on Steam [Электронный ресурс] – Режим доступа: <https://store.steampowered.com/app/648800/Raft/> (дата обращения: 20.05.2022).
11. Project Zomboid on Steam [Электронный ресурс] – Режим доступа: [https://store.steampowered.com/app/108600/Project\\_Zomboid/](https://store.steampowered.com/app/108600/Project_Zomboid/) (дата обращения: 20.05.2022).
12. Valheim on Steam [Электронный ресурс] – Режим доступа: <https://store.steampowered.com/app/892970/Valheim/> (дата обращения: 20.05.2022).

*Материал поступил в редакцию 12 сентября 2022 года*