Минобрнауки России

Федеральное государственное автономное образовательное учреждение высшего образования   
«Национальный исследовательский университет   
«Московский институт электронной техники»

Институт cистемной и программной инженерии и информационных технологий

(СПИНТех)

Артамонова Анастасия Юрьевна

Магистерская диссертация   
по направлению 09.04.04 «Программная инженерия»

Исследование и разработка математической модели и алгоритма движения человека

Студент Артамонова А.Ю.

Руководитель, Доцент, к.т.н. Федоров А.Р.

Москва 2025

ОГЛАВЛЕНИЕ

[ВВЕДЕНИЕ 3](#_Toc199862668)

[ГЛАВА 1. АНАЛИТИЧЕСКИЙ ОБЗОР СУЩЕСТВУЮЩИХ МЕТОДОВ И СРЕДСТВ МОДЕЛИРОВАНИЯ ДВИЖЕНИЙ ЧЕЛОВЕКА 7](#_Toc199862669)

[1.1. Анализ средств моделирования движений человека 7](#_Toc199862670)

[1.2. Анализ методов моделирования движений человека 17](#_Toc199862671)

[1.3. Обоснование метода решения 33](#_Toc199862672)

[Выводы по главе 1 36](#_Toc199862673)

[ГЛАВА 2. ФОРМАЛИЗОВАННОЕ ПРЕДСТАВЛЕНИЕ ПРОЦЕССА МОДЕЛИРОВАНИЯ ДВИЖЕНИЙ ЧЕЛОВЕКА 37](#_Toc199862674)

[2.1. Декомпозиция поставленной задачи моделирования движений человека 37](#_Toc199862675)

[2.2. Формализованное представление процесса поворота суставов с использованием кватернионов 39](#_Toc199862676)

[2.3. Формализация задачи математического моделирования движений 42](#_Toc199862677)

[Выводы по главе 2 43](#_Toc199862678)

[ГЛАВА 3. ПРОГРАММНАЯ РЕАЛИЗАЦИЯ 44](#_Toc199862679)

[3.1. Разработка математической модели движения человека 44](#_Toc199862680)

[3.2. Разработка алгоритма моделирования движений человека 47](#_Toc199862681)

[3.3. Программная реализация 53](#_Toc199862682)

[Выводы по главе 3 58](#_Toc199862683)

[ГЛАВА 4. ЭКСПЕРИМЕНТАЛЬНОЕ ПОДТВЕРЖДЕНИЕ РЕЗУЛЬТАТОВ ИССЛЕДОВАНИЯ 59](#_Toc199862684)

[4.1. Верификация и сравнение модели с помощью метрики качества 59](#_Toc199862685)

[4.2. Ход проведения эксперимента 60](#_Toc199862686)

[4.3. Обоснование достоверности полученных результатов 64](#_Toc199862687)

[Выводы по главе 4 69](#_Toc199862688)

[ЗАКЛЮЧЕНИЕ 70](#_Toc199862689)

[СПИСОК ЛИТЕРАТУРЫ 73](#_Toc199862690)

[ПРИЛОЖЕНИЕ 1 80](#_Toc199862691)

ВВЕДЕНИЕ

**Актуальность исследования.** Моделирование движений человека представляет собой одну из основных задач в таких областях, как современная биомеханика, робототехника и компьютерная графика. С прогрессом технологий виртуальной реальности, анимации, экзоскелетов и автономных систем наблюдается растущая потребность в точных и плавных моделях движения, которые могут воспроизводить естественные траектории человека. Тем не менее, существует актуальная проблема неустойчивости кинематических моделей, возникающей при сингулярностях – особых состояниях, в которых традиционные методы теряют свою устойчивость и точность. Такие ситуации могут возникать, например, во время сложных движений тела, когда углы суставов достигают предельных значений или происходит перераспределение нагрузки между суставами. В условиях быстрого развития технологий, когда модели человеческого движения всё чаще используются для управления роботами, разработки интерактивных симуляторов и медицинских устройств, эта проблема становится особенно актуальной.

Необходимость проведения научного исследования возникает из-за того, что существующие методы моделирования нередко не обеспечивают нужную плавность и точность кинематических траекторий вблизи участков с сингулярностями. Это может привести к устойчивым артефактам в анимации, ошибкам в управлении роботами и снижению эффективности работы медицинских устройств, таких как протезы и экзоскелеты. Ожидаемым результатом данного исследования является создание математической модели и алгоритма движения человека, которые будут устойчивыми к сингулярным ситуациям и обеспечат высокое качество кинематических траекторий. Подобная модель позволит решить проблему нестабильности, повысить надёжность систем, использующих модели человеческого движения, и расширить их применение в различных областях техники.

Новизна данного исследования заключается в применении комплексного подхода к решению проблемы сингулярностей, который объединяет современные математические методы и алгоритмические решения. Предлагаемая методология обеспечивает устойчивость моделей движения человека за счет интеграции различных вычислительных подходов, позволяющих эффективно обрабатывать особые состояния в кинематических цепях. Такой подход не только устраняет существующие ограничения традиционных моделей, но и создает основу для дальнейшего развития технологий моделирования движений человека.

Результаты данного исследования имеют значительную важность для таких областей, как биомеханика, робототехника и компьютерная графика, так как открывают новые перспективы для создания надежных и точных моделей человеческого движения. Разработанные подходы помогают разрешать основные задачи, связанные с нестабильностью в сингулярных режимах, что в свою очередь повышает качество анимации, улучшает управление экзоскелетами и роботизированными системами, а также способствует прогрессу технологий виртуальной и дополненной реальности. Кроме этого, исследование закладывает теоретическую основу для будущих научных изысканий, нацеленных на совершенствование алгоритмов моделирования движений и их применение в медицине, индустрии развлечений и других высокотехнологичных секторах.

**Проблемная ситуация.** Проблема моделирования движений человека в современной науке и технике неразрывно связана с обеспечением устойчивости кинематических моделей при возникновении сингулярностей. Под сингулярностью понимается особое состояние кинематической цепи, при котором происходит потеря ранга якобиана системы, что приводит к неустойчивости решения обратной задачи кинематики. Этот конфликт между необходимостью точного и плавного воспроизведения движений человека и ограниченной устойчивостью существующих математических моделей в определенных условиях создает значительные препятствия для прогресса технологий, основанных на моделировании движений.

Существующая проблема тесно связана с рядом известных научных вопросов, таких как разработка методов численной оптимизации, исследование динамики многозвенных систем и создание алгоритмов управления роботизированными устройствами. В частности, задача преодоления сингулярностей является частью более широкой проблемы адаптивного управления сложными механическими системами. Однако, несмотря на значительный прогресс в этих областях, вопросы обеспечения устойчивости моделей движения человека остаются недостаточно изученными.

**Цель и задачи исследования.** Увеличение точности кинематических траекторий при моделировании движений человека. В соответствии с целью исследования в диссертации необходиморешить следующие задачи:

* аналитический обзор существующих средств и методов моделирования движений человека;
* формализация задачи математического моделирования движений;
* разработка модели движения человека;
* разработка алгоритма моделирования движений человека;
* программная реализация разработанной модели и алгоритма;
* оценка точности модели и алгоритма движения человека.

**Научная новизна исследования.** Разработана математическая модель кинематических траекторий движения человека, устойчивая к сингулярным состояниям. Выведенные аналитические соотношения позволяют адекватно описывать поведение системы вблизи сингулярных зон и минимизировать ошибки, возникающие при потере ранга якобиана.

**Обоснованность и достоверность результатов** подтверждаются строгим математическим обоснованием разработанных моделей и алгоритмов, а также их верификацией на основе сравнения идеальных и расчетных траекторий движения. Для оценки точности предложенных решений проведены численные эксперименты, в ходе которых моделировались различные сценарии движений человека, включая случаи, приближенные к сингулярным состояниям. Расчетные траектории, полученные с использованием разработанных методов, сравнивались с эталонными (идеальными) траекториями, которые воспроизводились на основе данных реальных движений.

Сравнение показало высокую степень соответствия между идеальной и расчетной траекториями, что свидетельствует о корректности предложенных подходов. В частности, отклонения расчетных траекторий от идеальных не превышали допустимых значений, установленных для задач моделирования движений человека. Разработанные алгоритмы продемонстрировали способность сохранять плавность и устойчивость траекторий даже в таких сложных условиях.

**Апробация работы.** Основные положения и результаты диссертационной работы апробированы на следующей конференции:

1. Артамонова А.Ю., Киселев В.В. «Разработка программного модуля движения человека». Научно-практическая конференция с международным участием «Актуальные проблемы информатизации в цифровой экономике и научных исследованиях»: тезисы докладов. М.: МИЭТ, 2023 г., стр. 10.

ГЛАВА 1. АНАЛИТИЧЕСКИЙ ОБЗОР СУЩЕСТВУЮЩИХ МЕТОДОВ И СРЕДСТВ МОДЕЛИРОВАНИЯ ДВИЖЕНИЙ ЧЕЛОВЕКА

* 1. Анализ средств моделирования движений человека

Современные методы моделирования движений человека сталкиваются с серьезной проблемой, связанной с неустойчивостью кинематических моделей при возникновении сингулярностей. Эти особые состояния, характеризующиеся потерей ранга якобиана системы или другими вырожденными условиями, существенно влияют на качество воспроизводимых траекторий, снижая их плавность и точность. Для решения обозначенной проблемы неустойчивости кинематических моделей при сингулярностях целесообразно рассмотреть существующие средства моделирования движений человека.

AnyBody Modeling System (ABMS) – это программное обеспечение для биомеханического моделирования, которое позволяет создавать подробные виртуальные модели человеческого тела и анализировать их движения. Его используют в различных областях, в том числе в медицине, спорте, эргономике и других [1].

Основная идея ABMS – предоставить инструмент для создания анатомически точных моделей скелета, мышц и других тканей человеческого тела. Пользователи могут варьировать параметры модели, такие как длина и мышечная масса, и анализировать, как эти изменения влияют на движения тела.

Система работает на основе принципов обратной динамики, что означает, что она может рассчитывать силы, вызывающие движения, и то, как эти силы влияют на структуры тела. Это особенно полезно при моделировании движений при различных физических нагрузках или при решении реабилитационных задач.

ABMS также интегрируется с данными измерения трафика из систем проверки трафика (таких как Vicon или Motion Analysis), чтобы повысить точность моделей и результатов анализа.

Это программное обеспечение широко используется в исследованиях и разработках, связанных с биомеханикой человеческого тела, и дает возможность более глубокого понимания движения и его влияния на физиологию.

AnyBody Modeling System (ABMS) представляет собой одно из наиболее продвинутых программных решений, предназначенных для моделирования и анализа биомеханических систем с акцентом на взаимодействие человеческого тела с окружающей средой. Одним из ключевых преимуществ ABMS является наличие широкого спектра интерфейсов к другим программным пакетам, что открывает дополнительные возможности для интеграции и расширения функционала. Помимо этого, система позволяет экспертам анализировать работу различных мышц при выполнении разнообразных движений, обеспечивая детализированные данные о мышечных усилиях и нагрузках. Также стоит отметить, что AnyBody способна применяться в таких областях, как эргономическая оптимизация, благодаря использованию разнообразных мускуло-скелетных моделей.

Однако несмотря на всю мощь и гибкость системы, у неё всё же существуют некоторые недостатки. Например, специфика программного обеспечения предполагает наличие высокого уровня технической подготовки пользователей, поскольку требования к знаниям в области программирования и биомеханики достаточно велики, что ограничивает круг потенциальных пользователей. Кроме того, хотя сравнительные исследования показывают, что ABMS демонстрирует хорошие результаты совместимости с другими системами, например с OpenSim, в задачах прогнозирования сил при статических движениях, такие вычисления могут быть трудозатратными и требовать значительного времени на настройку моделей. Ещё одним ограничением является то, что большинство антропометрических моделей, доступных в системе, имеют фотореалистичный вид, который, хотя и удобен для визуального восприятия, но может не всегда точно отражать физические свойства и поведение реальной биологической ткани.

ABMS применяет оптимизационные методы для решения задач инверсной кинематики. Главный принцип заключается в определении угловых перемещений суставов, которые минимизируют расхождение между заданной конечной позицией и реальным положением конечностей модели. Для достижения этой цели система использует метод наименьших квадратов, что позволяет учитывать различные ограничения и уменьшать разницу между желаемыми и фактическими положениями.

ABMS создает модели, основываясь на уравнениях равновесия сил и моментов, а также на кинематических зависимостях между частями тела [2].

Software for Interactive Musculoskeletal Modeling (SIMM) – программное обеспечение, предназначенное для интерактивного биомеханического моделирования мышц и скелета. Разработан в Институте нейронаук в Сан-Диего. Программа позволяет исследователям создавать подробные компьютерные модели человеческого тела с целью анализа и моделирования движений [3].

SIMM основан на концепции создания виртуальных 3D-моделей анатомии, включая кости, суставы и мышцы. Пользователи могут взаимодействовать с моделями, чтобы изменять параметры мышц, исследуя, как эти изменения влияют на движения и динамику тела.

Одним из ключевых элементов SIMM является его способность предоставлять подробные данные о моментах силы и требованиях к силе в различных суставах. Это полезно для анализа биомеханики движения и понимания того, как различные факторы влияют на нагрузку на мышцы и суставы.

SIMM также предоставляет инструменты интеграции с данными о движении, позволяя пользователям комбинировать биомеханические модели с экспериментальными данными для более точного анализа. Программа может быть использована в медицинских исследованиях, реабилитационной медицине и создании протезно-ортопедических изделий.

В целом SIMM предоставляет ученым и инженерам инструмент для изучения биомеханики человеческого тела и его движений, а также для разработки улучшенных методов лечения и реабилитации.

Software for Interactive Musculoskeletal Modeling (SIMM) предоставляет широкие возможности для разработки, анализа и визуализации мускуло-скелетных моделей, что делает его популярным решением в биомеханических исследованиях. Одним из основных преимуществ SIMM является возможность графического представления моделей, позволяющего наглядно создавать и изменять структуры, включая кости, мышцы и суставы. Кроме того, SIMM позволяет рассчитывать движения суставов и моменты, которые каждая мышца может генерировать в различных позах тела, что важно для понимания функциональной активности мышечной системы. Также стоит отметить удобство использования программного обеспечения благодаря интерактивному подходу к моделированию и анимации, что упрощает процесс построения сложных моделей.

Однако у SIMM есть и определённые ограничения. Например, несмотря на популярность и развитый функционал, модели, созданные в SIMM, могут сталкиваться с трудностями при импорте в другие современные платформы, такие как OpenSim 4.0, что снижает уровень совместимости и усложняет интеграцию с другими системами. Ещё одной проблемой является необходимость глубокого знания специфики программирования и биомеханики для эффективного использования SIMM, поскольку без должного уровня подготовки работа с этим ПО может быть затруднена. Также стоит отметить, что описания отдельных функций и возможностей SIMM в научной литературе зачастую сосредоточены на общих принципах его применения, но недостаточно детально раскрыты технические аспекты и алгоритмы, лежащие в основе расчетов, что может затруднять их воспроизведение или адаптацию под конкретные задачи.

SIMM применяет метод псевдообратных матриц для решения задач инверсной кинематики. В данном методе вычисляется матрица Якоби, которая устанавливает связь между изменениями углов суставов и перемещениями конечностей. После этого используется псевдообратная матрица Якоби для определения приближенных значений углов суставов, которые минимизируют расхождение с заданной конечной позицией [4].

Open Simulation for Musculoskeletal Systems (OpenSim) – открытая система моделирования опорно-двигательного аппарата, предоставляющая инструменты для создания детальных вычислительных моделей человеческого тела. Разработан Национальными институтами здравоохранения (NIH) и Стэнфордским университетом [5].

Основная цель OpenSim – предоставить исследователям и инженерам инструменты для анализа биомеханики движений человека и понимания взаимодействия мышц, костей и суставов. Система позволяет создавать трехмерные модели анатомии, включая скелет, мышцы и другие ткани.

Одним из главных преимуществ OpenSim является его открытость и доступность для научного сообщества. Это позволяет исследователям вносить свой вклад, улучшать систему и обмениваться моделями и данными. OpenSim активно используется в медицинских исследованиях, биомеханике и разработке протезно-ортопедических изделий.

Система позволяет проводить виртуальные эксперименты с моделями, изменять параметры мышц и суставов, чтобы анализировать, как эти изменения влияют на движение и нагрузку на организм. OpenSim также интегрируется с данными движения, позволяя пользователям сравнивать моделирование с реальными данными.

Открытость и гибкость OpenSim делают его важным инструментом для тех, кто занимается биомеханическими и реабилитационными исследованиями. Система поощряет коллективное участие и обмен знаниями, способствуя развитию этой области науки и применению полученных знаний в практике здравоохранения.

Open Simulation for Musculoskeletal Systems (OpenSim) представляет собой программное обеспечение с открытым исходным кодом, предназначенное для моделирования, анализа и визуализации мускуло-скелетных систем. Одним из ключевых преимуществ OpenSim является его доступность и поддержка сообществом разработчиков, что обеспечивает регулярные обновления и развитие функционала. Программа позволяет создавать детализированные модели мышц и костей, а также точно рассчитывать динамику движений и мышечные усилия, что делает её полезной в исследованиях, связанных с биомеханикой и реабилитацией. Кроме того, OpenSim поддерживает кросс-платформенность, работая на различных операционных системах, таких как Windows, macOS и Linux, что повышает удобство использования. Также стоит отметить расширяемость платформы, позволяющую интегрировать модели не только человека, но и животных, а также роботизированных систем и их взаимодействие с окружающей средой.

Вместе с тем, у OpenSim есть и определённые ограничения. Например, точность моделирования работы мышц может быть недостаточной в некоторых случаях, особенно при сложных динамических задачах, что связано с упрощением алгоритмов управления мышечными силами. Также известны трудности с корректным масштабированием моделей под конкретные параметры тела, что может приводить к ошибкам или необходимости дополнительной настройки. Ещё одним недостатком является относительно высокий порог входа для новых пользователей – эффективное использование программы требует знаний в области биомеханики, программирования и анализа данных, что может затруднить её применение в образовательной среде или клинической практике без соответствующей подготовки.

OpenSim применяет метод наименьших квадратов для решения проблемы инверсной кинематики. Этот подход направлен на минимизацию расхождений между желаемыми и реальными положениями или ориентациями контрольных точек (например, маркеров) как в модели, так и в экспериментальных данных.

В OpenSim задача представляется в виде оптимизационной: система находит такие угловые значения суставов, которые уменьшают общую квадратичную ошибку между экспериментальными и моделируемыми позициями маркеров, принимая во внимание установленные ограничения на движения суставов [6].

Blender – мощное бесплатное программное обеспечение для 3D-моделирования, анимации, рендеринга, композитинга, создания игр и многого другого. Среди его многочисленных возможностей важное место занимает возможность имитировать движения человека [7].

Одна из лучших особенностей Blender – это то, что он бесплатный и имеет открытый исходный код, что делает его доступным для широкого круга пользователей. Blender предоставляет инструменты для создания персонажей, анимации и 3D-сцен, что делает его полезным инструментом для моделирования движений человека в контексте реабилитации.

В Blender есть возможность создавать гуманоидных персонажей, размещать их в разных позах и анимировать их движения. Поддерживая различные форматы данных, Blender может взаимодействовать с данными о движении, полученными из различных источников, таких как системы захвата движения.

Для анализа движений в Blender можно использовать инструменты, позволяющие изменять параметры скелета персонажа, настраивать анимацию и даже проводить виртуальные эксперименты по анализу влияния различных факторов на движения.

Важным аспектом Blender является сообщество пользователей и большое количество обучающих программ. Это облегчает изучение программы и обмен опытом с другими специалистами в области 3D-моделирования и анимации. Таким образом, Blender предоставляет среду, в которой исследователи и практики могут создавать и анализировать движения человека, применяя их в контексте реабилитации.

Одним из главных преимуществ Blender является его доступность: любой желающий может бесплатно использовать полный функционал программы без ограничений по времени или функционалу. Также значительным достоинством считается активное сообщество разработчиков и пользователей, которое регулярно участвует в развитии проекта, создании обучающих материалов и оказании поддержки новичкам. Программа предлагает широкие возможности для 3D-моделирования и анимации, включая редактор поз персонажей, нелинейную анимацию и продвинутые инструменты визуализации. Благодаря встроенному движку для создания интерактивного контента, Blender также может использоваться при разработке игр и других мультимедийных проектов. Кроме того, Blender предоставляет удобные инструменты для работы с материалами и текстурами, включая возможность создания и использования готовых наборов узлов для различных задач.

Однако у Blender есть и недостатки. Один из самых часто упоминаемых – это сложность освоения программы, особенно для новичков. Интерфейс может показаться непривычным, а большое количество горячих клавиш и специфических операций требует времени на изучение и привыкание. Некоторые пользователи отмечают, что система управления и логика работы Blender отличаются от коммерческих аналогов, что усложняет переход на неё с других программ. Ещё одной особенностью является то, что, хотя Blender идеально подходит для индивидуальных проектов и небольших студий, его использование в крупных профессиональных проектах может быть ограничено из-за отсутствия некоторых специализированных инструментов или менее развитой системы поддержки и документации в сравнении с коммерческими решениями.

Blender применяет метод, основанный на вычислении псевдообратных матриц, в сочетании с итеративными алгоритмами, такими как метод сглаженной градиентной оптимизации, для решения задач инверсной кинематики.

В частности, в Blender используется метод Дампеда псевдообратной матрицы Якоби (Damped Least Squares), который обеспечивает стабильность решения и позволяет справляться с ситуациями, когда стандартная псевдообратная матрица может приводить к нестабильным или неверным результатам (например, вблизи сингулярностей). Этот подход помогает Blender точно управлять позами сложных систем костей, обеспечивая реалистичное движение и минимизируя расхождения между желаемыми и фактическими позициями конечностей модели [8].

Unity представляет собой инструмент для разработки, который позволяет создавать и моделировать движения человека с помощью различных инструментов и пакетов. Для этих задач в основном применяются такие расширения, как Unity Animation Rigging, Unity Mecanim и Unity Humanoid Animation System [9].

Unity Animation Rigging – это инструмент для анимации в Unity, который предоставляет разработчикам возможность в реальном времени создавать и изменять движения персонажей. Он позволяет аниматорам задавать ограничения для костей, включая IK (обратная кинематика) и FK (прямая кинематика), а также управлять физическими взаимодействиями между конечностями и окружающей средой. Это способствует созданию правдоподобных движений персонажей, принимая во внимание их взаимодействие с объектами на сцене и с землей [10].

Одним из ключевых преимуществ этого инструмента является гибкость: он предоставляет возможность создавать сложные механики анимации, такие как динамические движения конечностей или изменение позы персонажа в зависимости от окружения. Также Animation Rigging позволяет комбинировать стандартную анимацию с кастомными эффектами, что делает её особенно полезной для реализации адаптивных движений, например, при взаимодействии персонажа с объектами или при корректировке ступеней лестницы под ноги персонажа. Кроме того, поскольку Animation Rigging основана на Playables API, она хорошо интегрируется с другими системами анимации в Unity и может использоваться совместно с Mecanim.

Однако у этой системы есть и недостатки. Во-первых, уровень сложности выше, чем у классических решений, поскольку работа с ригами требует понимания анимационного графа и дополнительных навыков программирования. Это увеличивает порог входа для новых пользователей и усложняет разработку для менее опытных разработчиков. Во-вторых, хотя Animation Rigging и позволяет добиваться высокой степени контроля над анимацией, она не заменяет полностью состояние и логику анимационного контроллера, поэтому часто используется вспомогательно, а не как самостоятельное решение.

Unity Mecanim – это система, предназначенная для анимации персонажей, особенно эффективная в проектах, где необходимо воспроизводить человеческие движения. Mecanim применяет сложные слои и параметры, что позволяет создавать адаптивные и настраиваемые анимации. Разработчики имеют возможность настраивать переходы между анимациями, связывая их с событиями и переменными для более детального управления поведением персонажа [11].

Одним из наиболее значимых достоинств Mecanim считается наличие интуитивно понятного интерфейса в виде графа состояний, который позволяет удобно организовывать поведение анимации персонажа и упрощает процесс прототипирования и тестирования различных сценариев. Кроме того, Mecanim предлагает удобную систему переадресации анимаций между моделями, что снижает трудозатраты на создание анимационного контента и позволяет повторно использовать однажды подготовленные анимации для разных персонажей. Также важно отметить, что Mecanim поддерживает сложносочинённые иерархии анимационных слоёв, что даёт возможность более точного управления отдельными частями тела персонажа без влияния на остальные анимации.

Вместе с тем Mecanim имеет и ряд ограничений. Например, его возможности по настройке и изменению анимации в реальном времени ограничены по сравнению с более современными решениями, такими как Animation Rigging. Также известны сложности с точным контролем мышечных движений и плавностью перехода между некоторыми анимациями, особенно если требуется гибкая реакция на внешние факторы, такие как столкновения или изменение позиции объектов в сцене. Ещё одним минусом является то, что Mecanim ориентирован преимущественно на человека (Humanoid), и использование его для нестандартных существ или объектов требует дополнительной настройки и иногда даже создания собственных решений.

Система анимации Humanoid от Unity обеспечивает анатомически точное моделирование движений. Она дает возможность загружать и применять готовые скелетные модели или данные о движениях, полученные с помощью систем, таких как Vicon или OptiTrack. Применяя данные захвата движений, можно достичь высокой степени реалистичности, особенно в сложных ситуациях, таких как спортивные тренировки или процессы реабилитации.

Humanoid Animation System использует предварительно определённый скелет (Avatar) и позволяет применять одну и ту же анимацию к разным моделям, сохраняя при этом пропорциональность и естественность движений. Преимуществом такого подхода является то, что он значительно ускоряет производство анимаций и позволяет легко масштабировать проект, так как одна и та же библиотека анимаций может быть задействована для множества персонажей с разной анатомией тела. Также стоит выделить простоту использования: благодаря стандартизованному Avatar, художники и дизайнеры могут работать с анимацией без глубокого знания программирования.

Недостатком Humanoid Animation System является её ограниченная применимость к не человекоподобным персонажам. Поскольку вся система завязана на конкретной структуре скелета, адаптация её для других типов существ становится затруднительной или невозможной без значительных правок. Кроме того, качество переадресации анимации может быть не всегда удовлетворительным, особенно если пропорции тела модели сильно отличаются от средних человеческих параметров, что может вызывать визуальные артефакты и необходимость ручной доработки. Также ограничения возникают при работе с детальной анимацией конкретных зон тела, поскольку стандартные настройки Avatar не всегда позволяют точно контролировать отдельные части модели

В Unity для решения задач инверсной кинематики (IK) применяется несколько подходов, среди которых наиболее распространённым является итеративный метод FABRIK (Forward and Backward Reaching Inverse Kinematics). Этот метод эффективно справляется с задачами IK, принимая во внимание ограничения по длине сегментов и обеспечивая плавное и естественное движение.

FABRIK функционирует итеративно, последовательно перемещаясь вперед и назад по цепочке костей, чтобы достичь необходимой позиции конечности, минимизируя разницу между заданным и текущим положением. Данный метод часто используется для управления конечностями, такими как руки и ноги, благодаря своей стабильности и простоте реализации [12].

* 1. Анализ методов моделирования движений человека

В современном мире моделирование движений человека становится все более востребованным направлением, находящим применение в различных областях, таких как робототехника, компьютерная анимация, медицина и спорт. Повышение требований к точности и реалистичности воспроизведения движений ставит перед исследователями задачу разработки устойчивых и эффективных математических моделей и алгоритмов. Особую сложность представляет решение задачи обратной кинематики, которая позволяет определить углы суставов для достижения заданного положения конечностей или выполнения конкретной траектории движения.

Одним из ключевых вызовов в данной области является проблема сингулярностей – состояний, при которых кинематическая цепь теряет устойчивость или уникальность решения. Это может привести к значительным искажениям траекторий, что делает модели непригодными для задач, требующих высокой точности и плавности. Например, в робототехнике управление антропоморфными устройствами вблизи сингулярных зон может вызывать неконтролируемые движения, а в анимации – заметные артефакты.

Существующие методы решения задач инверсной кинематики предлагают разнообразные подходы, начиная от аналитических методов, основанных на строгих математических расчетах, и заканчивая численными методами и алгоритмами машинного обучения. Каждый из этих подходов имеет свои преимущества и ограничения, которые зависят от специфики применяемой модели и поставленных целей. Рассмотрение существующих методов позволяет выявить наиболее эффективные инструменты и подходы, способствующие разработке персонализированных систем реабилитации.

Существуют несколько численных методов, которые можно использовать для решения задачи инверсной кинематики.

Метод Ньютона-Рафсона – это итерационный численный метод для приближенного нахождения корней уравнения f(x) = 0. Он основан на последовательных линейных приближениях функции и использовании касательных для нахождения лучшего приближения к корню [13].

Алгоритм метода Ньютона-Рафсона:

1. Начальная аппроксимация x0 выбирается достаточно близко к истинному корню.
2. Следующее приближение xn+1 вычисляется по формуле:

где – значение функции в точке ,– значение производной в точке .

1. Процесс повторяется, пока не достигнется заданная точность или пока изменения между последовательными приближениями не станут достаточно малы.

Для решения задачи инверсной кинематики методом Ньютона-Рафсона необходимо выполнить следующие шаги:

1. Определить начальные значения для θ1 и θ2.

2. Определить функции ошибки:

3. Составить матрицу Якоби, которая представляет собой частные производные функций f1 и f2 по переменным и :

Вычисление производных:

4. Итеративно обновить значения и , используя формулу Ньютона-Рафсона:

5. Продолжить процесс до тех пор, пока ошибка между текущей и целевой позицией не станет меньше заданного значения.

Основное преимущество метода Ньютона-Рафсона – это квадратичная сходимость при достаточно хорошем начальном приближении. Это означает, что при каждом шаге ошибка уменьшается пропорционально квадрату предыдущей ошибки, что делает метод очень быстрым по сравнению с другими численными методами, такими как метод бисекции или секущих [14].

Метод легко применим к уравнениям, где производную функции можно вычислить аналитически или численно, что позволяет быстро приближаться к решению.

Метод Ньютона-Рафсона используется не только для нахождения корней уравнений, но и для задач оптимизации, где необходимо находить экстремумы функций, решать системы нелинейных уравнений и выполнять другие сложные вычислительные задачи.

Метод можно расширить для решения систем нелинейных уравнений с помощью матрицы Якоби, что делает его мощным инструментом для многомерных задач.

К недостаткам метода Ньютона-Рафсона относится качество и скорость сходимости сильно зависят от выбора начальной точки 𝑥0. Если начальное приближение выбрано неудачно, метод может сходиться медленно, застрять в локальном минимуме или не сойтись вовсе [15].

Метод не гарантирует сходимость для всех функций. Вблизи точек перегиба, разрывов производной или для функций с несколькими корнями метод может давать ошибочные или расходящиеся решения. Например, если производная 𝑓′(𝑥) = 0, то следующая итерация метода не определена.

Метод требует дифференцируемости функции, и вычислимости её производной во всех точках. Если производная не существует или трудно вычисляется, метод может оказаться неприменимым.

Для каждой итерации необходимо вычислять производную функции, что может быть вычислительно дорого или затруднительно, особенно для сложных функций или систем уравнений.

Метод может сходиться к разным корням в зависимости от начального приближения. Это создает трудности при решении уравнений с несколькими корнями, так как нельзя точно знать, к какому корню метод будет стремиться.

Если производная функции в точке приближения очень мала, это может привести к большим шагам в направлении поиска, что может нарушить сходимость или сделать метод нестабильным.

Метод градиентного спуска – это один из основных численных методов оптимизации, используемый для минимизации (или максимизации) функции, обычно называемой функцией потерь. Он основывается на идее, что значение функции уменьшается быстрее всего, если двигаться в направлении, противоположном градиенту функции в данной точке [16].

Алгоритм метода градиентного спуска:

1. Выбирается начальная точка x0 (начальная аппроксимация).
2. На каждом шаге вычисляется градиент функции ∇f(xn) в текущей точке xn, который указывает направление наибольшего роста функции. Градиент – это вектор, который указывает направление наибольшего увеличения функции. Для многомерной функции f(x1, x2, …, xn) градиентом будет вектор частных производных:
3. Следующее приближение вычисляется по формуле:

где α – шаг обучения, который определяет, насколько сильно изменится точка на каждом шаге, – градиент функции в точке xn, который показывает направление наибольшего роста функции.

1. Процесс повторяется, пока не достигнется минимальное значение функции или изменение функции станет достаточно малым.

Для решения задачи инверсной кинематики методом градиентного спуска необходимо выполнить следующие шаги:

1. Определить начальные значения для θ1 и θ2.

2. Определить функцию ошибки:

3. Вычислить частные производные (градиенты) функции ошибки по углам и .

4. Итеративно обновить значения и в направлении, противоположном градиенту ошибки с использованием шага обучения :

5. Продолжить процесс до тех пор, пока ошибка не станет меньше заданного значения или не будет превышено максимальное количество итераций.

Главное достоинство метода градиентного спуска состоит в том, что он обеспечивает эффективный поиск минимумов функции, особенно в тех случаях, когда другие подходы требуют значительных вычислительных мощностей. Метод градиентного спуска отлично справляется с большими объемами данных и многомерными функциями, где аналитическое решение задачи может быть либо слишком сложным, либо недоступным [17].

Тем не менее, у данного метода имеются свои недостатки. Во-первых, он зависит от выбора величины шага обучения. Если шаг слишком мал, оптимизация будет происходить очень медленно, а если он слишком велик, метод может не достичь оптимального решения и даже привести к дивергенции. Кроме того, градиентный спуск может "застревать" в локальных минимумах, особенно в задачах с сильно нелинейными функциями. В таких ситуациях метод может не обнаружить глобальный минимум и остановиться на одном из локальных решений [18].

Еще одной сложностью является необходимость вычисления градиента, что может быть затруднительно для сложных моделей или больших объемов данных. В заключение, градиентный спуск может демонстрировать медленное сходимость на плоских участках функции потерь, где градиенты практически равны нулю. Это может потребовать применения дополнительных методов, таких как адаптивные алгоритмы изменения шага.

Матрица Якоби – это матрица частных производных векторной функции f(x), которая описывает локальные линейные зависимости между функцией и её переменными [19]. Для функции , векторная функция имеет матрицу Якоби Jf(x) следующего вида:

Задачи, связанные с матрицей Якоби, включают решение систем нелинейных уравнений вида f(x) = 0, где f(x) – векторная функция.

Матрица Якоби описывает локальное поведение нелинейной функции, что позволяет использовать её для приближенного решения систем уравнений. Матрица Якоби является основным инструментом в методах оптимизации, таких как метод Ньютона и другие градиентные методы. Она позволяет проводить анализ устойчивости динамических систем. Матрица Якоби легко обобщается на многомерные функции, что делает её полезной для работы с многими переменными [20].

Вычисление матрицы Якоби для сложных функций может быть трудоемким и требовать значительных вычислительных ресурсов. Для сильно нелинейных функций матрица может изменяться значительно между итерациями, что усложняет процесс оптимизации. В задачах с шумными данными матрица Якоби может быть нестабильной и не давать адекватных результатов. Для корректного вычисления требуется, чтобы функция являлось непрерывной и дифференцируемой [21].

Когда матрица Якоби не является квадратной или не обратима, вместо обычной обратной матрицы используется **псевдообратная матрица Мура-Пенроуза**. Псевдообратная матрица позволяет решить систему уравнений в смысле минимизации ошибок или нахождения наилучшего приближения решения [22].

Для вычисления псевдообратной матрицы Якоби используют различные методы, но один из самых распространенных – метод сингулярного разложения матрицы Jf(x).

Если матрица Jf(x) представлена в виде Jf(x) = U∑VT, где U и V – ортогональные матрицы, ∑ – диагональная матрица сингулярных чисел, то псевдообратная матрица определяется как:

где – псевдообратная матрица диагольной матрицы ∑, которая вычисляется путем обращения всех ненулевых элементов на диагонали, а все нулевые элементы остаются на месте.

Одним из ключевых достоинств псевдообратной матрицы Якоби является её способность справляться с переопределёнными системами, в которых количество уравнений превышает количество неизвестных, а также с недоопределёнными системами. В таких случаях традиционная обратная матрица не может быть использована, и псевдообратная матрица позволяет находить оптимальные решения в контексте метода наименьших квадратов. Кроме того, ещё одним её преимуществом является возможность стабилизации численных решений, особенно в ситуациях, когда матрица Якоби имеет плохую обусловленность (то есть её определитель близок к нулю) [23].

Тем не менее, у данного метода имеются и свои недостатки. Прежде всего, вычисление псевдообратной матрицы может потребовать значительных ресурсов при работе с большими матрицами, так как включает в себя процесс сингулярного разложения (SVD). Этот этап довольно медленный по сравнению с обычным обращением матриц, особенно в случае крупных систем. Кроме того, псевдообратная матрица может оказаться чувствительной к шумам в данных, что может привести к нестабильным результатам. Это связано с тем, что небольшие сингулярные значения в процессе SVD могут значительно увеличиваться при обращении, что, в свою очередь, приводит к усилению ошибок [24].

Метод экспоненциального сглаживания – это метод анализа временных рядов, используемая для уменьшения шума и колебаний в данных, что позволяет получить более плавные и предсказуемые результаты. Этот методосновывается на применении экспоненциальных весов к наблюдениям временного ряда. Более свежие данные получают больший вес, а старые данные – меньший. Таким образом, сглаженные значения реагируют на изменения в данных, но менее подвержены случайным колебаниям и шуму [25].

Сглаженное значение St для момента времени t вычисляется как:

где Yt – наблюдаемое значение временного ряда в момент времени t, St-1 – сглаженное значение в предыдущий момент времени, α – коэффициент сглаживания (0 < α < 1), который определяет, насколько сильно новое значение влияет на сглаженное значение.

Для решения задачи инверсной кинематики методом сглаживания необходимо выполнить следующие шаги:

1. Определить текущее положение конечного сустава (xcurr, ycurr) и позицию, к которой стремится конечный сустав – (xtarget, ytarget).
2. Составить матрицу Якоби:
3. Определить ошибку между текущим положением конца манипулятора и целевой точкой:
4. Воспользоваться псевдообратной матрицей Якоби для улучшения устойчивости вблизи сингулярных конфигураций:

где – транспонированная матрица Якоби, I – единичная матрица, λ – коэффициент сглаживания, обычно выбирается малым (0,01 или 0,001).

1. Изменение в углах сутавов:

где – изменения углов суставов.

Углы суставов обновляются с учетом найденных именений:

1. Процесс повторяется до тех пор, пока величина ошибки не станет достаточно малой, или пока не будет достигнуто максимальное количество итераций.

Одним из ключевых преимуществ является его простота и легкость в применении. Он не требует сложных расчетов или больших объемов информации, что делает его идеальным для быстрого использования в условиях ограниченных ресурсов. Кроме того, данный метод акцентирует внимание на более свежих данных, что особенно полезно для прогнозирования, так как актуальная информация чаще всего лучше отражает текущие тенденции. Еще одним плюсом является гибкость метода: различные варианты экспоненциального сглаживания, такие как метод Хольта или Хольта-Винтерса, позволяют учитывать как тренды, так и сезонные изменения [26].

Тем не менее, у данного метода имеются и свои недостатки. Один из них заключается в его ограниченной способности обрабатывать временные ряды со сложными, нерегулярными паттернами или резкими изменениями. Экспоненциальное сглаживание эффективно работает с плавными трендами, но может оказаться неэффективным в случаях, когда данные содержат значительный шум или внезапные скачки. Кроме того, существует проблема правильного выбора параметра сглаживания (альфа). Неправильный выбор этого параметра может привести как к избыточному сглаживанию и утрате важных деталей, так и к высокой чувствительности прогноза к случайным колебаниям в данных [27].

Генетические алгоритмы – это класс эвристических методов оптимизации, которые имитируют процесс естественного отбора, свойственный биологической эволюции. Генетические алгоритмы основаны на таких концепциях, как наследственность, мутация, селекция и скрещивание (кроссовер), и применяются для поиска глобальных оптимумов в сложных многомерных задачах [28].

Генетический алгоритм представляет возможные решения задачи как "особи" или "хромосомы" и эволюционирует их популяцию через несколько поколений, чтобы найти наилучшее решение. Алгоритм использует принципы естественного отбора для сохранения лучших решений и модифицирует их для получения еще лучших результатов в следующих поколениях.

Этапы работы генетического алгоритма:

1. Генерация начальной популяции возможных решений. Каждое решение кодируется в виде хромосомы – последовательности чисел или битов, представляющих переменные задачи.
2. Для каждой особи (решения) вычисляется значение функции приспособленности (fitness function), которая оценивает, насколько это решение приближает к оптимуму задачи. Цель состоит в том, чтобы максимизировать или минимизировать значение этой функции.
3. Выбор особей, которые будут участвовать в создании следующего поколения. Наиболее приспособленные особи имеют больше шансов быть выбраны для размножения. Один из популярных методов селекции – турнирный отбор, при котором несколько случайных особей сравниваются, и лучшая из них выбирается для скрещивания.
4. Скрещивание, процесс, в ходе которого две родительские хромосомы комбинируются для создания новых потомков. Кроссовер может быть одноточечным (разделение родительских хромосом в одной точке) или многоточечным (разделение в нескольких точках), что позволяет передавать часть информации от каждого родителя потомку.
5. Для поддержания разнообразия в популяции некоторые потомки подвергаются случайным изменениям. Мутации помогают избежать преждевременного застревания алгоритма в локальных оптимумах.
6. После скрещивания и мутаций потомки добавляются в популяцию, и процесс продолжается. На каждом этапе слабые особи могут быть исключены из популяции.
7. Алгоритм продолжает эволюцию, пока не достигнет критерия остановки, например: максимальное количество поколений, достижение заданной точности решения, стабильность популяции (отсутствие значительного улучшения в течение нескольких поколений).

Для решения задачи инверсной кинематики методом сглаживания необходимо выполнить следующие шаги:

1. У каждой особи генотип (хромосома), состоит из двух параметров – углов .
2. Оценка приспособленности:
3. На этапе отбора выбираются наиболее приспособленные особи для размножения.
4. Создаются новые особи(потомки) путем комбинирования генов (углов суставов) двух родителей.
5. Мутация позволяет вносить случайные изменения в особи для поддержания разнообразия популяции.
6. После отбора, скрещивания и мутации создается новая популяция, которая заменяет предыдущую.
7. Процесс повторяется, пока не найдена особь с минимальной ошибкой или не достигнуто максимальное количество итераций.

Одним из ключевых достоинств данного подхода является его умение находить качественные приближённые решения для задач с множеством локальных минимумов, где традиционные методы, такие как градиентный спуск, могут "застревать". Генетический алгоритм не требует, чтобы оптимизируемая функция являлась непрерывной или дифференцируемой, что делает его универсальным инструментом для различных задач. Используя операторы мутации, кроссинговера и селекции, этот метод охватывает обширные области пространства решений и эффективно их исследует, избегая преждевременной сходимости [29].

Еще одним достоинством является параллельный характер генетического метода: он одновременно анализирует множество вариантов (популяцию), что повышает шансы на нахождение глобального оптимума. Это качество особенно актуально при работе с обширными параметрическими пространствами или сложными нелинейными функциями. Кроме того, метод легко настраивается для решения различных типов задач, как дискретных, так и непрерывных, и может быть объединен с другими алгоритмами для повышения своей эффективности.

Тем не менее, у генетического метода имеются и серьезные недостатки. Во-первых, он может быть достаточно медленным, поскольку требует множество итераций с различными возможными решениями. Это приводит к значительным затратам вычислительных ресурсов, особенно при решении сложных задач с большим количеством параметров. Во-вторых, генетический метод не всегда обеспечивает нахождение точного глобального оптимума, особенно если параметры алгоритма (размер популяции, вероятность мутаций и кроссинговера) настроены неправильно. В таких ситуациях метод может привести к субоптимальным решениям [30].

Еще одним минусом является высокая чувствительность к настройке параметров. К примеру, слишком высокая вероятность мутации может привести к беспорядочному поиску без какой-либо структуры, в то время как слишком низкая вероятность может вызвать недостаток разнообразия в решениях. Кроме того, данный метод может оказаться неэффективным для задач, где требуется точное решение, так как он предоставляет скорее приближенные результаты.

**Метод роя частиц** (PSO, Particle Swarm Optimization) – это алгоритм оптимизации, основанный на имитации коллективного поведения (роевого интеллекта), наблюдаемого в природе, например, у стай птиц или косяков рыб. Алгоритм предложен в 1995 году Джеймсом Кеннеди и Расселом Эберхартом и используется для решения задач глобальной оптимизации в многомерных пространствах [31].

Метод FABRIK (Forward And Backward Reaching Inverse Kinematics) – это один из популярных методов решения задачи инверсной кинематики, который основан на пошаговом приближении к целевой позиции путём прямого и обратного перемещения суставов. В отличие от других методов, FABRIK не использует матрицы Якоби или сложные вычисления, а использует чисто геометрические соображения [32].

Метод начинается с инициализации начальной позиции конечности (например, конечности руки) и определяет желаемую целевую позицию. Каждая часть (звено) конечности имеет определённую длину и начальное положение.

Метод FABRIK сначала перемещает исследуемые суставы (от конечной точки к более близким суставам) с целью добиться того, чтобы конечная точка переместилась как можно ближе к заданной цели. Это достигается путем выравнивания суставов на основе их расстояний.

где – позиция i-го звена, – длина сегмента.

После обратной работы происходит прямая работа, где суставы корректируются в противоположном направлении, чтобы сохранить длину звеньев и гарантировать, что конечность остается ограниченной в своих движениях, соблюдая заданные ограничения.

Процесс повторяется для достижения заданной целевой позиции с достаточно высокой точностью. Итерации продолжаются до тех пор, пока конечная точка не будет достаточно близка к цели или не будет достигнуто максимальное количество итераций.

Одним из основных преимуществ метода FABRIK является его высокая эффективность. Алгоритм, благодаря своей итеративной структуре, может быстро достигать нужной конфигурации. Кроме того, он обладает простой реализацией, что делает его доступным для внедрения в различные проекты. FABRIK также обеспечивает качественные результаты, создавая плавные и естественные движения, что является критически важным для анимации персонажей и симуляций. Гибкость алгоритма позволяет адаптировать его под различные задачи и условия, что делает его универсальным инструментом для разработчиков в области анимации и кинематики [33].

FABRIK обладает рядом недостатков, которые могут повлиять на его эффективность и применимость. Во-первых, его производительность сильно зависит от начальной конфигурации суставов. Если начальная позиция конечной точки далеко от желаемого положения, это может привести к большому числу итераций, необходимых для достижения цели, что негативно сказывается на скорости работы алгоритма [34].

Кроме того, FABRIK может сталкиваться с проблемой локальных минимумов. Это значит, что алгоритм может застрять в конфигурации, где дальнейшие изменения суставов не приводят к улучшению, что требует дополнительных корректировок и усложняет выполнение задачи.

Другим важным аспектом является ограничение движений суставов. Алгоритм может не находить решения, если суставы имеют жесткие ограничения, что может потребовать ручных изменений или альтернативных методов для достижения желаемого результата.

Наконец, в сложных системах, например, при моделировании тела с высокой детализацией, метод может давать нестабильные результаты из-за коллизий между звеньями. Эта несоответствие часто требует дополнительных оптимизаций, чтобы обеспечить естественное и правильное движение.

Метод наименьших квадратов – это статистический метод, применяемый для нахождения функции (обычно линейной или нелинейной), которая лучше всего приближает набор данных [35]. Его цель – минимизировать сумму квадратов отклонений между наблюдаемыми значениями yi и значениями, предсказанными моделью .

Функция, минимизируемая методом наименьших квадратов:

Для решения задачи инверсной кинематики методом наименьших квадратов необходимо выполнить следующие шаги:

1. Определить целевую точку в пространстве и вычисляемую точку , определяемую через функции кинематики:

где – вектор параметров суставов (например, углы поворота).

1. Определить функцию ошибки:
2. Линейное приближение через Якобиан:

Для итерационного решения используется разложение в ряд Тейлора:

где – якобиан (матрица частных производных).  
Вектор изменения параметров:

где , а - псевдообратная матрица (рассчитывается с использованием метода наименьших квадратов).

1. Продолжить процесс до тех пор, пока ошибка между текущей и целевой позицией не станет меньше заданного значения.

Метод наименьших квадратов имеет множество преимуществ, которые делают его одним из наиболее распространенных и часто применяемых способов в статистике, анализе данных и оптимизации. Его главное достоинство заключается в том, что он предлагает легкое и интуитивно понятное решение для минимизации расхождений между фактическими и предсказанными значениями, что делает его идеальным для задач линейной регрессии. Кроме того, данный метод отличается математической строгостью и позволяет находить аналитическое решение для линейных моделей, что упрощает его использование и обеспечивает высокую эффективность вычислений. Метод устойчива к случайным шумам в данных, при условии, что ошибки распределены нормально, из-за чего он часто принят в качестве стандартного метода для анализа данных [36].

Тем не менее, у метода имеются свои недостатки. Одним из основных является чувствительность к выбросам, так как сумма квадратов ошибок присваивает непропорционально высокий вес значительным отклонениям. Это может вызывать серьёзные искажения в результатах, если в данных присутствуют выбросы. При использовании нелинейных моделей сложность метода возрастает: аналитическое решение невозможно, и требуется прибегать к численным методам, которые могут быть менее точными и более ресурсозатратными. Кроме того, метод основывается на предположении о нормальном распределении ошибок и одинаковой дисперсии, что не всегда соответствует реальным данным. При нарушении этих предположений точность и надёжность получаемых результатов могут существенно ухудшиться [37].

Кватернионы – это математическая структура, которая представляет собой расширение комплексных чисел в четырёхмерное пространство [38]. Они используются для описания вращений в трёхмерном пространстве, обеспечивая компактное и эффективное представление ориентации без некоторых проблем, связанных с другими методами, такими как углы Эйлера. Кватернионы имеют вид , где w, x, y, z – вещественные числа, а i, j, k – мнимые единицы, подчиняющиеся специфическим правилам умножения. Чаще всего кватернион записывают в виде , где w – скалярная часть, а v⃗ – векторная часть (x, y, z).

В задачах инверсной кинематики кватернионы применяются для представления и вычисления вращений суставов, а также для определения целевых ориентаций конечности. Кватернионы обеспечивают точное и стабильное управление ориентацией конечностей без артефактов, что делает их предпочтительным выбором для работы с роботизированными манипуляторами, реабилитационными устройствами и анимацией в компьютерной графике. Их применение упрощает вычисления и позволяет создавать плавные движения, что особенно важно для моделирования движений человека.

Кватернионы обладают рядом значительных преимуществ, которые делают их идеальным инструментом для работы с вращениями в трёхмерном пространстве. Одним из главных достоинств является их способность избегать проблемы гимбальной блокировки, с которой сталкиваются углы Эйлера. Это означает, что кватернионы могут корректно описывать вращения даже в сложных конфигурациях, не теряя степени свободы. Кроме того, они представляют вращения в компактной форме, используя всего четыре параметра, что делает их более вычислительно эффективными по сравнению с матрицами вращения, которые требуют девяти параметров. Кватернионы также обеспечивают высокую стабильность при множественных последовательных вращениях, поскольку они менее подвержены накоплению ошибок. Ещё одним важным преимуществом является лёгкость интерполяции. Сферическая линейная интерполяция (SLERP) позволяет плавно изменять ориентацию без артефактов, что особенно важно для анимации и управления движениями [39].

Однако у кватернионов есть и недостатки [40]. Одной из главных трудностей является их не интуитивность. Поскольку они работают в четырёхмерном пространстве, их сложнее интерпретировать и визуализировать по сравнению с углами Эйлера. Кватернионы требуют постоянной нормализации, так как только единичные кватернионы корректно описывают вращения, что накладывает дополнительные вычислительные затраты. Кроме того, существует сложность в преобразовании между кватернионами и более привычными углами, такими как углы Эйлера, что может вызвать дополнительные трудности в определённых задачах. Эти ограничения, хотя и менее значительны, требуют учитывать особенности применения кватернионов в зависимости от конкретных потребностей и условий задачи.

* 1. Обоснование метода решения

Проанализировав различные методы и средства, используемые для моделирования движений человека, определены их недостатки. На основе проведённого анализа составлена таблица 1.1.

Таблица 1.1. – Методы и средства моделирования движений человека

|  |  |  |
| --- | --- | --- |
| Средства | Методы | Формальное описание |
| ABMS | Метод наименьших квадратов |  |
| OpenSim |
| SIMM | Псевдообратная матрица Якоби |  |
| Blender |
| Unity | Метод FABRIK |  |

Среди основных недостатков рассматриваемых методов можно выделить целый ряд существенных ограничений, которые могут отрицательно сказываться на их эффективности и применимости в моделировании движений человека. Во-первых, данный подход не учитывает физиологические или механические ограничения суставов, что может приводить к генерации некорректных или невозможных с точки зрения анатомии движений. Во-вторых, некоторые алгоритмы обладают высокой чувствительностью к входным данным, то есть даже незначительные погрешности или шумы в данных способны оказывать существенное влияние на результаты работы, что снижает надёжность и точность получаемых решений.

Еще одной серьёзной проблемой является неустойчивость некоторых методов при наличии сингулярностей в математической модели системы, что может вызывать значительные скачки в управлении или полностью приводить к потере работоспособности алгоритма. Кроме того, методы полностью игнорируют динамические ограничения системы – такие как ограничения по скорости, ускорению, моментам и другим инерционным характеристикам, что делает их применение в реальных условиях достаточно спорным.

Также стоит отметить, что один из подходов не гарантирует нахождения оптимального решения, особенно в случае систем с избыточными степенями свободы, где пространство возможных решений становится слишком широким и плохо структурированным. В таких ситуациях алгоритм может как попадать в локальные экстремумы, так и вообще не обеспечивать минимальной функциональной эффективности.

Наконец, одним из наиболее заметных минусов является сравнительно медленная сходимость метода, которая может значительно увеличивать время вычислений и, как следствие, препятствовать его использованию в моделировании движений человека.

Для решения задач инверсной кинематики в системах с высоким числом степеней свободы, таких как моделирование движений человека, необходимо использовать эффективные методы, которые могут адекватно моделировать сложные движения.

Поскольку все вышеперечисленные методы, рассмотренные по отдельности, не обеспечивают необходимого уровня эффективности, точности или устойчивости в решении поставленной задачи, возникает обоснованная необходимость в их интеграции. Каждый из подходов обладает как своими преимуществами, так и недостатками, которые ограничивают его применение в сложных или динамически изменяющихся условиях. В связи с этим представляет интерес разработка алгоритма, которая позволит объединить сильные стороны каждого из существующих подходов, нивелировав при этом их индивидуальные слабые места.

Одним из подходов, который может стать важной составляющей такой методики, является применение псевдообратной матрицы Якоби. Этот метод обладает рядом важных свойств, делающих его особенно полезным при решении задач кинематического управления системами со сложной структурой. В частности, псевдообратная матрица позволяет находить решения даже в тех случаях, когда система является избыточной или недоопределённой, обеспечивая тем самым устойчивость и работоспособность алгоритма в широком диапазоне условий. Это означает, что количество степеней свободы может быть как больше, так и меньше, чем требуется для достижения заданного положения. Данный метод обеспечивает оптимальное приближение и может быть использован, когда стандартная матрица Якоби не имеет обратной. Таким образом, псевдообратная матрица Якоби позволяет эффективно распределять движения между степенями свободы, что способствует достижению целевого положения с минимальными отклонениями.

Тем не менее, применение псевдообратной матрицы Якоби имеет свои недостатки. Одним из них является шум, который может возникать во время итерационных процессов. Этот шум возникает из-за того, что небольшие изменения в исходных данных могут вызывать значительные колебания в движениях, особенно если система подвержена числовым ошибкам или если сингулярные значения в разложении матрицы Якоби находятся близко к нулю. Это приводит к тому, что движения становятся резкими и нестабильными, что критично для задач, где важны плавность анимации и точность управления.

Для решения этой задачи часто применяется метод сглаживания, который помогает снизить влияние шумов и стабилизировать движения. Экспоненциальное сглаживание считается одним из самых эффективных методов, так как оно фокусируется на последних изменениях и уменьшает значимость более старых данных. Это позволяет системе оставаться чувствительной к изменениям в исходных параметрах, одновременно смягчая резкие колебания, вызванные ошибками в расчетах. В результате метод сглаживания помогает уменьшить резкие скачки в управлении, делая движения более предсказуемыми и плавными.

Кроме того, для повышения эффективности решения задач инверсной кинематики может использоваться представление вращений в виде кватернионов. Кватернионы являются удобным инструментом для описания трёхмерных вращений, поскольку они избегают проблем, связанных с особенностями других форм представления, таких как матрицы поворотов или углы Эйлера. Одним из ключевых преимуществ кватернионов является их способность избегать явления “захвата” (“gimbal lock”), которое часто возникает при использовании углов Эйлера. Кроме того, кватернионы обеспечивают более компактное и вычислительно эффективное представление вращений, что особенно важно для систем с высоким числом степеней свободы.

В контексте инверсной кинематики кватернионы могут быть использованы для точного управления ориентацией конечных звеньев системы. Это достигается путём оптимизации целевой функции, учитывающей как положение, так и ориентацию объектов. Применение кватернионов также упрощает интерполяцию вращений, что делает их незаменимым инструментом в анимации и робототехнике, где требуется плавное и реалистичное движение. В сочетании с псевдообратной матрицей Якоби кватернионы позволяют значительно улучшить точность и стабильность управления движением, минимизируя ошибки и обеспечивая плавность траекторий.

В конечном итоге, использование псевдообратной матрицы Якоби, методов сглаживания и представления вращений с помощью кватернионов представляет собой набор инструментов для решения задач инверсной кинематики. Такой подход обеспечивает точность, адаптивность и стабильность в сложных системах с избыточными или недоопределёнными степенями свободы, делая движения более плавными, предсказуемыми и реалистичными.

Выводы по главе 1

Проведен аналитический обзор существующих средств и методов. Выявлены существенные недостатки описанных методов, поэтому предложено для решения задачи объединить использование псевдообратной матрицы Якоби, метода сглаживания и кватернионов, чтобы нивелировать слабые стороны методов.

ГЛАВА 2. ФОРМАЛИЗОВАННОЕ ПРЕДСТАВЛЕНИЕ ПРОЦЕССА МОДЕЛИРОВАНИЯ ДВИЖЕНИЙ ЧЕЛОВЕКА

2.1. Декомпозиция поставленной задачи моделирования движений человека

Моделирование движений человека представляет собой многогранный и многоуровневый процесс, в котором требуется учитывать широкий спектр биомеханических, математических и вычислительных аспектов. Само по себе движение человека – это не просто последовательность перемещений в пространстве, но результат сложной и согласованной работы различных частей тела, взаимодействующих между собой в рамках единой системы. Чтобы приблизиться к точному воспроизведению этой системы в условиях виртуального моделирования, необходимо произвести декомпозицию исходной задачи, разложив её на отдельные составляющие, каждая из которых отвечает за свой уникальный аспект общего процесса.

Первым и фундаментальным шагом на этом пути является решение задачи инверсной кинематики. Это направление математического анализа нацелено на то, чтобы, имея заданное положение конечной точки – например, руки, ноги или пальца – определить, какие именно углы поворота в суставах необходимы для того, чтобы достичь данной точки. Суть инверсной кинематики заключается в поиске таких конфигураций суставов, при которых нужная точка окажется в целевом положении. Это не всегда однозначная задача, так как для одной и той же позиции конечной точки может существовать несколько возможных вариантов поворотов суставов, особенно при высокой подвижности конечностей. Тем не менее, именно эта процедура лежит в основе создания динамически корректных анимаций, в которых движения персонажа выглядят логично, правдоподобно и органично.

После того как углы поворота определены, возникает необходимость преобразовать эти параметры в такие формы, которые будут удобны и устойчивы для применения в процессе анимации. Здесь на сцену выходят кватернионы – особая форма представления вращений, обладающая рядом ключевых преимуществ. Использование кватернионов позволяет устранить проблему гимбальной блокировки, избежать резких скачков и дискретностей в анимации, а также обеспечивает плавные и стабильные повороты во всех трех измерениях. Благодаря кватернионному представлению можно последовательно и без потери точности применять сложные вращения, что особенно важно при воспроизведении естественных, непрерывных движений. В этом контексте кватернионы выступают как связующее звено между абстрактными углами, полученными из инверсной кинематики, и фактическими преобразованиями, производимыми над виртуальной моделью тела.

Однако движение человека – это не просто совокупность отдельных поворотов. Человеческое тело – это интегрированная система, в которой каждый сустав, каждая кость и каждая мышца участвуют в общей динамике. Поэтому важным этапом декомпозиции становится организация суставов в логически и функционально обоснованные группы. Наиболее эффективным способом структурирования является объединение суставов в триады, то есть группы по три, что позволяет легче проследить за взаимосвязями между ними и более последовательно управлять их положением в процессе движения. Такая группировка обеспечивает удобную основу для рекурсивного обхода тела: начиная с корневого звена, например, таза, можно постепенно проходить по всем остальным частям, контролируя их положение и движение относительно предыдущих. Это особенно актуально при построении иерархических скелетных моделей, где каждый элемент влияет на поведение следующих за ним звеньев.

Ещё один критически важный аспект, который следует включить в декомпозицию, заключается в необходимости наложения ограничений на углы поворота суставов. В реальном мире движения человека ограничены анатомией: связками, мышцами, формой суставов. И хотя виртуальные модели могут технически выполнять любые возможные трансформации, отсутствие ограничений приведёт к тому, что движения станут неестественными и даже абсурдными с точки зрения наблюдателя. Поэтому для обеспечения реалистичности необходимо заранее задать допустимые диапазоны значений для каждого из суставов. Это позволит фильтровать некорректные результаты, полученные в ходе решения инверсной кинематики, и тем самым сохранить правдоподобие движений.

В совокупности все эти этапы образуют цельную, непрерывную цепочку преобразований, направленных на приближение поведения виртуального тела к естественному человеческому движению. Декомпозиция задачи позволяет систематизировать процесс моделирования, сделать его более управляемым и гибким, а также открывает путь к дальнейшей автоматизации и улучшению качества анимации. Именно благодаря такой структурной проработке становится возможным достичь тех результатов, которые визуально и динамически соответствуют реальному поведению человеческого тела во времени и пространстве.

2.2. Формализованное представление процесса поворота суставов с использованием кватернионов

Кватернионы – это система гиперкомплексных чисел, образующая векторное пространство, которая может быть использована как способ описания вращения объекта в пространстве вокруг произвольной оси в пространстве [41].

Если при переходе от вещественных к комплексным добавляется одна мнимая компонента i: z = a + bi, то при переходе от комплексных к кватернионам необходимо добавить ещё одну компоненту j, которая не является частью комплексных чисел: q = (z1 + z2 j), j∉C.

Пусть z1​ и z2 имеют следующий вид:

Определим произведение i ∗ j=k, тогда итоговый вид кватерниона будет:

где a – действительная часть, а bi + cj + dk– мнимая часть кватерниона,

Числа a, b, c, d являются вещественными, а мнимые единицы i, j, k обладают свойством i2 = j2 = k2 = ijk = –1 [42]. Так как группа по умножению не коммутативна таблица умножения определена следующим образом (таблица 2.1).

Таблица 2.1. – Таблица умножения базисных кватернионов

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 1 | i | j | k |
| 1 | 1 | i | j | k |
| i | i | -1 | k | -j |
| j | j | -k | -1 | i |
| k | k | j | -j | -1 |

Мнимые компоненты можно использовать для представления трех декартовых единичных векторов i, j, k (рисунок 2.1).

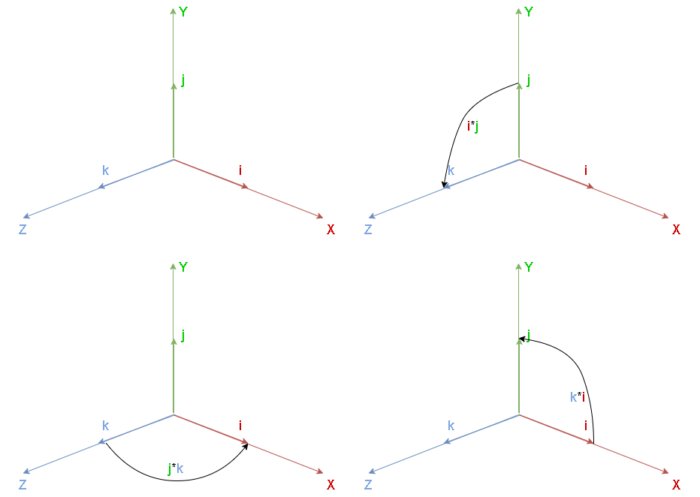


Рис 2.1. – Использование мнимых чисел как представление декартовых единичных векторов

Использование кватернионов обусловлено рядом существенных преимуществ перед альтернативными методами представления вращений, такими как углы Эйлера и матрицы вращений. Одной из наиболее известных проблем при использовании углов Эйлера является явление "складывания рамок" (gimbal lock) [43]. Это происходит, когда два из трех углов поворота совпадают, что приводит к потере одной степени свободы.

Матрицы вращений могут приводить к накоплению ошибок округления при выполнении последовательных преобразований.

При анимации движений человека часто требуется плавный переход между различными положениями суставов. Однако интерполяция углов Эйлера может приводить к нереалистичным движениям, так как углы не всегда изменяются линейно. Например, при попытке интерполировать углы между двумя состояниями сустава (например, между сгибанием и разгибанием руки) можно получить нежелательные "рывки" или "перекручивания", что делает движение менее естественным. Сустав может совершить лишние движения, такие как повороты влево, вправо или даже полный оборот на 360 градусов, прежде чем достигнет желаемого положения. Кватернионы устраняют подобные проблемы, предоставляя более надёжный и естественный способ представления вращений.

Рассмотрим использование кватернионов для поворота сустава на примере кисти вокруг оси вращения, представленной единичным вектором (α, β, γ), на градус θ. Необходимо, чтобы кисть переместилась в положение, указанное на рисунке 2.2. Для этого началом координат возьмем точку сустава локтя.

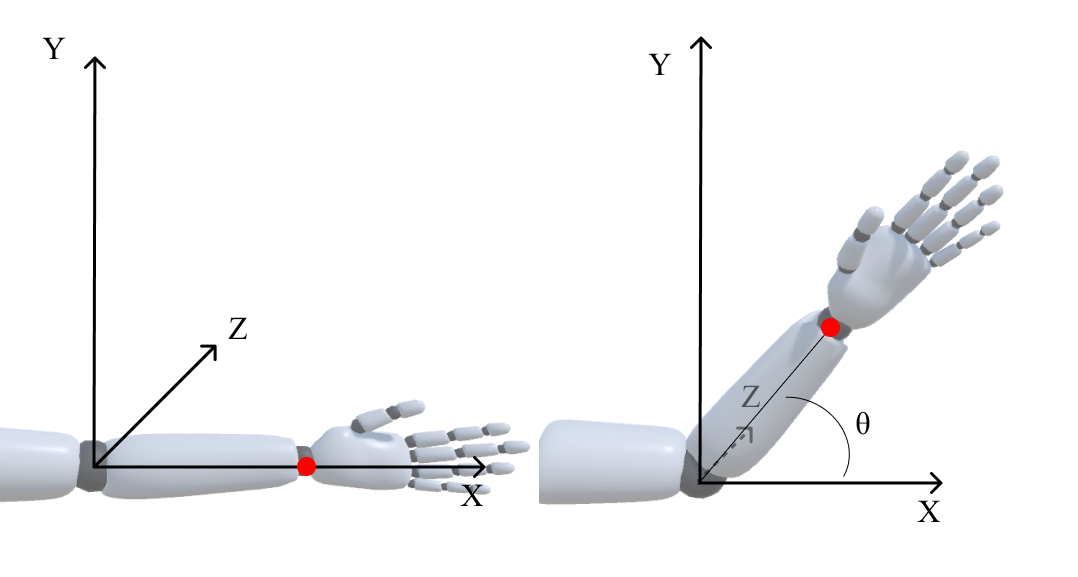


Рис 2.2. – Смещение кисти

Если q некоторый кватернион с длинной равной единице

сопряженный кватернион

то вращение кватерниона h описывается уравнением

Представим вектор от начала координат до сустава кисти (x, y, z) в виде кватерниона

Используя единичный вектор оси вращения (α, β, γ) и угол вращения θ составим кватернион

Тогда сопряженный кватернион имеет вид

Из результирующего кватерниона вычленяем вектор (m, n, l), являющийся результатом поворота

Аналогичным образом поворачивается сфера сустава локтя. Необходимо представить вектор (x, y, z) как радиус от центра сферы.

2.3. Формализация задачи математического моделирования движений

В области робототехники выделяется две ключевые задачи кинематики: прямая и обратная. Прямая задача включает в себя определение местоположения рабочего элемента манипулятора, основываясь на известных углах поворота его звеньев и заданной кинематической модели. Это позволяет вычислить координаты (X, Y, Z) конца манипулятора, исходя из текущих углов в суставах.

Обратная задача, с другой стороны, подразумевает определение нужных углов поворота звеньев манипулятора для того, чтобы его рабочий орган достиг необходимого положения. Это значит, что, зная заданную цель в пространстве, нужно установить конфигурацию манипулятора, которая позволит достичь этой цели [44].

Решение прямой задачи обычно является однозначным: для конкретного набора углов существует единственное положение рабочего органа. Однако обратная задача может иметь несколько решений или вовсе не иметь их, особенно если заданное положение недостижимо для конкретной конструкции манипулятора. Такая неоднозначность обусловлена тем, что разные комбинации углов могут приводить к одному и тому же положению конца манипулятора. В аналитических решениях это проявляется через математические выражения, содержащие, например, квадратные корни, что может приводить к нескольким возможным значениям углов.

Для устранения неоднозначности при решении обратной задачи часто вводят дополнительные ограничения, такие как пределы допустимых углов поворота, длины звеньев, необходимость обхода препятствий или строго заданные траектории движения. Тем не менее, даже при наличии таких ограничений, решение может отсутствовать, если, например, требуемое положение находится вне досягаемости манипулятора.​

Дополнительную сложность в решении обратной задачи вносит использование тригонометрических функций, поскольку их обратные функции могут иметь несколько значений, что увеличивает неопределенность решения. Поэтому обратная задача кинематики считается более сложной и критически важной в робототехнике, особенно при разработке систем управления и планирования движений манипуляторов [45].

Задача инверсной кинематики заключается в определении конфигурации суставов (например, углов поворота звеньев) манипулятора, необходимой для достижения заданного положения и ориентации его рабочего органа (энд-эффектора) в пространстве.

Пусть манипулятор имеет n степеней свободы, и его конфигурация описывается вектором суставных переменных

Положение и ориентация энд-эффектора в пространстве описываются вектором

где x,y,z – координаты, а – углы, определяющие ориентацию.​

Прямая кинематическая функция f связывает суставные переменные с положением энд-эффектора:​

Задача инверсной кинематики состоит в нахождении такого вектора Q, при котором энд-эффектор достигает желаемого положения Xd​.

Выводы по главе 2

В ходе исследования представлена декомпозиция задачи моделирования движений человека на подзадачи поворота суставов с использованием кватернионов и решения задачи инверсной кинематики. Также представлено формализованное представление процесса поворота суставов с использованием кватернионов для избежания гимбальной блокировки. Выполнена формализация задачи математического моделирования движений. Описана задача прямой и инверсной кинематики.

ГЛАВА 3. ПРОГРАММНАЯ РЕАЛИЗАЦИЯ

* 1. Разработка математической модели движения человека

Математическая модель движения человека строится на основе физической модели человека. Расстояния между суставами заданы как фиксированные параметры, соответствующие усредненным анатомическим размерам человеческого тела. Предлагается разделение суставов на взаимосвязанные группы с целью оптимизации работы алгоритма (рисунок 3.1). Такое разбиение позволяет сократить вычислительные затраты за счёт локализации расчетов внутри групп и обеспечивает гибкость при моделировании различных движений.

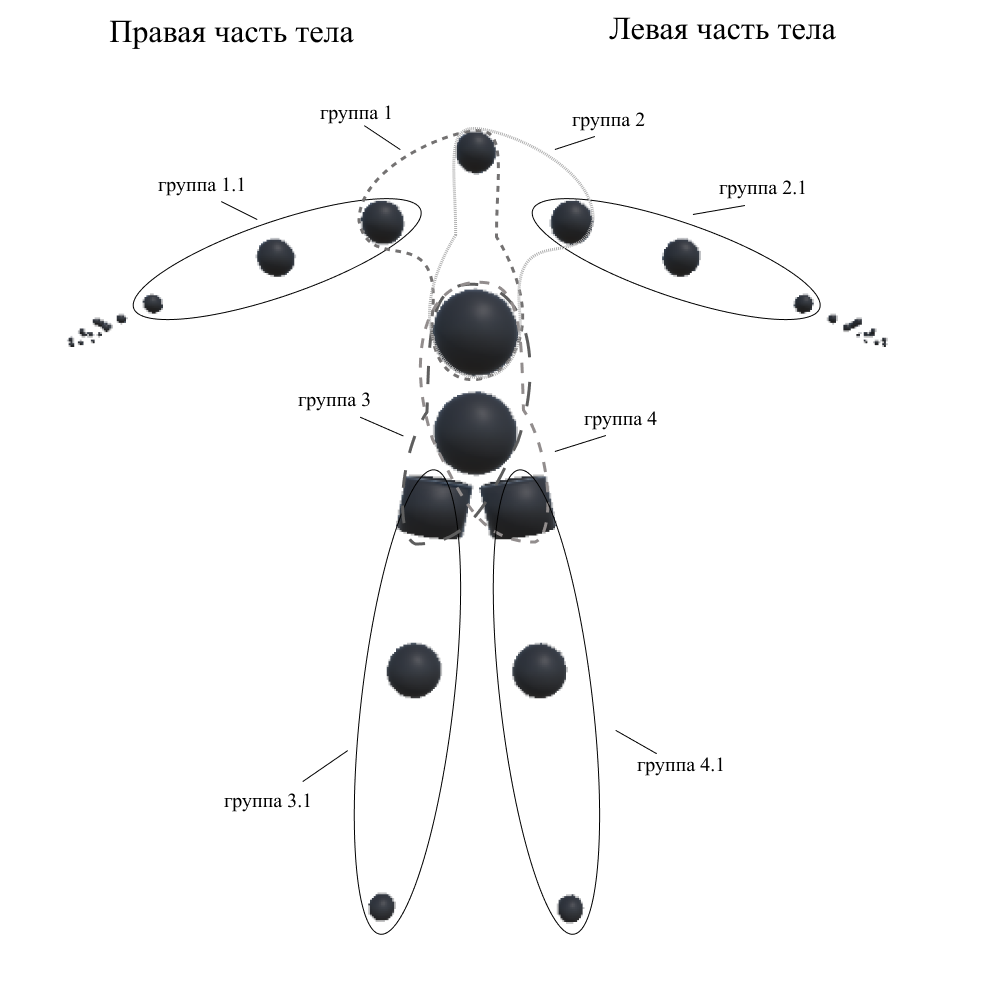


Рис 3.1. – Группы суставов

Центральным элементом модели является сустав спины, относительно которого строится вся структура взаимодействий. Первая группа включает суставы правого плеча, шеи и спины; вторая – левого плеча, шеи и спины. Третья и четвёртая группы охватывают правую и левую нижние части тела соответственно и включают суставы таза, спины и тазобедренные суставы.

К каждой из этих групп присоединяются подгруппы, описывающие локальные движения конечностей. Группа 1.1 объединяет суставы правой кисти, локтя и плеча, группа 2.1 – соответствующие суставы левой руки. Аналогично, группы 3.1 и 4.1 включают суставы правой и левой ноги соответственно: голени, колена и тазобедренного сустава.

Присоединение подгрупп осуществляется через общие суставы, принадлежащие сразу нескольким группам. Такая структура обеспечивает согласованность движений и передачу информации между различными частями тела в рамках единой модели.

Так как задача инверсной кинематики может иметь несколько решений определяются ограничения углов поворота суставов. Поскольку задача инверсной кинематики может иметь несколько решений, необходимо ввести ограничения, учитывающие анатомическую допустимость движений. Для каждого сустава определяются предельные значения углов вращения вокруг трёх осей, соответствующие биомеханическим характеристикам человеческого тела. Эти ограничения формализуются как неравенства для каждого угла , что исключает некорректные конфигурации и обеспечивает естественность движений.

Для предотвращения проблем, связанных с сингулярными точками при решении задачи инверсной кинематики, применяется псевдообратная матрица Якоби. Для повышения устойчивости вычислений и сглаживания шумов, возникающих из-за её использования, вводится метод экспоненциального сглаживания.

Кинематическая схема верхней конечности для задачи инверсной кинематики представлена на рисунке 3.2. для демонстрации работы метода на примере руки.

Исходное положение суставов в системе координат определяется уравнениями прямой кинематики:

,

где L1 и L2 – длины сегментов руки, θ1 и θ2 – углы поворота суставов, θ3 – угол вращения вокруг оси Z.

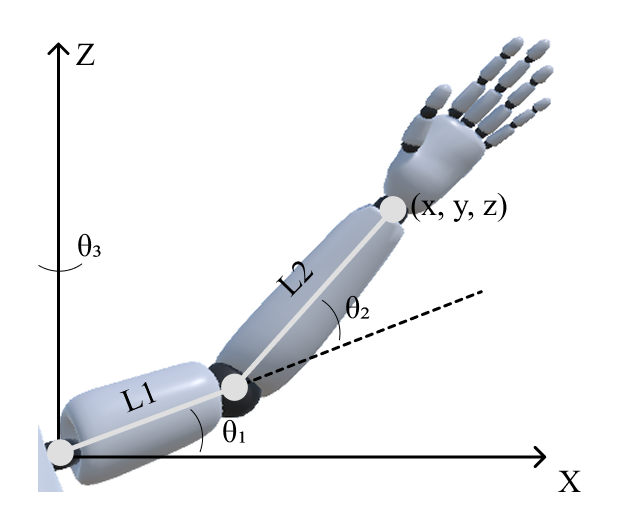


Рис 3.2. – Кинематическая схема верхней конечности для задачи инверсной кинематики

После определения целевой точки (xtarget, ytarget, ztarget), в которую должен переместиться манипулятор, производится расчет ошибки положения между текущими координатами (xcurr, ycurr,zcurr) и целевыми координатами:

Матрица Якоби описывает связь между изменениями углов суставов и положением конечного сустава. Она помогает определить, как небольшие изменения в углах суставов повлияют на положение конца конечности:

Для улучшения устойчивости вблизи сингулярных конфигураций используется сглаженная псевдообратная матрица Якоби:

где – транспонированная матрица Якоби, I – единичная матрица, λ – коэффициент сглаживания, который обычно выбирается малым (0,01 или 0,001).

Изменение углов суставов определяется следующим образом:

где – изменения углов суставов.

Для представления поворотов суставов используются кватернионы, что позволяет избежать проблем, связанных с углами Эйлера. Преобразование углового изменения в кватернион, который применяется для обновления ориентации суставов

где – норма вектора .

Рекурсивный алгоритм расчета координат суставов внутри группы прекращает выполнение, если текущая ошибка позиционирования становится меньше заданного порога или превышено допустимое количество итераций, что обеспечивает баланс между точностью вычислений и их вычислительной эффективностью.

* 1. Разработка алгоритма моделирования движений человека

Алгоритмы, использующие инверсную кинематику, находят широкое применение в симуляции движений человека, особенно в областях реабилитации, робототехники и анимации. Одним из важных процессов при решении таких задач является применение псевдообратной матрицы Якоби. Этот метод позволяет вычислять изменения углов суставов, необходимые для того, чтобы конечность достигла заданного положения. Для повышения стабильности и точности алгоритма используется метод сглаживания, который вводит регуляризацию, помогающую избежать неопределенности решений, возникающей при вырожденности матрицы Якоби. Данный подход особенно эффективен при работе с многосуставными моделями, где конфигурация системы может быть неопределенной.

На рисунке 3.3 представлен алгоритм работы подпрограммы расчета координат суставов внутри группы.

Работа подпрограммы начинается с расчета ошибки, которая отражает разницу между фактическим положением и заданной целью. Этот процесс важен для выявления необходимых корректировок углов суставов для достижения поставленной цели.

Затем строится матрица Якоби, которая описывает, как изменения углов суставов влияют на позицию конечности в пространстве. Для этого производятся вычисления частных производных координат x, y и z относительно каждого угла сустава. Полученная матрица впоследствии используется для обратного преобразования. На данном этапе применяется псевдообратная матрица Якоби с учетом сглаживания, что позволяет определить изменения углов суставов, минимизируя ошибку и обеспечивая стабильность решения.

После того как корректировки рассчитаны, программа обновляет показатели углов суставов. Если целевая точка не достигнута, алгоритм вновь выполняет указанные шаги, начиная с расчета текущей позиции и ошибки. Этот процесс продолжается до достижения необходимой точности.



Рис 3.3. – Подпрограмма расчета координат суставов внутри группы

Для упрощения вычислений и удобства управления движениями тела суставы организованы в группы. Такая группировка позволяет более эффективно моделировать кинематику человеческого тела. Каждая группа суставов имеет свою зону ответственности и взаимодействует с другими группами для обеспечения слаженного движения всего тела.

Первая группа формируется из суставов правой верхней части тела: правое плечо, шея и спина. К ней добавляется дополнительная группа суставов, включающая правую кисть, правый локоть и правое плечо, что обеспечивает интеграцию движений правой руки с остальным телом.

Аналогичным образом организуется вторая группа, которая охватывает левую верхнюю часть тела. Она состоит из левого плеча, шеи и спины. Затем ко второй группе присоединяется дополнительная группа суставов, включающая левую кисть, левый локоть и левое плеча. Эти суставы обеспечивают связь между движениями руки и остальной частью тела, позволяя координировать их для выполнения сложных действий. Таким образом, обе группы верхней части тела работают взаимосвязано, обеспечивая баланс и согласованность движений.

Третья группа формируется из суставов правой нижней части тела: правого тазобедренный сустава, таза и спины. К третьей группе присоединяется дополнительная группа суставов, включающая правую голень, правое колено и правый тазобедренный сустав. Эти суставы обеспечивают связь между правой ногой и остальным телом, способствуя координации движений и поддержанию стабильности.

Четвертая группа охватывает суставы левой нижней конечности: левый тазобедренный сустав, таз и спину. К данной группе дополнительно присоединяется группа суставов, состоящая из левой голени, левого колена и левого тазобедренного сустава. Эти суставы играют ключевую роль в обеспечении связи между нижней конечностью и верхней частью тела.

Важно отметить, что все описанные группы суставов взаимосвязаны и работают в единой системе. Группы суставов нижней части тела (третья и четвертая группы) не только управляют движениями ног, но и активно взаимодействуют с группами верхней части тела (первой и второй группами). Это взаимодействие обеспечивается за счет участия общих суставов, таких как таз, спина и шея, которые служат связующим звеном между различными частями тела. Благодаря такой организации движения становятся более плавными, естественными и эффективными.

При разбиении суставов на взаимосвязанные группы осуществляется рекурсивный перерасчет их координат и углов, что является важным условием для обеспечения естественности и плавности движений. Данный процесс предполагает последовательное прохождение по каждой группе суставов, где изменения в одном элементе цепи автоматически влияют на состояние связанных с ним соседних узлов. Благодаря рекурсии, алгоритм обрабатывает не только непосредственные связи внутри группы, но и распространяет корректировки на более удаленные суставы, сохраняя согласованность всей системы (рисунок 3.4).

В первую очередь, при смещении сустава производится перерасчет углов внутри той группы, к которой он принадлежит. Этот этап является основным и направлен на адаптацию движения в пределах текущей группы. Однако, если изменение в одной группе оказывает влияние на другие части тела или делает невозможным сохранение целостности всей системы, активируется перерасчет углов в соседних группах. Такая последовательность действий гарантирует, что движения остаются согласованными и соответствуют анатомическим возможностям человека.

Например, если пользователь перемещает манипулятор, отвечающий за левую кисть, и это приводит к ограничению подвижности в области плеча или спины, система учитывает эти изменения и корректирует углы суставов в смежных группах, таких как шея или правая рука. Аналогичный принцип применяется и к нижним конечностям: при смещении манипулятора, управляющего голенью, может потребоваться перерасчет углов в тазобедренном суставе, тазе и даже в группах верхней части тела для поддержания равновесия и координации.

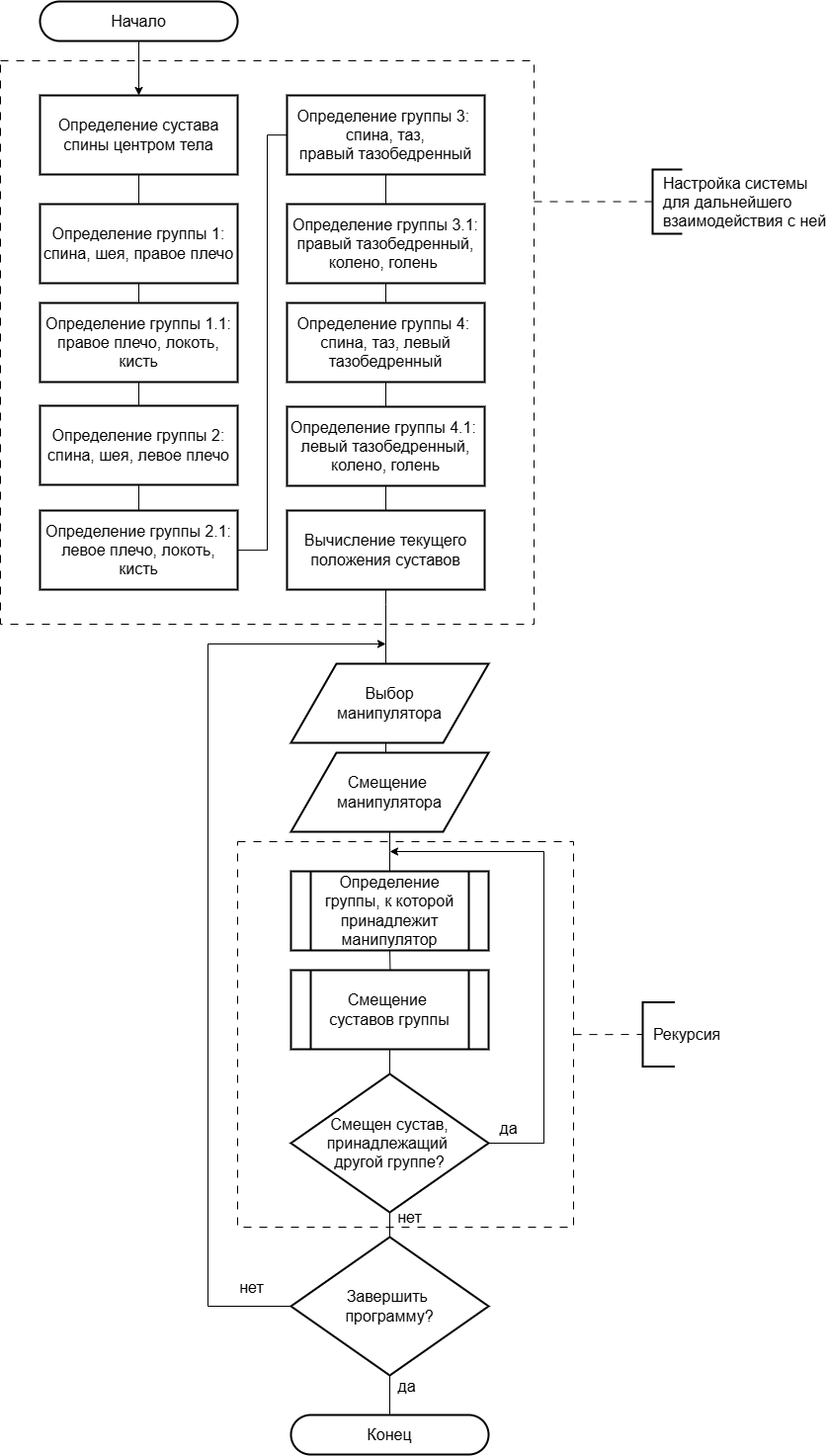


Рис 3.4. – Схема рекурсивного расчета координат суставов

Этот метод дает возможность эффективно воспроизводить движения человека, принимая во внимание индивидуальные характеристики кинематики и обеспечивая стабильное функционирование системы, даже в сложных конфигурациях.

* 1. Программная реализация

В ходе выполнения работы разработана программа, предназначенная для интерактивного взаимодействия пользователя с моделью человека с целью моделирования кинематических движений. Интерфейс системы (рис. 3.5) реализован в виде графической среды, обеспечивающей интуитивное управление элементами модели.

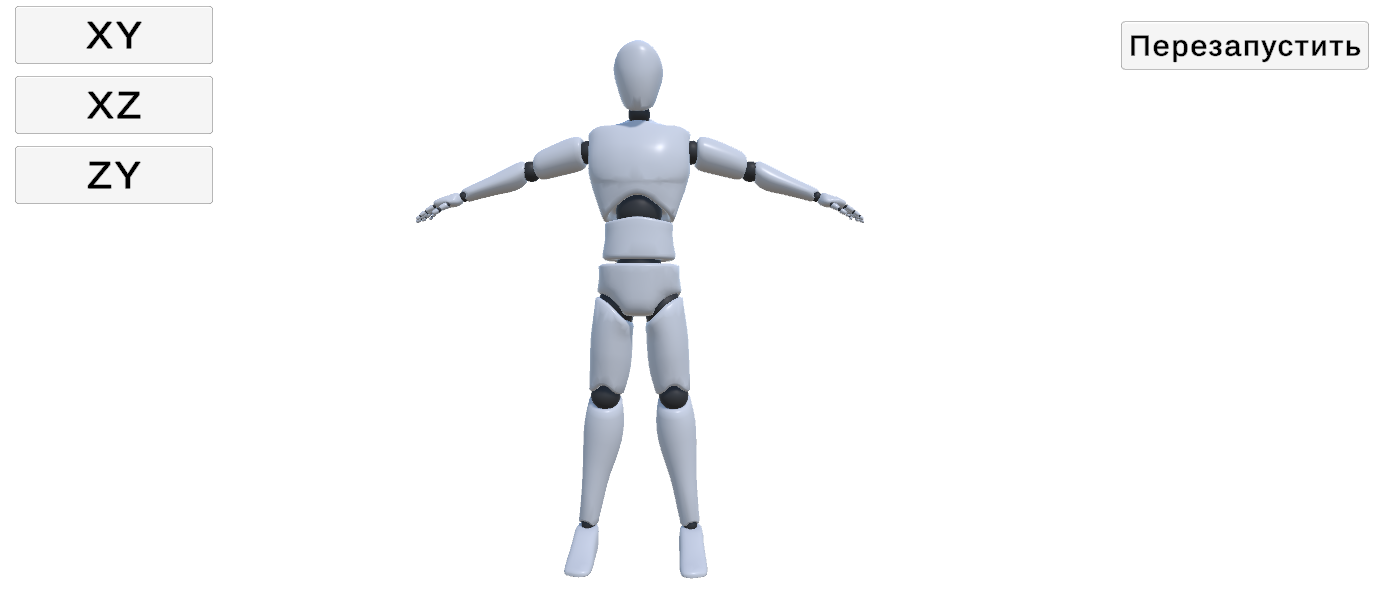


Рис 3.5. – Интерфейс программы

Основными элементами системы являются манипуляторы – суставы, доступные для перемещения. Данные элементы визуализированы на рисунке 3.6 и поддерживают интерактивное перемещение посредством удержания левой клавиши мыши. Для повышения удобства навигации в трехмерном пространстве реализована функция вращения камеры вокруг модели при удержании правой клавиши мыши, что позволяет пользователю динамически изменять ракурс наблюдения.

На данный момент расстояния между суставами заданы как фиксированные параметры, соответствующие усредненным анатомическим размерам человеческого тела. Для унификации и упрощения восприятия эти параметры обозначены конкретными наименованиями, которые представлены на рисунке 3.7.

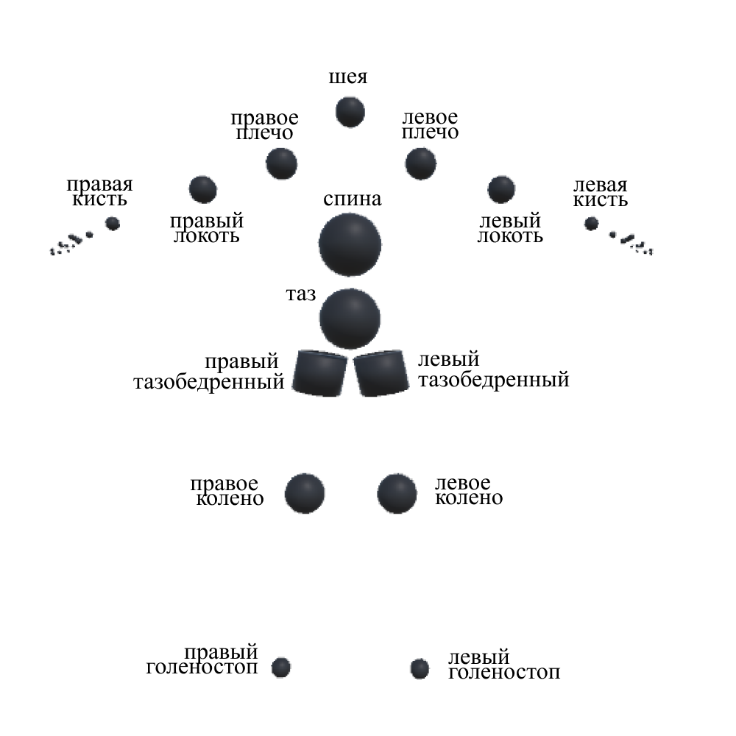


Рис 3.6. – Суставы для перемещения (манипуляторы)

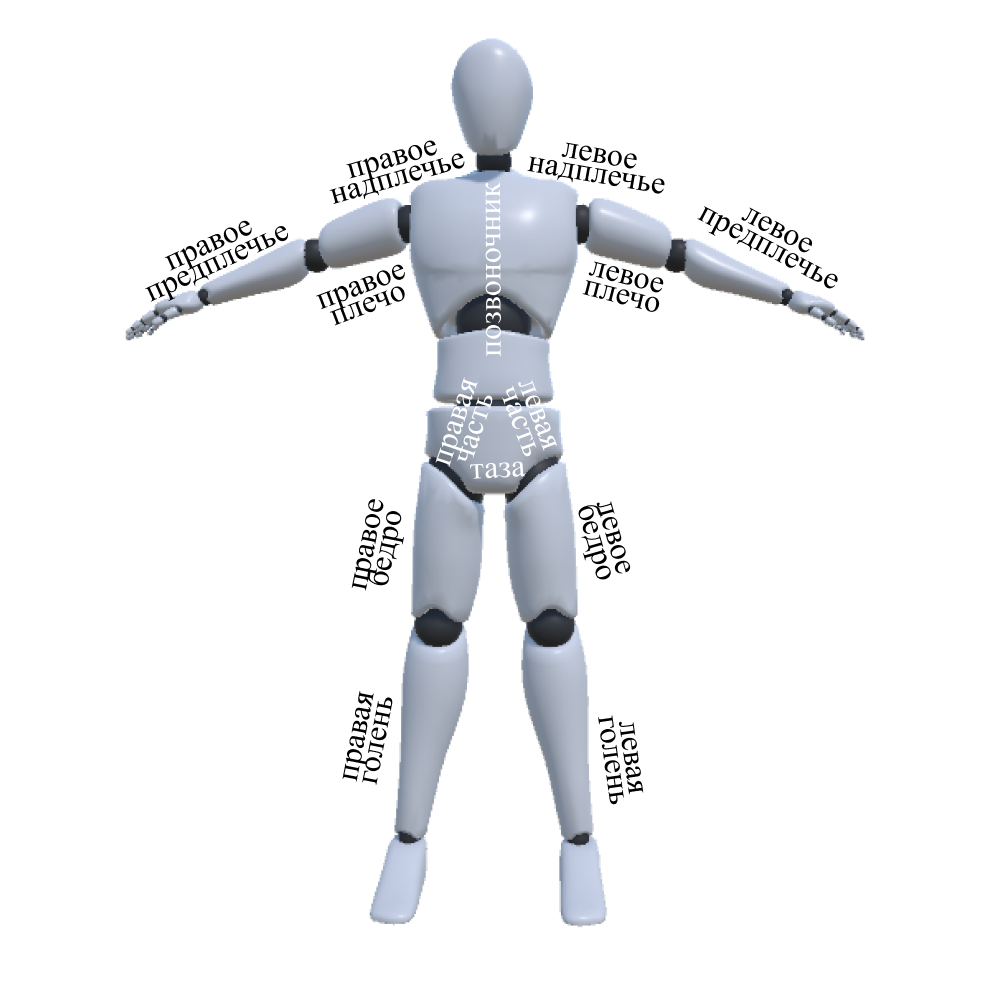


Рис 3.7. – Расстояния между суставами

Для удобного перемещения манипуляторов в программе представлены кнопки для ограничения их движения в плоскостях XOY, XOZ, ZOY соответственно (рисунок 3.8)

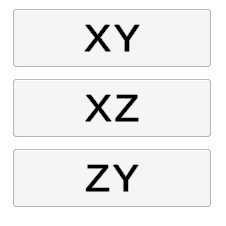


Рис 3.8. – Интерфейс

Оси координат расположены следующим образом относительно человека: ось Y от стоп к темени, ось Z от живота к спине, ось X от правой части тела к левой (рисунок 3.9).

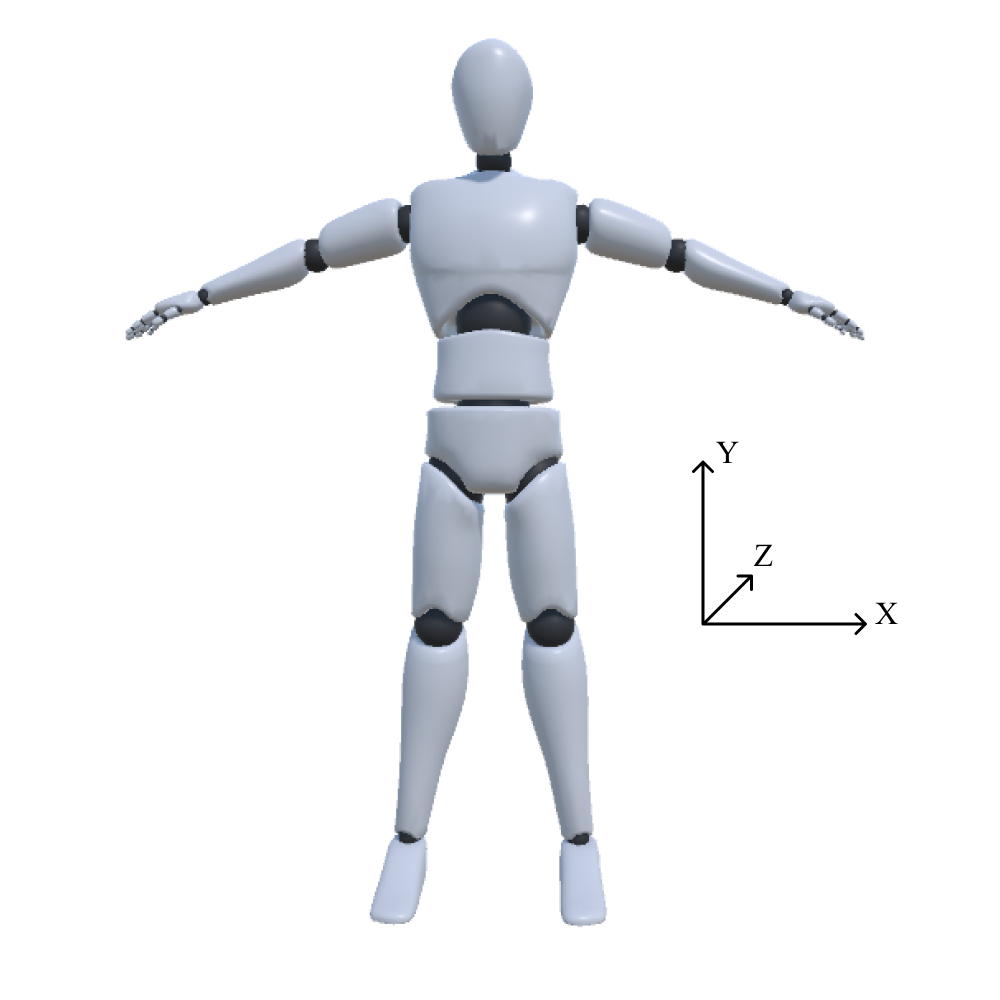


Рис 3.9. – Расположение человека в пространстве

Программа позволяет взаимодействовать с человеком с помощью перемещения манипуляторов. Таким образом моделируются различные движения. Например, верхними конечностями, можно совершать круговые движения, ограничив его по осям YZ (рис. 3.10), или поднимать и опускать, сгибать в локте (рис. 3.11).

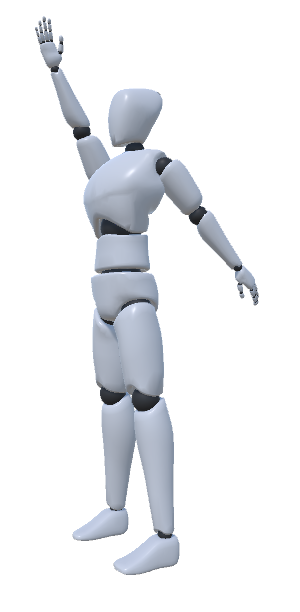
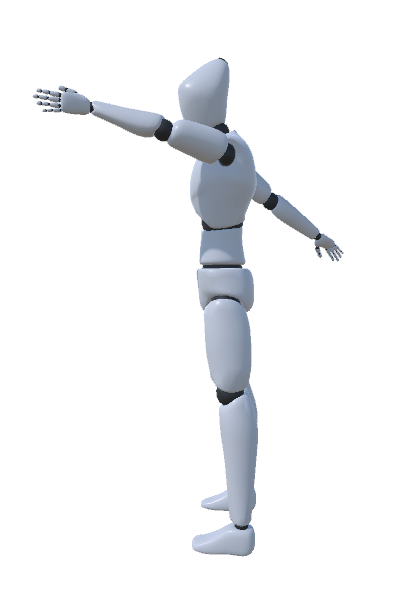


Рис 3.10. – Круговые вращения руками

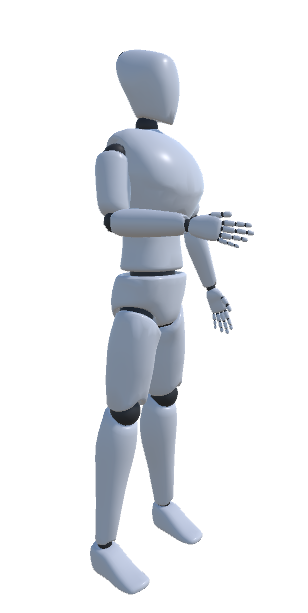
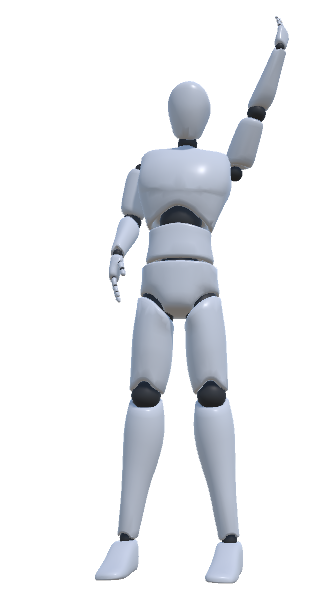


Рис 3.11. – Поднятие и опускание рук

Используя, манипуляторы нижних конечностей можно воспроизвести махи ногами вперед или в бок (рис. 3.12), а также поднятие колена (рис. 3.13).

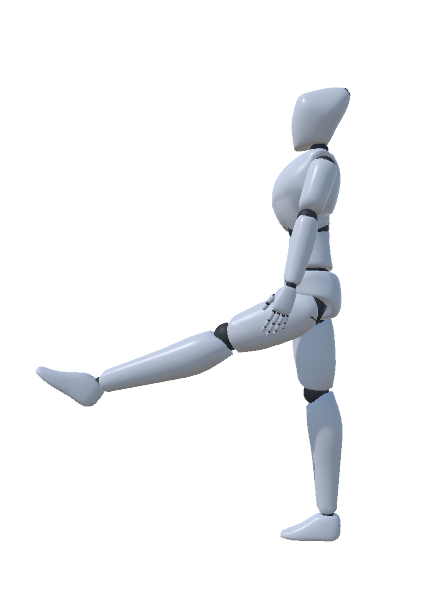
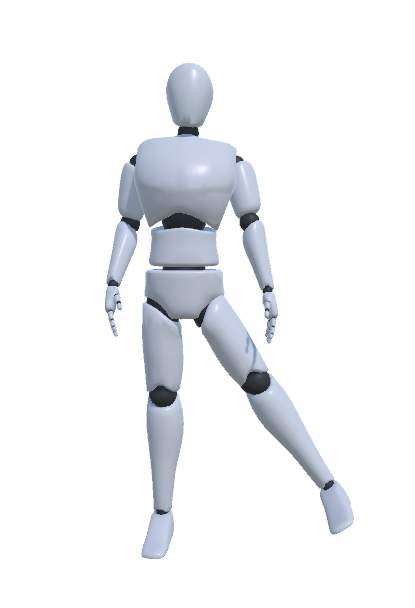


Рис 3.12. – Махи ногами

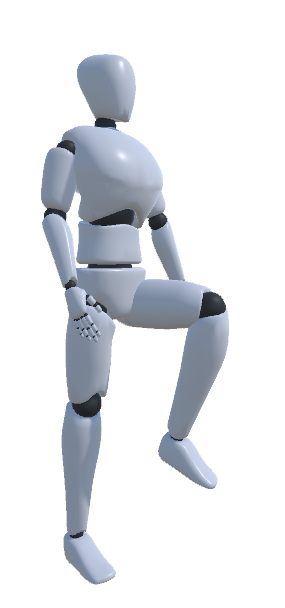
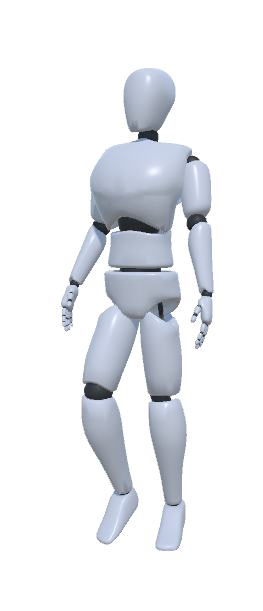
 

Рис 3.13. – Поднятие колена

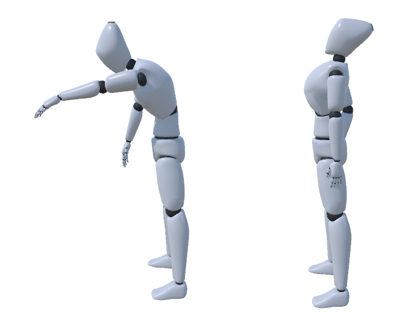


Рис 3.14. – Пример работы программы

Как можно видеть на рисунке 3.14 при положении манипулятора, где невозможно дальнейшее смещение суставов текущей группы, происходит изменение координат смежных групп.

Выводы по главе 3

В ходе исследования описана разработка математической модели движения человека, которая включает в себя разбиение суставов на взаимосвязанные группы для рекурсивного расчета координат суставов всего тела, определение ограничений поворота суставов, решение задачи инверсной кинематики с использованием псевдообратной матрицы Якоби и метода экспоненциального сглаживания для поиска углов поворота суставов рычага и поворота суставов на найденный угол с использованием кватернионов. Также описана разработка алгоритма моделирования движений человека, в котором обозначены последовательность шагов для решения задачи и рекурсивная подпрограмма перерасчета координат суставов внутри группы. Описана программная реализация с возможностями интерфейса для взаимодействия пользователя с моделью человека. Кроме того, описаны некоторые из возможных для моделирования движения.

ГЛАВА 4. ЭКСПЕРИМЕНТАЛЬНОЕ ПОДТВЕРЖДЕНИЕ РЕЗУЛЬТАТОВ ИССЛЕДОВАНИЯ

* 1. Верификация и сравнение модели с помощью метрики качества

Для оценки качества траектории использованы такие безреференсные метрики как MME (Mean Map Entropy), MPV (Mean Plane Variance) и MOM (Mutually Orthogonal Metric). Эти метрики позволяют анализировать траектории без необходимости сравнения с эталонными данными. Безреференсные метрики основаны на численном измерении шума и консистентности карты – множества облаков точек, расположенных согласно их позициям в траектории.

В данных метриках энтропия h для точки  вычисляется по формуле:

где – выборочная ковариация точек в -окрестности точки .

Mean Map Entropy (MME) вычисляется через усреднение локальной энтропии, которая отражает неопределенность положения соседних точек относительно друг друга. Чем выше значение MME, тем менее предсказуема траектория, что может указывать на шумы, артефакты или ошибки в её построении. Эта метрика особенно чувствительна к резким изменениям направления или разрывам в движении.

Mean Plane Variance (MPV) оценивает среднюю дисперсию точек относительно аппроксимированной плоскости в локальной окрестности. Предполагается, что большинство окружающих поверхностей являются плоскими, поэтому отклонения точек от этих плоскостей могут свидетельствовать о шуме или ошибках в регистрации. Низкие значения MPV указывают на более точное и согласованное построение карты.

где – наименьшее собственное значение выборочной ковариации точек в -окрестности точки .

Mutually Orthogonal Metric (MOM) фокусируется на оценке качества траектории, анализируя согласованность точек, расположенных на взаимно ортогональных поверхностях. Метрика измеряет среднюю дисперсию точек относительно соответствующих плоскостей в этих областях. Поскольку ортогональные векторы имеют нулевое скалярное произведение, минимальные значения MOM соответствуют плавным и предсказуемым траекториям, тогда как высокие значения указывают на хаотичные изменения направления.

Результатом метрики является среднее арифметическое значений метрики MPV по каждому из направлений предварительно извлеченного ортогонального базиса.

Для верификации точности модели, описывающей движения человеческого тела, применяется специальная метрика качества. Данная метрика основана на прямом сравнении траекторий быстрого и медленного движения. Траектория медленного движения сустава характеризует плавные перемещения. Данная траектория сформирована в режиме, обеспечивающем максимальную стабильность. Траектория быстрого движения сустава имитирует динамичные движения и отличается сжатым временным интервалом.

Метрика точности определяется как разность между траекторией быстрого движения сустава и медленного движения по модулю:

где траектория быстрого движения сустава, траектория медленного движения сустава.

* 1. Ход проведения эксперимента

Для оценки работы модели и сбора необходимых данных проведен эксперимент. В ходе исследования записывается положение суставов кисти и локтя левой руки. При проведении эксперимента проведены следующие действия:

1. Ограничить возможность движения манипулятора по оси XY, нажав соответствующую кнопку. Затем, зажав левой клавишей мыши на манипуляторе (кисть), вести руку вверх.

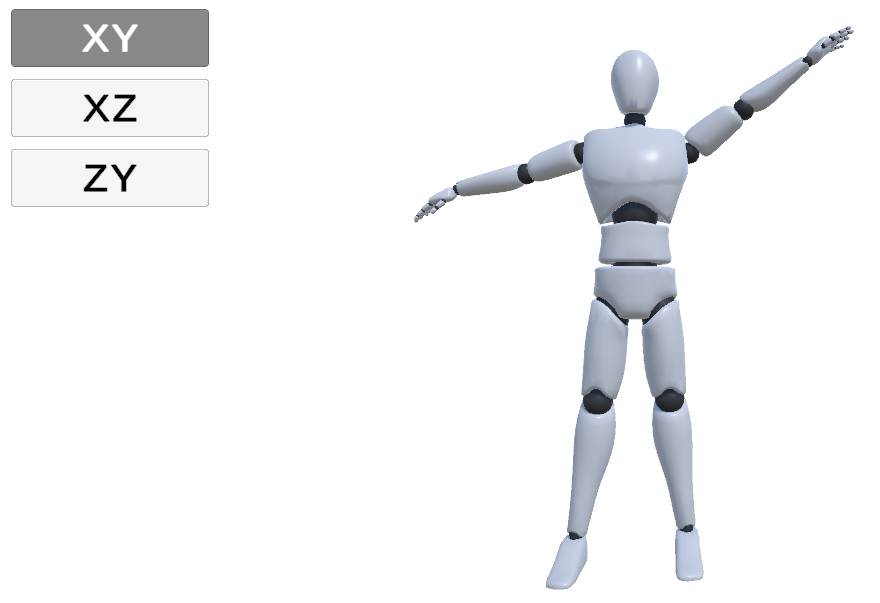


Рис 4.1. – Движение руки вверх

1. Для удобства перемещения манипулятора, зажав правую клавишу мыши, расположить человека как показано на рисунке 4.2.

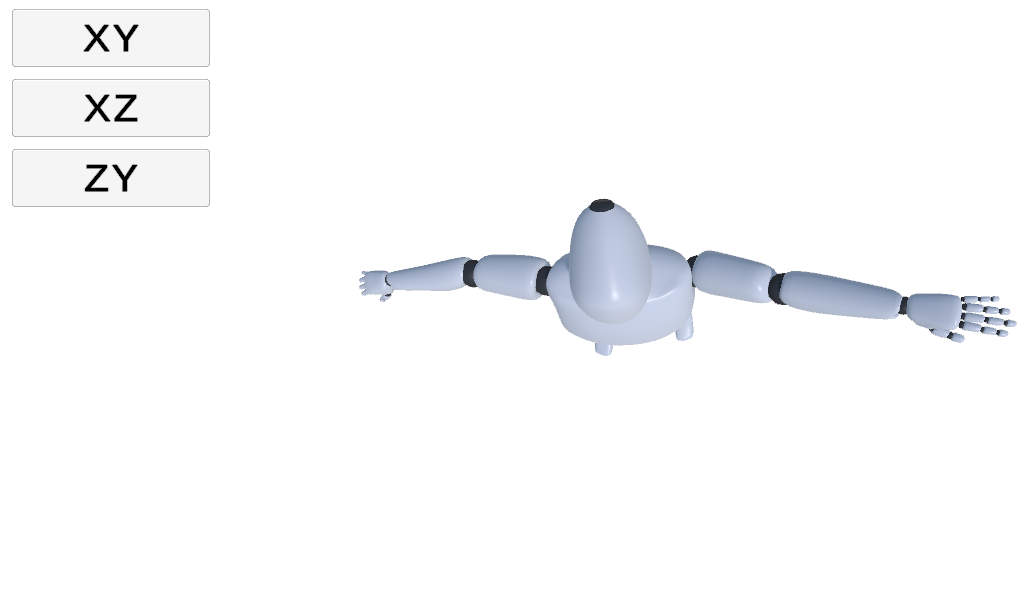


Рис 4.2. – Вид сверху

1. Ограничить возможность движения манипулятора по оси XZ. Вести руку назад.

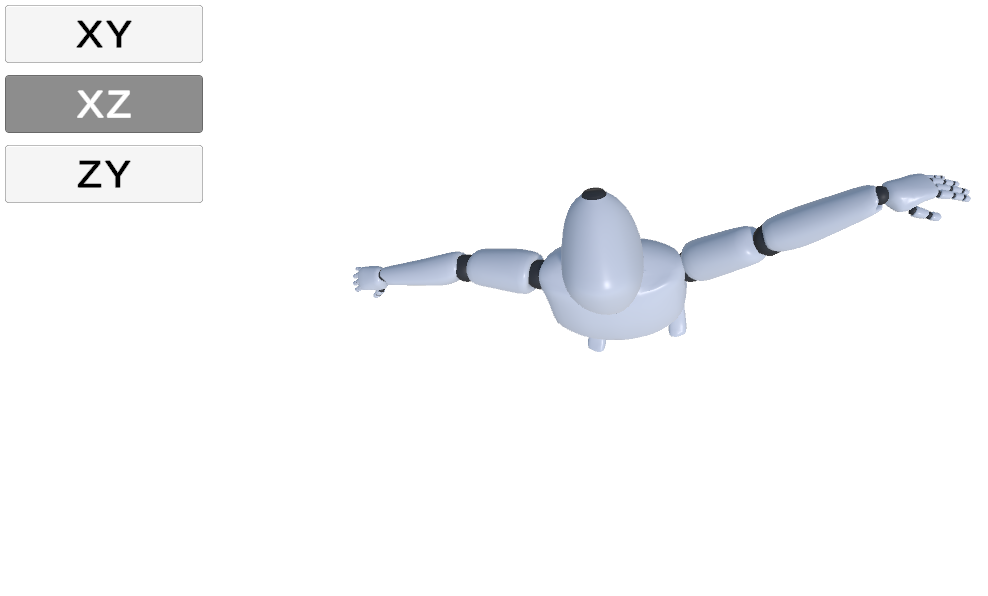


Рис 4.3. – Движение руки назад

1. Расположить человека боком к камере и ограничить движение манипулятора по оси ZY.

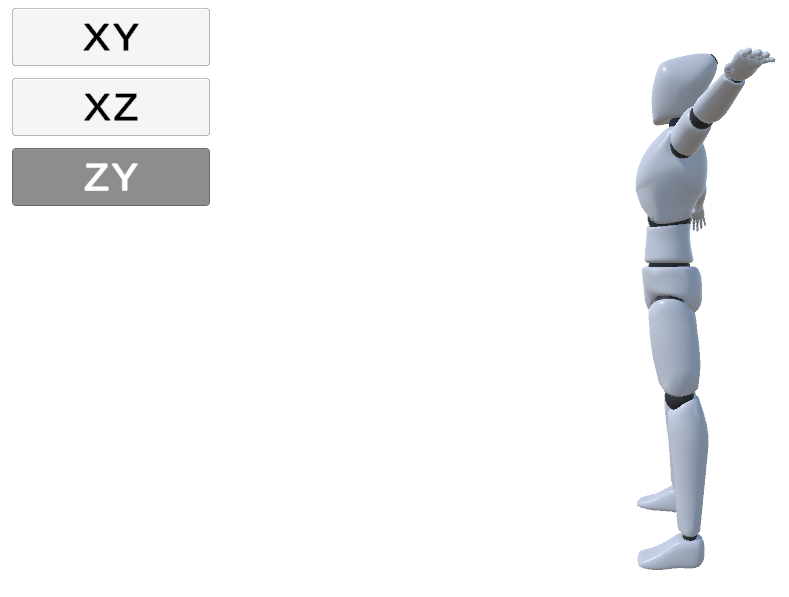


Рис 4.4. – Вид сбоку

1. Опустить руку вниз.

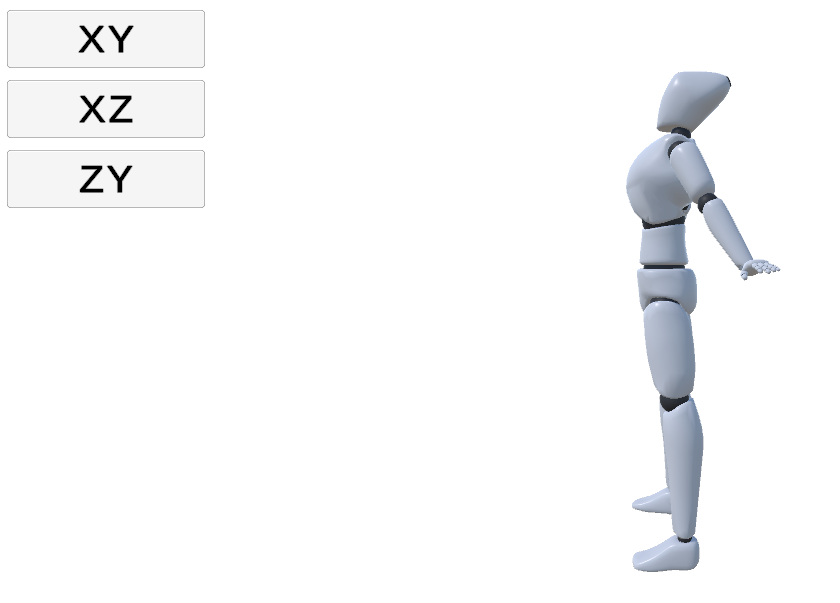


Рис 4.5. – Движение руки вниз

1. Вернуться к виду сверху, описанном в пункте 2, ограничить движение по оси XZ и двигать руку ближе к телу.

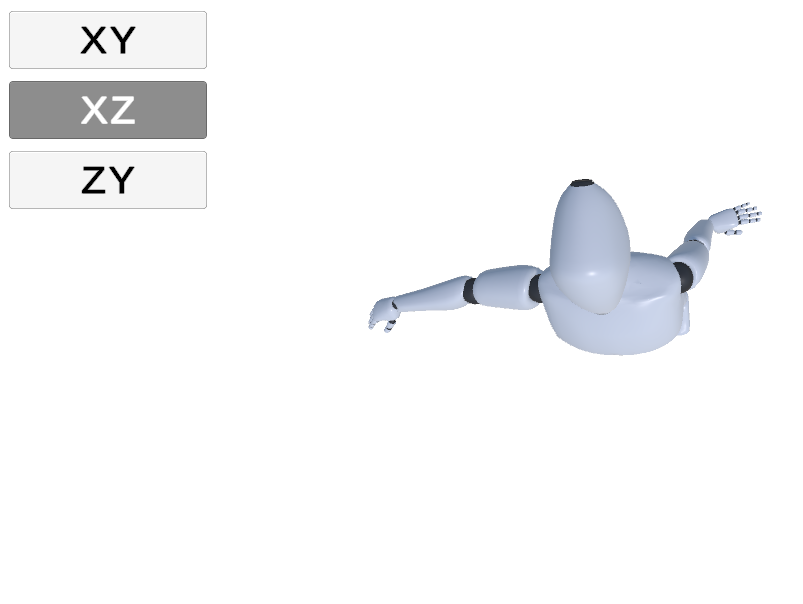


Рис 4.6. – Движение руки к телу

1. Повторно выполнить пункт 4 и переместить манипулятор по диагонали (вперед и вверх).

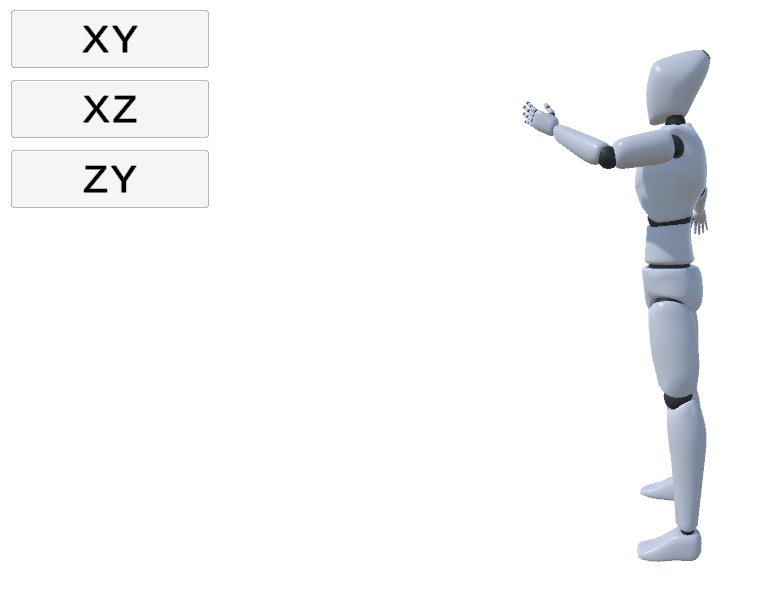


Рис 4.7. – Движение руки вперед и вверх

Для оценки модели эксперимент проведен дважды. В первом случае следует выполнять плавные и медленные перемещения манипулятора, что позволит зафиксировать траекторию с минимальным влиянием ошибок и даст возможность детально проанализировать точность воспроизведения заданных движений.

Во втором цикле эксперимента необходимо увеличить скорость перемещения манипулятора, тем самым имитируя более динамичные условия функционирования системы.

Если эксперимент не удается провести в соответствии с инструкцией, имеется кнопка «Перезапустить» для сброса счетчика времени и положения тела.

* 1. Обоснование достоверности полученных результатов

Для наглядной оценки работы модели построен график, на котором представлены траектории быстрого и медленного движения кисти и локтя (рис. 4.8, рис. 4.9). Стоит отметить, что на протяжении всей траектории расстояние между локтем и кистью оставалось неизменным, следовательно движение воспроизводится корректно без разрыва скелета.



Рис 4.8. – Траектория движения

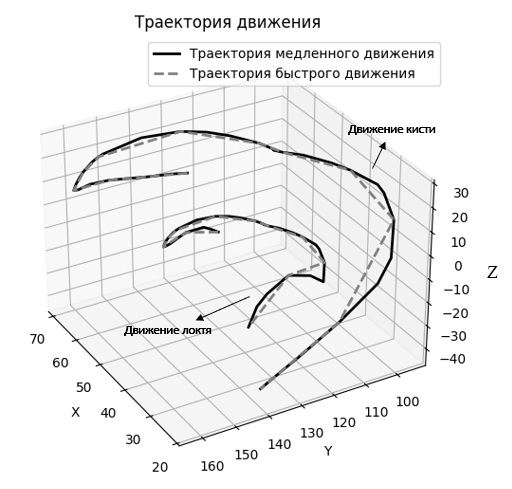


Рис 4.9. – Траектория движения

Важной частью этой оценки стало исследование графика отклонений, который демонстрирует отклонение траектории быстрого движения от медленного (рис. 4.10). Для повышения достоверности эксперимента выбраны ключевые контрольные точки, в которых координаты должны совпадать как в режиме медленного, так и быстрого перемещения, поэтому отклонение в этих точках равно нулю. Это позволило обеспечить сопоставимость данных в разных условиях.

Для объективного сравнения отклонений между траекториями медленного и быстрого движения траектории движения манипулятора приведены к единому масштабу. Медленная траектория, характеризующая плавные перемещения, сформирована в режиме, обеспечивающем максимальную стабильность, тогда как быстрая траектория, имитирующая динамичные движения, изначально отличалась сжатым временным интервалом. Для обеспечения сопоставимости точек обеих траекторий определенная точка ускоренной траектории соотносилась с соответствующей фазой замедленного движения, что позволило выровнять их по позициям.

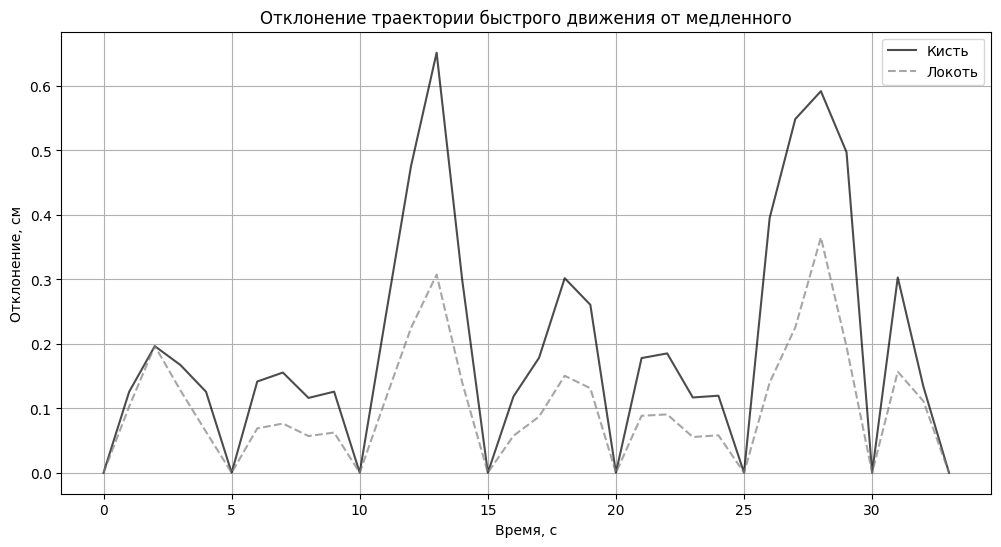


Рис 4.10. – Точность расчетной траектории

Среднее отклонение составило 0,15 см. Максимальное зафиксированное отклонение не превышает 0,65 см и наблюдалось в моменты наиболее резких изменений траектории, однако даже этот показатель остается в пределах допустимых значений для большинства практических задач. После усреднения полученных отклонений рассчитана общая точность модели, составившая 99,68%. Этот результат подтверждает, что разработанная модель способна точно воспроизводить траектории движений даже при высоких скоростях.

Безреференсные метрики использовались для оценивания качества траектории медленного движения. Результаты и ожидаемые значения представлены в таблице 4.1. Для более точного анализа траектории движение записывалось с частотой 60 Гц. Ожидаемые значения для медленного движения определены исходя из типичных характеристик контролируемого движения при указанной частоте записи.

Таблица 4.1. – Метрики качества

|  |  |  |  |
| --- | --- | --- | --- |
| Метрика | Ожидаемое значение (см) | Фактическое значение для кисти (см) | Фактическое значение для локтя (см) |
| MME | < 0,1 | 0,03833 | 0,03955 |
| MPV | < 2 | 2,59555 | 2,36242 |
| MOM15 пиков | < 1 | 0,79697 | 1,53955 |

Траектория кисти демонстрирует высокую степень плавности и стабильности. Значения метрик находятся в пределах ожидаемых диапазонов, что говорит о равномерном и контролируемом характере движения. Максимальный скачок немного превышает ожидаемый уровень, однако это может быть связано с локальным изменением направления или небольшими колебаниями в процессе выполнения задачи. В целом, динамика кисти соответствует заявленному режиму – медленное, аккуратное движение без выраженных рывков. На графике пиков скорости видно, что большинство скачков имеют малую величину, однако один выброс выделяется, что соответствует зарегистрированному максимальному скачку (рис. 4.11).

Анализ траектории локтя также показывает гладкое движение, однако отмечаются более выраженные локальные изменения скорости. Некоторые значения метрик выходят за рамки ожидаемых, особенно среднее по максимальным скачкам (MOM), что указывает на наличие участков с повышенной активностью. Это связано с особенностями решения обратной кинематики, необходимостью компенсации изменений ориентации кисти или переходом через зоны, где манипулятор менее устойчив. На графике пиков скорости локтя можно наблюдать несколько выраженных всплесков, которые указывают на переходы через особые точки или перестройку конфигурации плеча и предплечья (рис. 4.12).



Рис 4.11. – График пиков скорости для сустава кисти



Рис 4.12. – График пиков скорости для сустава кисти

Анализ траектории показал, что движение соответствует корректному медленному перемещению с высокой стабильностью и минимальными скачками.

Выводы по главе 4

В ходе исследования описаны существующие метрики для оценки траектории движения, такие как MME, MPV, MOM. Также предложено сравнение траекторий быстрого и медленного движения для оценки устойчивости модели в динамических условиях. Описан ход проведения эксперимента для сбора необходимых данных, используемых при оценке достоверности модели. Приведены результаты эксперимента, которые показывают, что разработанная модель функционирует корректно и демонстрирует устойчивость в условиях динамических нагрузок. Анализ метрик подтвердил их соответствие допустимым пределам: отклонения не превышают критических значений, а распределение ошибок равномерно по всей траектории. Общая точность модели, рассчитанная на основе усредненных данных, составила 99,68%

ЗАКЛЮЧЕНИЕ

Реализация движений человека с использованием инверсной кинематики (ИК) открывает широкие возможности в таких областях, как робототехника, медицина и реабилитация, виртуальная и дополненная реальность, компьютерная анимация и видеоигры.

Инверсная кинематика играет ключевую роль в робототехнике, особенно при управлении гуманоидными роботами, стремящимися воспроизводить движения человека [46].

Инверсная кинематика позволяет вычислять необходимые углы поворота суставов для достижения заданного положения конечностей. Это особенно важно при выполнении задач, требующих высокой точности, таких как захват объектов или взаимодействие с окружающей средой. Например, при телеприсутствии ИК обеспечивает синхронное движение робота с действиями оператора, что позволяет эффективно выполнять удалённые операции.

Для гуманоидных роботов поддержание равновесия является критически важным. Инверсная кинематика используется для расчёта положений тела, обеспечивающих стабильность при ходьбе, беге или выполнении других динамических действий. Современные методы, такие как глубокое обучение, помогают роботам адаптироваться к изменяющимся условиям и сохранять устойчивость в реальном времени.

Инверсная кинематика применяется для анализа и воспроизведения человеческих движений, что позволяет роботам имитировать жесты, походку и другие действия человека. Это находит применение в области анимации, реабилитации и взаимодействия человек-робот. Использование данных о движениях человека позволяет роботам выполнять задачи более естественно и эффективно [47].

Инверсная кинематика способствует разработке систем, способных обучаться и адаптироваться к новым задачам. С помощью методов машинного обучения роботы могут улучшать свои движения, оптимизируя траектории и повышая эффективность взаимодействия с окружающей средой. Это особенно полезно в условиях, где заранее запрограммированные действия недостаточны [48].

Инверсная кинематика играет ключевую роль в современной медицинской реабилитации, особенно в восстановлении двигательных функций верхних и нижних конечностей [49].

ИК используется для управления экзоскелетами, предназначенными для восстановления движений у пациентов после инсульта или травм. Модели, учитывающие комфорт суставов, энергопотребление и безопасность, позволяют экзоскелетам адаптироваться к индивидуальным особенностям пациента, обеспечивая более эффективную терапию.

Инверсная кинематика применяется для точного анализа движений пациента, что важно при диагностике и мониторинге прогресса в реабилитации. Использование оптических систем захвата движения и алгоритмов ИК позволяет оценивать углы суставов и координацию движений, что способствует более точной настройке реабилитационных программ.

ИК интегрируется в системы телереабилитации, позволяя пациентам выполнять упражнения под удалённым контролем специалистов. Использование доступных устройств, таких как Kinect, в сочетании с программами анализа движений, делает реабилитацию более доступной и персонализированной [50].

Инверсная кинематика также применяется в реабилитации детей с двигательными нарушениями. Анализ движений с использованием ИК помогает в оценке эффективности терапевтических вмешательств и адаптации программ лечения под индивидуальные потребности ребёнка.

Инверсная кинематика используется в разработке реабилитационных игр, которые делают процесс восстановления более увлекательным и мотивирующим для пациентов. Такие игры адаптируются к уровню двигательных способностей пациента, обеспечивая оптимальную нагрузку и стимулируя прогресс [51].

​Инверсная кинематика широко применяется в компьютерной анимации, видеоиграх, а также в виртуальной (VR) и дополненной реальности (AR), обеспечивая реалистичное и интерактивное поведение персонажей и аватаров [52].

ИК позволяет аниматорам управлять конечными точками скелетных структур (например, руками или ногами), автоматически вычисляя положения промежуточных суставов. Это упрощает процесс анимации и обеспечивает более естественные движения персонажей.

Инверсная кинематика обеспечивает корректное размещение конечностей персонажа на поверхности, например, ноги автоматически адаптируются к неровностям земли, что делает движения более правдоподобными.

Аниматоры могут задавать положение конечной точки (например, руки), и ИК автоматически рассчитывает необходимые углы в суставах, что ускоряет процесс создания анимаций.

Инверсная кинематика позволяет персонажам реагировать на изменения в окружении в реальном времени, например, автоматически хвататься за объекты или избегать препятствий [53].

В VR и AR ИК используется для создания реалистичных аватаров, которые точно отражают движения пользователя, даже при ограниченном количестве отслеживаемых точек.

Даже при наличии только трекеров на голове и руках, ИК позволяет реконструировать положение всего тела пользователя, создавая ощущение полного присутствия в виртуальном пространстве.

Инверсная кинематика применяется в обучающих VR-приложениях, позволяя пользователям взаимодействовать с виртуальными объектами так, как они бы делали это в реальной жизни, что повышает эффективность обучения.

ИК обеспечивает более естественные и предсказуемые движения аватаров, что снижает вероятность возникновения дискомфорта или "киберболезни" у пользователей.

В AR-приложениях ИК позволяет виртуальным персонажам или объектам взаимодействовать с реальным окружением, например, правильно размещаться на поверхности или реагировать на действия пользователя [54].

Таким образом, применение инверсной кинематики в моделировании человеческих движений имеет широкое распространение и значительное влияние на развитие технологий в различных сферах.

# СПИСОК ЛИТЕРАТУРЫ

|  |  |
| --- | --- |
| 1. | «The AnyBody Modeling System,» [В Интернете].  Available: https://www.anybodytech.com/software/anybodymodelingsystem/.  [Дата обращения: 02 04 2025]. |
| 2. | «AnyBody modeling system,» [В Интернете].  Available: https://www.sciencedirect.com/science/article/abs/pii/  B9780128239131000075. [Дата обращения: 11 04 2025]. |
| 3. | «Computational Biomechanics Tools,» [В Интернете]. Available: https://www.imagwiki.nibib.nih.gov/content/computational-biomechanics-tools. [Дата обращения: 02 04 2025]. |
| 4. | «A Software Tool for Faster Development of Complex Models of  Musculoskeletal Systems and Sensorimotor Controllers in Simulink TM,»  [В Интернете]. Available: https://www.researchgate.net/publication/242659587  \_A\_Software\_Tool\_for\_Faster\_Development\_of\_Complex\_Models\_of\_  Musculoskeletal\_Systems\_and\_Sensorimotor\_Controllers\_in\_Simulink\_TM. [Дата обращения: 11 04 2025]. |
| 5. | «Welcome to OpenSim,» [В Интернете].  Available: https://opensimconfluence.atlassian.net/wiki/spaces/OpenSim  /pages/53089821/Welcome+to+OpenSim. [Дата обращения: 02 04 2025]. |
| 6. | «Interactive Software-based Modeling for Gait Analysis of,» [В Интернете].  Available: https://scholarhub.ui.ac.id/cgi/viewcontent.cgi?  article=1542&context=mjt. [Дата обращения: 11 04 2025]. |
| 7. | «blender.org,» [В Интернете]. Available: https://www.blender.org.  [Дата обращения: 02 04 2025]. |
| 8. | «Damped Track Constraint,» [В Интернете].  Available: https://docs.blender.org/manual/en/latest/animation/constraints/  tracking/damped\_track.html. [Дата обращения: 11 04 2025]. |
| 9. | «Платформа Unity для разработки в реальном времени,» [В Интернете].  Available: https://unity.com/ru. [Дата обращения: 02 04 2025]. |
| 10. | «Unity Manual: Animation Rigging,» [В Интернете].  Available: https://docs.unity3d.com/Manual/com.unity.animation.rigging.html.  [Дата обращения: 02 04 2025]. |
| 11. | «Unity Manual: Mecanim Animation System,» [В Интернете].  Available: https://docs.unity3d.com/462/Documentation/Manual/  MecanimAnimationSystem.html. [Дата обращения: 02 04 2025]. |
| 12. | «2D Inverse Kinematics (IK),» [В Интернете].  Available: https://docs.unity3d.com/Packages/com.unity.2d.animation  %405.0/manual/2DIK.html. [Дата обращения: 11 04 2025]. |
| 13. | «Newton Raphson Method,» [В Интернете].  Available: https://www.geeksforgeeks.org/newton-raphson-method/.  [Дата обращения: 02 04 2025]. |
| 14. | «В чем достоинство и недостаток метода Ньютона нахождения корней  нели­нейного уравнения» [В Интернете].  Available: https://studfile.net/preview/8843842/page:12/.  [Дата обращения: 11 04 2025]. |
| 15. | «Итерационные методы решения: Алгоритм Ньютона-Рафсона (NR),»  [В Интернете]. Available: https://help.solidworks.com/2010/russian/  SolidWorks/cworks/LegacyHelp/Simulation/AnalysisBackground/  NonlinearAnalysis/Iterative\_Solution\_Methods\_\_Newton-Raphson  \_(NR)\_Scheme.htm. [Дата обращения: 11 04 2025]. |
| 16. | «Метод градиентного спуска,» [В Интернете].  Available: https://www.dmitrymakarov.ru/learning/gradient/.  [Дата обращения: 02 04 2025]. |
| 17. | «Стохастический градиентный спуск,» [В Интернете].  Available: https://neerc.ifmo.ru/wiki/index.php?title=  Стохастический\_градиентный\_спуск. [Дата обращения: 11 04 2025]. |
| 18. | «What is Gradient descent?,» [В Интернете].  Available: https://www.geeksforgeeks.org/what-is-gradient-descent/.  [Дата обращения: 11 04 2025]. |
| 19. | «Матрица Якоби,» [В Интернете].  Available: https://bigenc.ru/c/matritsa-iakobi-7739ee.  [Дата обращения: 02 04 2025]. |
| 20. | «Аппроксимация матрицы Якоби в методах решения жестких задач,»  [В Интернете]. Available: https://www.mathnet.ru/links/  fa232d89ab08fc6082372dca86fae3e8/zvmmf9586.pdf.  [Дата обращения: 11 04 2025]. |
| 21. | «Магия тензорной алгебры: Часть 3 – Криволинейные координаты,»  [В Интернете]. Available: https://habr.com/ru/articles/261717/.  [Дата обращения: 11 04 2025]. |
| 22. | «Псевдообратная матрица,» [В Интернете].  Available: https://vmath.ru/vf5/algebra2/inverse/p\_inverse.  [Дата обращения: 02 04 2025]. |
| 23. | «Псевдообратная матрица,» [В Интернете].  Available: https://ru.wikipedia.org/wiki/Псевдообратная\_матрица.  [Дата обращения: 11 04 2025]. |
| 24. | «Использование псевдообращения в задачах обучения искусственных  нейронных сетей,» [В Интернете].  Available: https://cyberleninka.ru/article/n/ispolzovanie-psevdoobrascheniya  -v-zadachah-obucheniya-iskusstvennyh-neyronnyh-setey.  [Дата обращения: 11 04 2025]. |
| 25. | «Экспоненциальное сглаживание,» [В Интернете].  Available: https://help.fsight.ru/ru/mergedProjects/lib/02\_time\_series\_  analysis/uimodelling\_expsmooth.htm. [Дата обращения: 02 04 2025]. |
| 26. | «Exponential smoothing,» [В Интернете].  Available: https://en.wikipedia.org/wiki/Exponential\_smoothing.  [Дата обращения: 11 04 2025]. |
| 27. | «Методы скользящего среднего и сглаживания: количественное  прогнозирование,» [В Интернете].  Available: https://studfile.net/preview/4179777/page:2/.  [Дата обращения: 11 04 2025]. |
| 28. | «Генетические алгоритмы – математический аппарат,» [В Интернете].  Available: https://loginom.ru/blog/ga-math. [Дата обращения: 02 04 2025]. |
| 29. | «Introduction to Optimization with Genetic Algorithm,» [В Интернете].  Available: https://www.geeksforgeeks.org/introduction-to-optimization-with  -genetic-algorithm/. [Дата обращения: 11 04 2025]. |
| 30. | «Достоинства и недостатки генетических алгоритмов,» [В Интернете].  Available: https://studfile.net/preview/9704700/page:39/.  [Дата обращения: 11 04 2025]. |
| 31. | «Алгоритм роя частиц. Описание и реализации на языках Python и C#,»  [В Интернете]. Available: https://jenyay.net/Programming/ParticleSwarm.  [Дата обращения: 02 04 2025]. |
| 32. | «Инверсная кинематика: простой и быстрый алгоритм,» [В Интернете].  Available: https://habr.com/ru/articles/222689/.  [Дата обращения: 02 04 2025]. |
| 33. | «АВТОМАТИЧЕСКОЕ УПРАВЛЕНИЕ НЕПРЕРЫВНЫМ РОБОТОМ,»  [В Интернете]. Available: https://moitvivt.ru/ru/journal/pdf?id=34.  [Дата обращения: 11 04 2024]. |
| 34. | «A Combined Inverse Kinematics Algorithm Using FABRIK with  Optimization,» [В Интернете].  Available: https://cs.paperswithcode.com/paper/a-combined-inverse-  kinematics-algorithm-using. [Дата обращения: 11 04 2025]. |
| 35. | «Метод наименьших квадратов,» [В Интернете]. Available: http://www.mathprofi.ru/metod\_naimenshih\_kvadratov.html. [Дата обращения: 02 04 2025]. |
| 36. | «Least Squares Method: What It Means, How to Use It, With Examples,»  [В Интернете]. Available: https://www.investopedia.com/terms/l/least-  squares-method.asp. [Дата обращения: 11 04 2025]. |
| 37. | «Формулы для коэффициентов регрессии. Обязательные свойства  линии регрессии. Недостатки метода наименьших квадратов,»  [В Интернете]. Available: https://studfile.net/preview/9268127/page:7/.  [Дата обращения: 11 04 2025]. |
| 38. | «Доступно о кватернионах и их преимуществах,» [В Интернете].  Available: https://habr.com/ru/articles/426863/.  [Дата обращения: 02 04 2025]. |
| 39. | «Quaternions and spatial rotation,» [В Интернете].  Available: https://en.wikipedia.org/wiki/Quaternions\_and\_spatial\_rotation.  [Дата обращения: 11 04 2025]. |
| 40. | «КВАТЕРНИОНЫ,» [В Интернете].  Available: https://sciencejournals.ru/issues/vychmat/2023/vol\_63/iss\_1  /VychMat2301014Veliev/VychMat2301014Veliev-site.html.  [Дата обращения: 11 04 2025]. |
| 41. | «Теория о кватернионах,» [В Интернете].  Available: https://rekovalev.site/quaternion/#quaternion.  [Дата обращения: 18 04 2025]. |
| 42. | «Кватернион,» [В Интернете].  Available: https://ru.wikipedia.org/wiki/Кватернион.  [Дата обращения: 18 04 2025]. |
| 43. | «Складывание рамок,» [В Интернете].  Available: https://ru.wikipedia.org/wiki/Складывание\_рамок.  [Дата обращения: 18 04 2025]. |
| 44. | «Инверсная кинематика в двухмерном пространстве,» [В Интернете].  Available: https://habr.com/ru/articles/358798/.  [Дата обращения: 18 04 2025]. |
| 45. | И. С. Д.С. Колтыгин, «Метод и программа решения прямой  и обратной задачи». |
| 46. | «Human Motion-Inspired Inverse Kinematics Algorithm for a Robotics-Based  Human Upper Body Model,» [В Интернете].  Available: https://digitalcommons.usf.edu/etd/10252/.  [Дата обращения: 27 04 2025]. |
| 47. | «Recursive inverse kinematic analysis for humanoid robot based on depth  camera data,» [В Интернете].  Available: https://link.springer.com/article/10.1007/s12206-021-0640-y.  [Дата обращения: 27 04 2025]. |
| 48. | «Solving Inverse Kinematics of Humanoid Robot Using A Redundant  Tree-shaped Manipulator Model,» [В Интернете].  Available: https://dl.acm.org/doi/10.1145/3448823.3448885.  [Дата обращения: 27 04 2025]. |
| 49. | «Inverse Kinematics for Upper Limb Compound Movement Estimation in  Exoskeleton-Assisted Rehabilitation,» [В Интернете].  Available: https://pmc.ncbi.nlm.nih.gov/articles/PMC4925945/.  [Дата обращения: 27 04 2025]. |
| 50. | «MusculoSkeletal Modeling Using Kinect Data For Telerehabilitation,»  [В Интернете]. Available: https://arxiv.org/abs/1612.08808.  [Дата обращения: 27 04 2025]. |
| 51. | «An adaptive self-organizing fuzzy logic controller in a serious game for  motor impairment rehabilitation,» [В Интернете].  Available: https://arxiv.org/abs/1804.10392. [Дата обращения: 27 04 2025]. |
| 52. | «User-specified inverse kinematics taught in virtual reality reduce time and  effort to hand-guide redundant surgical robots,» [В Интернете].  Available: https://www.nature.com/articles/s44172-025-00357-x.  [Дата обращения: 27 04 2025]. |
| 53. | «Inverse Kinematics for Game Programming,» [В Интернете].  Available: https://medium.com/%40turtle50vp/inverse-kinematics-for-game  -programming-5f9a408e24b2. [Дата обращения: 27 04 2025]. |
| 54. | J. W. S. N. A. Y. C. Y.-T. K. Chong,  «An Application of Augmented Reality (AR) in the Teaching of an Arc  Welding Robot». |

**ПРИЛОЖЕНИЕ 1**

ПРОГРАМНЫЙ МОДУЛЬ ДВИЖЕНИЯ ЧЕЛОВЕКА

ИСХОДНЫЙ ТЕКСТ ПРОГРАММЫ

Москва, 2025

1. **Функция вычисления матрицы Якоби**

Matrix4x4 jacobian = new Matrix4x4();

Vector3 joint1Pos = RightArm.transform.position;

Vector3 joint2Pos = RightForearm.transform.position;

Vector3 joint3Pos = RightHand.transform.position;

Vector3 arm1 = joint2Pos - joint1Pos;

Vector3 arm2 = joint3Pos - joint2Pos;

jacobian.SetRow(0, new Vector4(-arm1.y, -arm2.y, 0, 0));

jacobian.SetRow(1, new Vector4(arm1.x, arm2.x, 0, 0));

jacobian.SetRow(2, new Vector4(0, 0, 1, 0));

return jacobian;

1. **Функция решения задачи инверсной кинематики**

for (int iteration = 0; iteration < MaxIterations; iteration++)

{

// Текущая позиция конечной точки

RightHandCurrent = RightHand.transform.position;

// Ошибка между текущей и целевой позициями

Vector3 delta = RightHandTarget - RightHandCurrent;

// Проверка, достигнута ли точность

if (delta.magnitude < Tolerance)

{

Debug.Log($"Цель достигнута после {iteration} итераций");

break;

}

// Матрица Якоби (вычисляется для текущих углов)

Matrix4x4 jacobian = CalculateJacobian();

// Транспонированная матрица Якоби

Matrix4x4 jacobianT = jacobian.transpose;

// Псевдообратная матрица Якоби

Matrix4x4 jacobianPseudoInverse = PseudoInverse(jacobian, jacobianT, Lambda);

// Изменения углов

Vector3 deltaTheta = MultiplyMatrixVector(jacobianPseudoInverse, delta);

// Применение углов к суставам

ApplyJointRotationsAndPositions();

}

1. **Функция поворота камеры вокруг модели человека**

if (Input.GetMouseButton(1))

{

float horizontal = Input.GetAxis("Mouse X") \* rotationSpeed \* Time.deltaTime;

float vertical = Input.GetAxis("Mouse Y") \* rotationSpeed \* Time.deltaTime;

mainCamera.transform.RotateAround(

pivotPoint,

Vector3.up,

horizontal

);

mainCamera.transform.RotateAround(

pivotPoint,

mainCamera.transform.right,

-vertical

);

mainCamera.transform.LookAt(pivotPoint);

}

1. **Функция ограничения движения суставов по плоскостям**

switch (activeAxis)

{

case MoveAxis.XY:

return new Vector3(newPosition.x, newPosition.y, originalPosition.z);

case MoveAxis.XZ:

return new Vector3(newPosition.x, originalPosition.y, newPosition.z);

case MoveAxis.ZY:

return new Vector3(originalPosition.x, newPosition.y, newPosition.z);

default:

return newPosition;

}