

Федеральное государственное автономное образовательное учреждение  
высшего профессионального образования  
«Уральский федеральный университет имени первого  
Президента России Б.Н. Ельцина»  
Институт фундаментального образования  
Кафедра интеллектуальных информационных технологий

На правах рукописи

**Гайниязов Игорь Мадыхатович**

**Разработка алгоритмического и программного обеспечения  
для обработки сигналов программно-аппаратного комплекса  
измерения и сопоставления движений**

2.3.1 — Системный анализ, управление и обработка информации, статистика

## **ДИССЕРТАЦИЯ**

на соискание ученой степени  
кандидата технических наук

Научный руководитель :  
кандидат технических наук, доцент  
Обабков Илья Николаевич

Екатеринбург — 2023

# Оглавление

Основные обозначения и соглашения . . . . .	4
Введение . . . . .	6
1 Анализ состояния предметной области. Постановка задач исследования . . . .	12
1.1 Актуальность исследования комплексов контроля движений кисти человека	12
1.2 Методики контроля движений . . . . .	13
1.3 Проблемы применения инерционных датчиков и методик анализа их вы- ходных сигналов . . . . .	27
1.4 Постановка задач исследования . . . . .	35
2 Анализ особенностей информационных сигналов в комплексе контроля движе- ний кисти человека . . . . .	37
2.1 Анализ основных характеристик информационных сигнала . . . . .	38
2.2 Разработка алгоритмического обеспечения для анализа сигналов . . . . .	55
2.3 Выводы по главе . . . . .	76
3 Методики уменьшения размерности первичной информации, собираемых с инер- ционных датчиков . . . . .	77
3.1 Формирование набора данных о движении кисти . . . . .	77
3.2 Методика трансформации данных о движении с использованием вейвлет"-преобразования . . . . .	80
3.3 Методика снижения количества столбцов в матрице с данными о движении	95
3.4 Выводы по главе . . . . .	98
4 Разработка алгоритмической и программной части системы контроля движе- ний кисти человека . . . . .	100
4.1 Учет алгоритмических особенностей в аппаратной части комплекса . . .	101
4.2 Разработка программного обеспечения комплекса . . . . .	102
4.3 Анализ результатов использования программно - аппаратного комплекса	105
4.4 Выводы по главе . . . . .	117
Заключение . . . . .	119

Список литературы . . . . .	122
Приложение А. Свидетельство о регистрации программы для ЭВМ . . . . .	133
Приложение Б. Свидетельство о регистрации программы для ЭВМ . . . . .	134
Приложение В. Свидетельство о регистрации программы для ЭВМ . . . . .	135
Приложение Г. Свидетельство о регистрации программы для ЭВМ . . . . .	136
Приложение Д. Свидетельство о регистрации программы для ЭВМ . . . . .	137
Приложение Е. Листинг программы микроконтроллера . . . . .	138
Приложение Ж. Листинг подпрограммы приема данных с МК . . . . .	140
Приложение З. Листинг программы обработки и визуализации данных . . . . .	142
Приложение И. Листинг подпрограммы снижения избыточности данных . . . . .	152
Приложение К. Листинг подпрограммы выделения паттерна движения . . . . .	155
Приложение Л. Листинг подпрограммы формирования общего паттерна движения	156
Приложение М. Листинг подпрограммы отбора признаков . . . . .	158
Приложение Р. Акт использования программного обеспечения в 2022 году . . . . .	159
Приложение Н. Аппаратная реализация программно-аппаратного комплекса для измерения и сопоставления движений кисти человека . . . . .	160

## Основные обозначения и соглашения

- AR — Augmented Reality (англ.) – расширенная реальность. Набор технологий, при котором виртуальные объекты отображаются в реальном окружении, видимом пользователями.
- CS — Chip Select (англ.) – выбор микросхемы. Название линии управления в цифровой электронике, используемой для выбора микросхемы. В данной работе выбираются МЭМС с помощью МК.
- CPU — Central Processing Unit (англ.) – центральное обрабатывающее устройство. Интегральная схема, исполняющая машинные инструкции (код программ), главная часть аппаратного обеспечения компьютера.
- DoF — Degrees Of Freedom (англ.) – число степеней свободы. Указывает на возможность совершать геометрические повороты вокруг одного из трех взаимно перпендикулярных направлений (рыскание, тангаж, крен).
- DTW — Dynamic Time Warping (англ.) – алгоритм динамического искажения временной шкалы.
- SPS — Samples Per Second (англ.) – число выборок в секунду.
- I2C — Inter-Integrated Circuit (англ.) – последовательная асинхронная шина данных с 2-х проводным интерфейсом.
- MoCap — Motion Capture (англ.) – захват или оцифровка движений.
- SPI — Serial Peripheral Interface (англ.) – последовательная синхронная шина данных с 4-х проводным интерфейсом.
- VR — Virtual Reality (англ.) – виртуальная реальность. Набор технологий взаимодействия с виртуальными объектами.

- ВП — вейвлет-преобразование (или всплеск) – интегральное преобразование, которое представляет собой свертку вейвлет-функции с сигналом. В данной работе применяется исключительно дискретное вейвлет-преобразование, для которого вейвлеты дискретизируются.
- МК — Микроконтроллер – микросхема, предназначенная для управления электронными устройствами.
- МЭМС — Микроэлектромеханические датчики – в данном случае будет подразумеваться только инерциальный тип.
- НД — Набор данных – коллекция из начальных или обработанных записей МЭМС, хранящихся в формате CSV с применением разделителя - запятой «,».
- ОС — Обратная Связь – информирование человека об обработанных данных, поступивших из системы захвата движения.
- ПЛИС — Программируемая Логическая Интегральная Схема – электронный компонент, используемый для создания цифровых интегральных схем.
- Цифровой паттерн движения — Структурированная последовательность данных, полученных с инерциальных микроэлектромеханических сенсоров, включающих акселерометр и гироскоп. Эти данные описывают изменения линейного ускорения и угловой скорости во времени. Путем объединения этих рядов в матрицу размерности  $M \times N$ , где  $M$  представляет собой количество временных отсчетов, а  $N$  - количество выходных каналов, создается цифровой паттерн, который содержит информацию о движении объекта в пространстве.

# Введение

## Практическая постановка задачи

В современных образовательных и коммерческих организациях (спортивные клубы, медицинские центры и технологические компании) существует значительный интерес к обучению двигательным навыкам, телемедицинскому мониторингу и эффекту погружения в виртуальную реальность.

Например, при обучении бегу, игре на фортепиано или физической реабилитации пациентов возникает потребность в использовании эталонной техники, которая определяется определенным набором правильных действий. Неправильное выполнение этих действий может замедлить достижение желаемого результата. В данной диссертационной работе внимание уделяется сопоставлению цифровых паттернов движений, полученных с инерциальных микроэлектромеханических датчиков, с предварительно известными шаблонами движения, которые являются эталонами.

Для оцифровки всех действий, связанных с обучением двигательным навыкам, нужно разработать программно-аппаратный комплекс. Одним из основных требований к этому комплексу является высокая производительность, позволяющая эффективно сопоставлять цифровые паттерны движений. Важно разработать компактный формат хранения информации о движении, который бы сохранял качество этой информации.

Однако достижение автономности такого комплекса представляет свои сложности. Это требует пересмотра методов обработки данных о движении. Существует способы оптимизации работы комплексов, но не все из них учитывают избыточность описания движений. Кроме того, некоторые алгоритмы накапливают интегральную ошибку.

## Цели, актуальность и новизна работы

### Актуальность темы исследований и степень ее разработанности

В организациях, занимающихся обучением двигательным навыкам, часто возникает проблема контроля за техникой выполнения необходимых движений. В этом случае использование видеокамер и микрофонов может оказаться недостаточным. Однако существуют практические применения, такие как удаленный мониторинг цифровых паттернов движения пациентов в телемедицине, например, для отслеживания прогресса лечения при нарушениях двигательных функций человека.

Общей задачей в таких сферах является научение пользователей выполнять определенные упражнения схожим образом с эталонными цифровыми паттернами движения с допустимыми вариациями исполнения.

В сфере VR-реабилитации нашли применение платформы для реабилитации после инсульта, основанной на контроле движений kinect (А. Kashheev, G. Tokunov). В данной диссертационной работе рассматривается контроль мелкой моторики, в отличие от работы «Интеллект и инновации». Исследования по выбранному направлению внесли вклад в биомеханику и реабилитацию, позволив ввести термин биомеханической избыточности в описании данных о движении. В данной работе этот термин играет важную роль, поскольку одна часть работы посвящена снижению избыточности с помощью корреляционного анализа, а другая часть - правильному сжатию данных без потерь. Среди институтов стоит выделить Сеченовский Университет (за разработку датчиков определения движения людей L. Ichkitidze, V. Zaborova, A. Tsymbal) и Московскую государственную академию физической культуры (А. Tambovskij)

Национальный исследовательский университет «МИЭТ» внес свой вклад в исследование погрешностей МЭМС с помощью анализа по вариации Аллана (А. Arhipov, А. Timoshenkov). Результаты исследования помогли оценить датчики BMI160, применяемые в работе. Зарубежные компании Xsens (D. Roetenberg, H. Luinge, C. Baten) и Perception NEURON (S. Ryan, F.

Steph) являются аналогами систем захвата движений на МЭМС, которые применяются для исследований движений в спорте. Благодаря работам, выполненным на базе этих систем, удалось собрать статистику возможных архитектур систем контроля движений и вариантов их применения.

## **Цель работы**

Цель диссертационного исследования заключается в разработке и применении алгоритмического обеспечения в программно-аппаратном комплексе для обучения технике движения кисти человека с использованием методики сравнения исследуемого движения с эталонным.

Для достижения поставленной цели сформулированы и решены следующие **задачи**:

- Анализ состояния предметной области.
- Анализ особенностей информационных сигналов программно-аппаратного комплекса контроля движений.
- Разработка алгоритмического обеспечения программно-аппаратного комплекса контроля движений.
- Проведение экспериментальной апробации программно-аппаратного комплекса контроля движений.

**Объектом исследования** являются методы визуализации, трансформации, обработки и анализа информации в компьютерных системах контроля движений.

**Предметом исследования** являются алгоритмы сравнения сигналов, определяющие степень схожести движений, и алгоритмы снижения размерности хранящихся исходных сигналов движений.

## **Методология и методы исследования**

Результаты, представленные в диссертационной работе, получены с применением методов системного анализа, вариации Аллана, корреляционного анализа, вейвлет-преобразования и алгоритма искажения временной шкалы.



## **Научная новизна**

1. Разработана методика хранения информации о движении кисти человека на основе анализа информационных сигналов с МЭМС-датчиков, позволяющая учитывать дрейф нулевого уровня выходных каналов датчиков единожды.

**Соответствует 5 пункту паспорта специальности.**

2. Обоснована возможность уменьшения размерности матрицы с данными о движении кисти человека от 24 до 48 раз без потери значимой информации о самом движении, основанная на вейвлет-преобразовании и корреляционном анализе.

**Соответствует 12 пункту паспорта специальности.**

3. Разработана научно-обоснованная методика для сравнения цифровых паттернов движений, обеспечивающая увеличение быстродействия программно-аппаратного комплекса от 14 до 17 раз по сравнению с аналогичными решениями без сжатия матрицы с данными о движении.

**Соответствует 14 пункту паспорта специальности.**

## **Теоретическая значимость работы**

Теоретический вклад данной работы представлен в виде новой методологии, которая успешно решает проблему дрейфа нулевого уровня на выходных каналах сенсоров, а также способна уменьшить размерность входных данных, сохраняя при этом критически важную информацию о движении при сопоставлении паттернов движений. Учёт дрейфа нулевого уровня выполняется однократно на этапе предобработки данных.

## **Практическая значимость работы**

Практическая ценность исследования выражается в разработанном программно-аппаратном комплексе для мониторинга движений. Комплекс поддерживает анализ биомеханической избыточности и сопоставление движений. Этот подход применим в дистанционном обучении, телемедицине и виртуальной реальности, снижая вычислительную нагрузку.

## **Положения, выносимые на защиту**

1. Предложенная методика хранения информации о движении кисти человека, основанная на анализе сигналов с МЭМС-датчиков, позволяет единожды учитывать дрейф нулевого уровня выходных каналов датчиков в задаче сопоставления цифровых паттернов движения.
2. Предложенная методика уменьшения матрицы с данными о движении обеспечивает снижение исходного размера от 24 до 48 раз без потери значимой информации о движении.
3. Предложенная методика сравнения цифровых паттернов движений обеспечивает увеличение быстродействия разработанного программно-аппаратного комплекса от 14 до 17 раз по сравнению с аналогичными решениями без сжатия матрицы с данными о движении.

## **Апробация работы.**

Результаты диссертации ранее были представлены и обсуждены на различных научных конференциях и семинарах (в том числе международных):

- на 27 международной конференции по машинному зрению, системам виртуального окружения «GraphiCon» (Пермь, 2017 г.);
- на IV международной конференции «Физика. Технологии. Инновации ФТИ–2017» (Екатеринбург, 2017 г.);
- на семинарах сектора компьютерной визуализации и лаборатории анализа сложных систем в Институте Математики и Механики УрО РАН (Екатеринбург, 2017, 2019 – 2021 гг.);
- на научно–практической конференции «Молодые учёные и научно–технический прогресс России» (Екатеринбург, 2018 г.);
- на всероссийском конкурсе научно–технологических проектов «Сириус» (Сочи, 2019 г.);
- на 29 международной конференции по машинному зрению, системам виртуального окружения «GraphiCon» (Брянск, 2019 г.);
- международный форум по когнитивным нейронаукам на «Cognitive

neuroscience — 2019» (Екатеринбург, 2019 г.);

- на международной конференции по инженерным, компьютерным и информационным наукам «Сибиркон» (Екатеринбург, 2019 г.);
- на всероссийской конференции по машинному зрению «Наука и глобальные вызовы XXI» (Пермь, 2020 г.);
- на международной конференции 2020 Global Smart Industry Conference «GloSIC» (Челябинск, 2020 г.);

**Публикации.** По результатам диссертации лично автором и в соавторстве опубликованы 13 работ, в том числе 7 статей опубликованы в рецензируемых научных журналах, из них 5 статей в изданиях, входящих в международные базы цитирования Scopus и Web Of Science и 5 свидетельств о регистрации программ ЭВМ:

- 7 работ в российских научных журналах, рекомендованных ВАК [1–7];
- 5 работ в журналах и материалах международных конференций [9–13];
- 5 свидетельств о регистрации программ для ЭВМ [14–18].

## **Структура и объем диссертации.**

Диссертация состоит из списка основных обозначений и соглашений, введения, четырех глав, заключения, списков литературы из 128 наименования, приложения и публикаций автора по теме диссертации. Кроме того, работа содержит 56 иллюстрации, 20 таблиц и 3 алгоритмические вставки на псевдокоде.

## **Благодарности**

Автор выражает глубокую благодарность научному руководителю Илье Николаевичу Обабкову, коллективу лаборатории анализа сложных систем и сектору компьютерной визуализации Института математики и механики им. Н.Н. Красовского УрО РАН (в особенности Владимиру Лазаревичу Авербуху и Илье Сергеевичу Стародубцеву) за ценные советы и поддержку.

# Глава 1

## Анализ состояния предметной области. Постановка задач исследования

### 1.1 Актуальность исследования комплексов контроля движений кисти человека

Традиционно, в научных работах стремятся к полному представлению движений с возможностью 3D визуализации. Однако в данном исследовании задача ставится иначе и предполагает более простой подход: сравнение текущего движения, зарегистрированного датчиками, с известным заранее шаблоном (цифровым паттерном) движения. При этом, желательно иметь наименьший объем данных, так как все цифровые паттерны известны заранее, и задача состоит в том, чтобы оперативно предоставить обратную связь о качестве исполнения движения.

Если сравнивать начальные цифровые паттерны движений целиком, то это превращается в крайне сложную задачу из-за избыточности параметров. Для достижения результата необходимо вычислить расстояние между цифровыми паттернами движения со сложностью алгоритма  $O(n^2)$ .

Рассмотрим матрицы размерности  $M_1 \times N$  и  $M_2 \times N$ , где  $M_1$  и  $M_2$  представляют собой количество временных точек ученика и тренера соответственно, а  $N$  - количество выходных каналов датчиков. Задача заключается в определении степени близости движений тренера и ученика. Это является более конкретной и упрощенной задачей по сравнению с создани-

ем общих моделей движения. В рамках данной работы рассматриваются методики снижения размерности цифрового паттерна движения с сохранением информации о самом движении.

В итоге можно выделить несколько областей применения комплексов контроля движений: анализ движений, реабилитация, обучение и естественные интерфейсы.

## 1.2 Методики контроля движений

### Развитие методов контроля движений

Традиционно выделяют следующие контексты применения контроля движений:

- **Анализ движений.** Исследования по выявлению цифровых паттернов в движениях [19]. Применение подобных комплексов дает важную дополнительную информацию об эффективности восстановительного лечения [20]. Выделение базовых элементов в движении [21] позволяет понять структуру того или иного движения.
- **Реабилитация, в том числе домашняя.** Определение патологических движений [22,23] с оповещением лечащего врача. Мониторинг нарушения двигательной активности при болезни Паркинсона [24]. Комплекс удаленного мониторинга для обнаружения патологических движений [25–29]. В данном случае уже известны параметры нормального, субоптимального и аномального движений; задачей становится своевременное информирование о появлении патологического движения. Существует своя восстановительная методика восстановления движений кисти человека и при домашней реабилитации [30].
- **Обучение.** Проверка техники выполнения физической тренировки [31–33]. Подобные решения для мониторинга тренировок используют и в спорте. Например, в боксе [34], спортивной стрельбе [35] или подготовке космонавтов [36,37].
- **Естественные интерфейсы, в том числе VR.** Подобные комплек-

сы востребованы как естественные интерфейсы человек – компьютер [38]. Например, распознавание жестового языка [39] становится более естественным при применении МЭМС датчиков, так как нет необходимости наводить камеру телефона для распознавания жестов. Не осталась без внимания и сфера исследований эффекта присутствия в VR [40].

Более подробно решаемые задачи и цели можно увидеть в следующей таблице:

Таблица 1.1 – Задачи для контроля движений.

Область задачи	Решаемая задача	Сопутствующие задачи	Цель
Анализ техники движений спортсмена	Бокс <i>(И.А. Хасанишин, О.В. Осипов)</i>	Построение 3д модели движения контролируемого объекта	Контроль техники удара боксера. <b>Тренер:</b> правильный удар. <b>Ученик:</b> текущее выполнение удара
	Теннис <i>(G.Delgado-Garcia)</i>		Обучение правильной техники удара по мячу. <b>Тренер:</b> правильный удар с ракеткой. <b>Ученик:</b> текущее выполнение удара ракеткой
	Бег <i>(J. Vanrenterghem, Y. Matsuda)</i>		Контроль техники бега. <b>Тренер:</b> правильная техника бега. <b>Ученик:</b> текущее выполнение бега
Мониторинг процесса реабилитации	Восстановление двигательной функции (после перелома или инсульта). <i>R. Felius, M. Geerars, E. Hansson, A.M. Формальский</i>		Мониторинг — процесс восстановления. <b>Тренер:</b> движения, свойственные норме движения. <b>Ученик:</b> текущие движения пациента
	Отслеживание падений <i>F.S. Ayachi, H.P. Nguyen, M. Kos, I. Kramberger</i>		Мониторинг двигательных функций пожилых людей. <b>Тренер:</b> пример падения. <b>Ученик:</b> текущее движение
	Анализ двигательной активности младенцев. Метод Прехтла. <i>C.B. Redd, M. Karunanithi, S. Reich, D. Zhang</i>		Мониторинг движения двигательной активности младенцев. <b>Тренер:</b> движения, свойственные ребенку с нормальной активностью. <b>Ученик:</b> текущие движения ребенка
Управление системами виртуальной реальности	Человеко-машинный интерфейс (отслеживание пальцев рук, целевых действий). <i>R. Sohail, S. Neyret, A.I. Rivas</i>	Управление. Человеко-машинный интерфейс. <b>Тренер:</b> управляющие движения (перемещение указателя, выделение, клик). <b>Ученик:</b> текущее движение пользователя	

Историческое развитие методов представлено далее в главах.

### Анатомический анализ походки человека

Биомеханика имеет глубокие корни: еще Аристотель заложил основы науки о движении человека. Впоследствии, с развитием классической механики, ученые все детальнее описывали тело людей через опорно-двигательный аппарат. Например, немецкие ученые Эдуард и Вильгельм Верберы еще в XIX веке привели наиболее полные для своего времени данные о кинематических характеристиках ходьбы человека [41]. Однако вследствие

несовершенства измерительных приборов данные имели значительное зашумление.

### **Пневмография и хронофотографический аппарат**

Данную проблему решал физиолог Этьен-Жюль Маре, который разработал комплекс пневмографии. Этот способ позволял записывать опорные реакции и распределение массы объекта в динамике, но вопрос о более полном анализе движений был решен еще недостаточно полно. Параллельно велись прикладные исследования: в 1839 г., после изобретения фотографии, Маре разработал комплекс последовательных фотографий, которая позволяла фиксировать снимки с частотой дискретизации порядка 19 Гц, а в 1882 г. изобрел «хронофотографический аппарат», который коротко называли фоторужьем Маре.

### **Циклография, кимоциклография и видеомониторинг за движениями**

Дальнейшее развитие методики дали Ф.Б. Гильберт [42] и Н.А. Бернштейн [43]. Полученная методика — один из первых методов, позволяющий с достаточной точностью анализировать движения, — назвали циклографией. Основным его приложением была оптимизация движений на производстве. Это снижало нагрузку на самих рабочих и повышало качество продукции — вместе с увеличением производительности. Кроме того, уже в то время методику применяли для изучения движения водителей, музыкантов и т. д. [44]. Параллельно неврологи в процессе лечения пациентов начали рассматривать патологические проявления, относящиеся к движениям, а Сеченов так же анализировал движения рабочих на производстве [45].

С появлением первой видеокамеры методика получила дальнейшее развитие. Новая методика названа кимоциклографией. В дальнейшем он развился в комплексы отслеживания и контроля движения видеокамерами с использованием световых маркеров, что на сегодня чрезвычайно востребовано в области анимации. Главное отличие этой методики от предшествующих состоит в применении информационных технологий.

## **Инерциальные МЭМС**

С развитием технологий в XX веке, а именно миниатюризацией устройств, появляется направление микроэлектромеханических датчиков, или МЭМС. Активное развитие МЭМС-технологий началось с 90-х годов прошлого века. МЭМС-методика для анализа движений начал применяться сравнительно недавно, начиная с 2000-х годов.

## **Классификация технологий контроля движений**

Примеры комплексов контроля движения можно найти как в отечественной, так и в зарубежной литературе. Далее представлены известные из современной практики технологии, которые могут быть использованы для реализации комплекса оцифровки движений человека.

Типы существующих технологий контроля движений:

- Маркерная технология.
  - Оптическая.
    - \* Пассивная.
    - \* Активная.
  - Магнитная.
  - Механическая.
    - \* Тензорезистивная.
    - \* Экзоскелетная.
  - МЭМС.
  - Гониометрическая.
  - Акустическая.
  - Актографическая.
- Безмаркерная технология.

Также нужно учитывать возможность совместного использования разных типов комплексов контроля движений [46].

Методика оптически пассивного контроля движений использует светоотражающий материал в качестве маркеров, значения которых фиксируются камерами, расположенными в рабочей зоне. Этот подход позволяет



цифровать как мимическую активность, так и движения тела. Несмотря на требование большого количества маркеров, он отличается надежностью и простотой организации. Однако есть и недостатки: возможность неправильной идентификации маркеров и необходимость значительного времени для их размещения на человеке. Оптически активная методика характеризуется самостоятельным излучением света маркерами, что позволяет их адресовать, но увеличивает сложность системы и требует наличия источника питания.

Магнитная технология включает использование магнитов в качестве маркеров, при этом комплекс регистрирует изменения магнитного поля и преобразует исходные значения датчиков в углы ориентации в пространстве. Однако, требуется вычисление положения приемных катушек и возникают сложности с экранированием от магнитных и электрических помех. Также проблемой является ограниченная рабочая зона даже в специально подготовленном помещении и переменная чувствительность датчиков в зависимости от положения в рабочей зоне.

Тензорезистивная технология использует датчики, состоящие из материала, сопротивление которого изменяется в зависимости от степени изгиба. Такие датчики имеют по 1 DoF [47], и для сустава с 2 DoF требуются два таких датчика. Они подвержены быстрому механическому износу, что ведет к потере исходных свойств и требует частой калибровки. Несмотря на это, этот подход эффективен для контроля простых движений благодаря простоте изготовления датчиков и низкой вычислительной нагрузке.

Экзоскелетная технология предполагает использование внешнего экзоскелета для выполнения движений человеком. Поворотные механизмы экзоскелета обеспечивают точное измерение угла сгиба и выдают аналоговые значения, что после калибровки обеспечивает получение сигнала с высокой точностью. Однако к недостаткам этого подхода относятся массивность конструкции и механический износ элементов комплекса.

Стоит отметить, что выбор конкретной технологии обычно определяется спецификой исследуемых движений и требованиями к точности их

фиксации. Каждая из представленных методик имеет свои преимущества и недостатки, и их использование должно основываться на целях конкретного исследования или применения.

Технология, основанная на использовании МЭМС-датчиков, применяет миниатюрные инерциальные датчики для цифровой обработки поворотных характеристик во времени. В большинстве случаев, инерциальные МЭМС-датчики требуют преобразования исходных данных в угловые значения поворота в динамике [48]. Применительно к человеческому телу используется суставно-поворотный подход для описания [49]. В качестве примера, можно привести упрощенное биомеханическое представление руки человека, где механизм состоит из 18 шарниров и 27 DoF звеньев. При нормальной функциональности используются только они, в то время как остальные вызывают болезненные ощущения [50]. Важно отметить, что данные могут быть обработаны иначе, в отличие от скелетного подхода, когда для каждого датчика рассчитывается его ориентация в пространстве.

Гониометрическая технология использует медицинский прибор гониометр, который функционирует аналогично специализированному транспортеру. Эти комплексы применяются в основном в медицине для анализа движений отдельных суставов [51].

В акустической технологии источник звука располагается на самом человеке, а приемники, расположенные в оборудованном помещении, принимают сигнал. С помощью триангуляции вычисляется координата источника. Эти комплексы чувствительны к внешнему звуку [52].

Актографическая технология обычно основана на использовании пневматических датчиков и применяется в контексте изучения двигательной активности во время сна или для пациентов, находящихся на постельном режиме. Для актографии используется специальный прибор — актограф, который состоит из двух частей: воспринимающей, реагирующей на движения объекта, и регистрирующей, фиксирующей полученные результаты. Комплекс обычно крепится либо под матрас, либо на самого человека — в зависимости от цели исследования (дневных или ночных движений со-

ответственно). В сравнении с другими методиками анализа, актография позволяет оценивать обобщенные движения без учета мелкой моторики.

Безмаркерные технологии используют машинное зрение для распознавания необходимых движений в видео. Наиболее часто используются инфракрасный приемо-излучатель или обычная RGB-камера.

## **Сравнение методов мониторинга за движениями**

Сравним между собой разные комплексы для контроля движений по нижеприведенным критериям.

- **Стационарность или мобильность.** Данный критерий раскрывает, в большей степени технология адаптирована для разворачивания в определенном помещении или нет. Например, оцифровка движений с помощью видеокамер возможна как в случае подготовки помещения для стационарного использования, так и в мобильном исполнении (Oculus Quest 2 с оцифровкой кисти руки).
- **Объем передаваемых данных для одного датчика.** Под одним датчиком подразумевается одно устройство первичных данных: например, одна видеокамера или один МЭМС.
- **Минимальное количество датчиков для захвата движения руки.** Для качественного захвата различные технологии требуют разного числа датчиков. Под минимальным числом мы будем иметь в виду количество датчиков, нивелирующее недостатки конкретной методики.
- **Максимальное количество датчиков для захвата движения руки.** Под максимальным числом датчиков мы будем иметь в виду наибольшее количество датчиков, встреченное в литературе (см. список использованной литературы).

Таблица 1.2 – Сравнение методов контроля движений.

Название методики	Стационарность (в подготовленном помещении) или мобильность (автономная система)	Размерность матрицы одного движения	Необходимая частота перекалибровки (временные интервалы калибровки системы)
Оптический <i>A. Kirk, B.B. Борзиков, H.H. Ружина</i>	Все варианты	3036 x 3036	Редкая
<b>Инерциальный</b> <i>A.B. Алпатов, M.C. Ашапкина, E.Kruk</i>	<b>Мобильность</b>	<b>96 x 200</b>	<b>Редкая</b>
Тензорезистивный <i>A.Roda, J.L.Sancho, M.Kwan, Д.В.Шадрин</i>	Мобильность	132 x 200	Частая
Экзоскелетный <i>X.Gu, Y.Zhang, W.K.Lee, E.A.Дудоров</i>	Стационарность	96 x 200	Средняя
Магнитный <i>J.F. O'Brien, S.Yabukami</i>	Стационарность	96 x 200	Средняя
Акустический <i>E.Volná, M.Kotyřba</i>	Стационарность	96 x 200	Средняя

Для анализа выберем методики, наиболее подходящие для отслеживания движений кисти руки в условиях дома или на улице. Исключим из рассмотрения актографические и гониометрические технологии, поскольку их применение обычно ограничено медицинскими сферами.

Сгруппируем активные и пассивные оптические, а также безмаркерные технологии в одну категорию — видеооцифровка. Несмотря на различия в используемых диапазонах длин волн электромагнитного излучения, эти методы объединяет наличие устройства для приема видеоданных, сходный объем первичных данных, получаемых от устройства, и необходимость решения проблемы перекрытия конечностей. Тот факт, что необходимые зоны для контроля определяются по-разному (в маркерном методе требуется установка физических маркеров, в безмаркерном — определение нужных точек алгоритмически) является менее значимым.

Для дальнейшего более глубокого сравнения были выбраны способы ви-

деооцифровки и МЭМС как самые подходящие для поставленной задачи. Они позволяют оцифровывать движения в мобильном исполнении и количество датчиков для этого минимально.

### **Применимость к заданному контексту**

При проведении сравнительного анализа методик, общие результаты отображаются аналогичными [53–56]. Процедура сравнения включает последовательные этапы. Изначально формируется перечень действий для записи. Затем испытуемый оснащается обоими комплексами одновременно. Это не вносит дополнительных искажений в работу обеих систем, поскольку они функционируют на различных принципах. Единственное потенциальное исключение — МЭМС-магнитометр, который по своей природе чувствителен к внешним воздействиям при изменении локального магнитного поля.

Для достоверного видеозахвата движения требуется не одна камера, а несколько [57]. Для каждого конкретного помещения необходимо учитывать уникальное расположение камер. В условиях низкой освещенности качество видеозахвата движения ухудшается. Указанные обстоятельства делают внедрение подобного решения на основе видеокамер трудоемким и ресурсозатратным заданием, а в некоторых условиях — невозможным. В то же время, к комплексу на основе МЭМС-датчиков предъявляются только два постоянных требования: производительность комплекса и время автономной работы от аккумулятора. Отметим, что при сравнении требований к производительности аналогичных комплексов на основе МЭМС-датчиков и видеозахвата, требования к МЭМС будут ниже, так как обрабатывается меньший объем входящих данных при схожей информативности.

В контексте рассмотрения всех аналогов целесообразно обратить внимание на производительность решений. Технологии, использующие различные видеокамеры, обладают высоким разрешением. Однако, если требуется объемные данные объекта с разных сторон, необходимо применение множества таких камер [53, 58]. На входе мы имеем не информативные части данных, которые требуют сегментации. Это влечет за собой увеличение

времени на анализ данных.

### Тепловые карты расположения маркеров

Прежде всего стоит рассмотреть, в каких местах чаще всего располагают датчики в обоих случаях. Для создания тепловой карты расположения МЭМС (рис.1.1 была проанализирована соответствующая литература [19, 22, 59–62].

Карта показывает несколько наиболее часто встречающихся зон крепления. Это 14 точек, позволяющих собрать минимальное биомеханическое представление движения человека. Однако это достигается за счет отсутствия оцифровки пальцев рук, что может быть критичным для случаев, когда необходимо отслеживать мелкую моторику.

Для аналогичных задач видеоотслеживания была создана соответствующая тепловая карта (см. рис.1.2. Для ее составления потребовались данные по использованию OpenNI, Kinect v1 SDK, Kinect v2 SDK и комплексы контроля движений различных исследовательских групп [24, 53, 63, 64]).

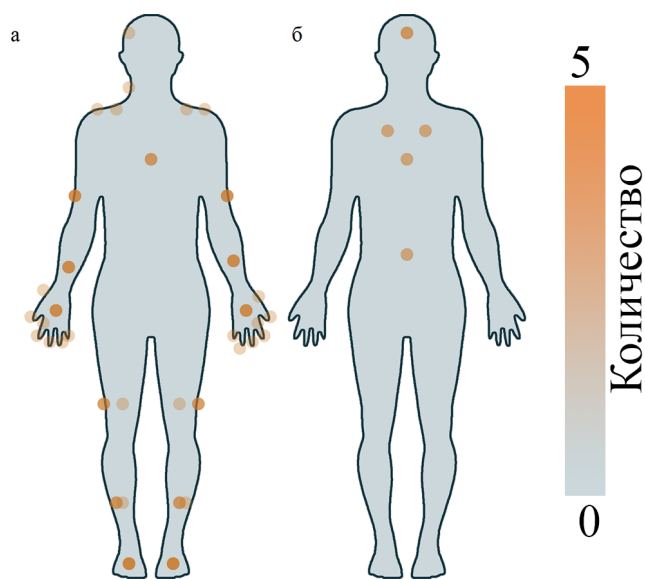


Рисунок 1.1 – Тепловая карта расположения МЭМС-датчиков: а) человек стоит лицом; б) человек стоит спиной.

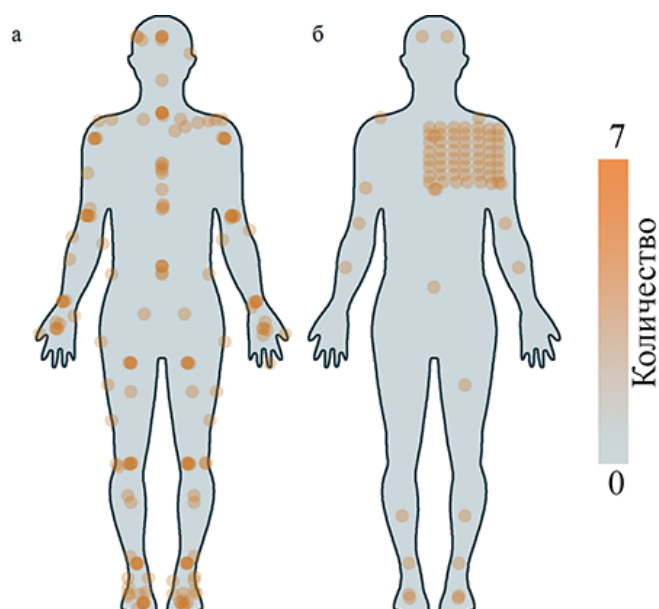


Рисунок 1.2 – Тепловая карта расположения оптических датчиков. (а) Человек стоит лицом. (б) Человек стоит спиной.

Сопоставление двух тепловых карт позволяет увидеть, что вариативность расположения датчиков у МЭМС-комплексов для контроля движений ниже, чем у технологий видеотслеживания. Это может свидетельствовать о более устоявшемся наборе задач и методах их решения. На рис. 1.2 б) массив оптических меток расположен на плече для исследования движений плеча во время игры в теннис.

При сопоставлении аналогичных моментов, наблюдается, что в некоторых случаях авторы реализовали отслеживание движений кисти человека и отдельных пальцев. Можно предположить, что полноценный мониторинг кисти руки может быть не столь необходимым, когда речь идет о более сложных движениях (локомоциях), которые характерны для всего тела.

### **Аналоги для контроля движения кисти человека на МЭМС**

Был осуществлен отдельный анализ перчаток, способных отслеживать движения. В основном, их применение распределено между распознаванием движений руки для языка жестов и игровых контроллеров. С целью определения типичного расположения датчиков, была составлена тепловая карта по специфической выборке (см. рис.1.3) [65–73].

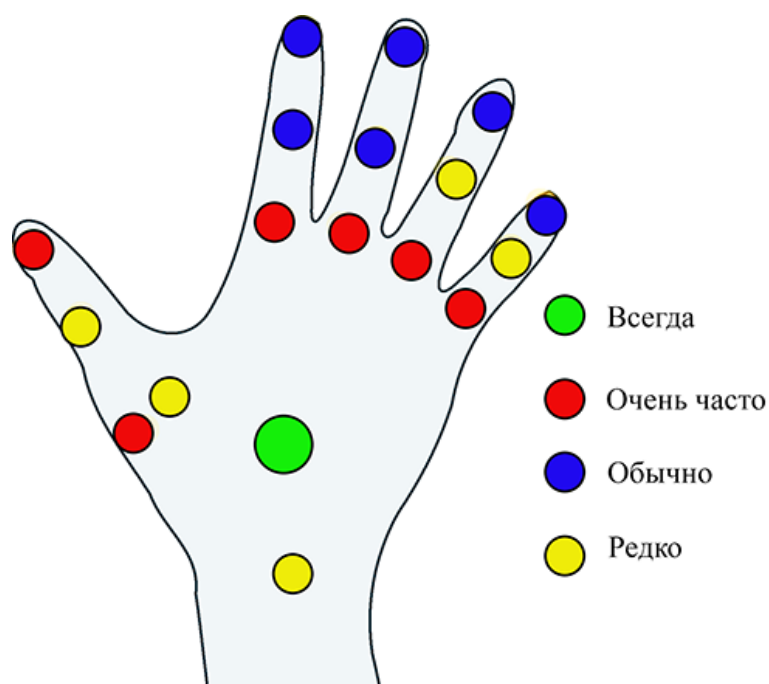


Рисунок 1.3 – Частотная карта расположения МЭМС-датчиков.

Анализ данной тепловой карты демонстрирует наличие основных областей из 6 точек, которые уже обеспечивают достаточность для создания контроллера ввода информации для компьютера. В задачах реабилитации при мониторинге мелкой моторики, это уже приводит к хорошим результатам [74, 75], однако данное направление всё ещё обладает значительным потенциалом для развития. В рамках данной работы будет рассмотрен подход к определению минимального числа степеней свободы (DoF) для МЭМС-комплекса контроля движений на примере перчатки с 16 датчиками — по 6 DoF на каждый. В указанных источниках вопрос разработки рассматривался лишь в контексте применения к конкретной задаче. В нашем исследовании мы сосредоточимся на универсальном варианте разработки, используя биомеханический подход. Перчаточный интерфейс будет рассматриваться как наиболее примечательный пример применения данного подхода, поскольку, с одной стороны, он требует миниатюрности решения, а с другой — способствует минимизации обработки данных за счет использования функциональных связей между датчиками.



## Конфигурации комплексов контроля движений на МЭМС

В рамках разработки комплекса можно выделить пять ключевых блоков. В контексте данного исследования не все из них представляют собой новаторский подход, однако все были реализованы в ходе работы над проектом. Остальные результаты в основном повторяют усилия, уже предпринятые другими исследователями. Для иллюстрации основного фокуса настоящего исследования обратимся к схеме, представленной на рис. 1.4. В данной схеме особо выделены разделы, которым посвящена работа. Толщина линии соответствует максимальному значению.

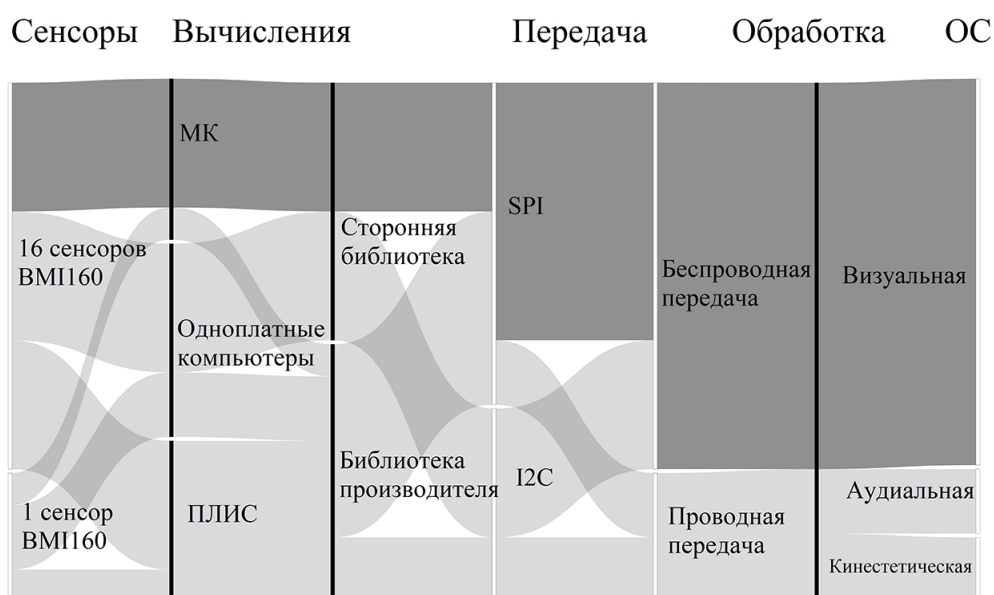


Рисунок 1.4 – Движение данных в зависимости от конфигурации блоков комплекса контроля движений.

В данной схеме можно выделить 5 блоков.

- *Блок получения сигнала о движении.* Можно рассматривать разные способы оцифровки движений по количеству датчиков.
- *Блок вычислений.* Разные типы вычислительных узлов, которые делятся на микроконтроллеры, одноплатные компьютеры, ПЛИС. Различные программные реализации для обработки данных.
- *Блок передачи.* Вид канала — разные шины данных (например, SPI или I2C). Способ передачи: беспроводной или нет. В первом случае

это может быть организация связи посредством Wi-Fi или Bluetooth, а также передача авторским способом с использованием радиоканала.

- *Блок обработки.* Обработка данных, приведение данных к заданной структуре, формирование набора эталонов движений, алгоритмы сопоставления цифровых паттернов движения с эталоном.
- *Блок обратной связи.* Форма представления результатов человеку.

## **Подходы в разработке комплексов контроля движений на МЭМС**

При проектировании аппаратных комплексов для цифровой регистрации движений разработчики обычно руководствуются тремя базовыми принципами.

Применение методики минимизации датчиков способствует конечной компактности системы, но может негативно отразиться на качестве распознавания жестов из-за упуска значимых признаков. Разработчики без проведения предварительного анализа не в состоянии определить, существуют ли такие значимые столбцы матрицы движения и где они могут быть найдены. Вследствие этого скорость проектирования оказывается ниже потенциально возможной.

Подход, основанный на максимизации, позволяет получить все возможные значимые столбцы. Однако это приводит к проблеме избыточности данных, что, в свою очередь, может привести к недостаточной частоте дискретизации. С увеличением числа датчиков степень избыточности возрастает нелинейно, что особенно заметно при добавлении в комплекс датчиков для отслеживания мелкой моторики, как это реализовано в костюмах компании Neuron с 32 датчиками (см. рис.1.1). Это обуславливает возможность определения границы числа датчиков, после которой эффективность работы комплекса значительно ухудшается, или когда системы видеонаблюдения начинают работать более эффективно.

Третий подход, являющийся наиболее распространенным, используется крупными поставщиками подобных решений, такими как Neuron и Xsens [76]. Следует отметить, что они часто используют двухточечную цифро-

вую регистрацию двух пальцев, в то время как остальные пальцы оцифровываются по одной точке. Однако для задач мелкой моторики может потребоваться более полное представление кисти или других частей тела человека. В данной работе предлагается универсальный подход, определяющий минимально необходимый набор датчиков для решения конкретных задач.

### **1.3 Проблемы применения инерционных датчиков и методик анализа их выходных сигналов**

#### **Алгоритмы обработки информации в комплексах контроля движений на инерциальных микроэлектромеханических датчиках**

##### **Алгоритмы ориентации в пространстве**

Для преобразования данных в углы поворота можно использовать различные открытые и проприетарные алгоритмы такие как:

- Маджвика;
- Махони;
- комплементарный первого и второго порядков;
- тригонометрического преобразования проекций на вектор гравитации «atan2»;
- алгоритм расчета угла только по гироскопу.

##### **Алгоритмы фильтрации данных о движении**

Для фильтрации данных о движении обычно применяют следующие алгоритмы:

- Калмана;
- плавающая средняя;
- Беттерворта;
- Бесселя;
- Барлетта;
- Барлетта–Ханна.

Далее будет приведено сравнение открытых алгоритмов и фильтров, используя следующую литературу [77–84].

### Сравнение самых цитируемых алгоритмов по уровню трансформации данных

Алгоритмы, которые традиционно используются в данной области, представлены в Таблице 1.3. Основной проблемой этих методов является их ограниченная способность к трансформации данных, которая приводит к уменьшению размерности матрицы данных максимум в 2 раза.

Это представляет значительный интерес, так как алгоритм рассматривается не как отдельный процесс трансформации данных, а как интегральная часть всего комплекса. После этапа преобразования данных следует этап сопоставления двух образцов движения, который имеет алгоритмическую сложность  $O(n^2)$ .

Таблица 1.3 – Сравнительный анализ алгоритмов трансформации данных о движении.

Алгоритм	Общая сложность вычислений системы контроля движений	Снижение общего числа столбцов матрицы с данными о движении после трансформации данных, раз
Madgwick <i>S. Madgwick, N. Nuscher, D. Gis, S.A. Ludwig</i>	$O(n \log(n)) + O(n^2)$	1,5
Mahony <i>R. Nahoby, T.K. Hamel, P.A. Absil, M. Henein</i>	$O(n) + O(n^2)$	1,5
Комплементарный фильтр <i>D. Capriglione, M. Carratu, А.М. Корилов</i>	$O(n) + O(n^2)$	2
Прямое интегрирование <i>A. Harindranath, M. Arora, Н.А. Ус, С.П. Задорожний</i>	$O(n / 2) + O(n^2)$	1

Это означает, что на этапе трансформации данных снижение размерности матрицы данных о движении приводит к нелинейному (квадратичному) увеличению производительности вычислений на этапе сопоставления.

Это самая важная часть, превосходящая по значимости сложность самого алгоритма трансформации данных. Ведь наша цель - оптимизация общей производительности комплекса, а не эффективности отдельных его компонентов.

Кроме того, для некоторых алгоритмов (в частности, для прямого интегрирования) отмечается ухудшение качества данных. При вычислении двойного интеграла (акселерометр) и простого интеграла (гироскоп) от выходного сигнала датчика, результирующий столбец данных может значительно отличаться от реальных данных. В этом случае шум может подавить всю информацию.

Различные алгоритмы применяются для устранения этой проблемы. Наша задача состоит не в определении абсолютных значений движений, а в сравнении трансформированных данных без интегрирования. Это позволяет использовать алгоритмы, которые с точки зрения вычислений являются более простыми.

Целью является разработка алгоритма, который с одной стороны позволит решить проблему

нулевого дрейфа, а с другой - снизить размерность цифрового паттерна движения без потери точности при сравнении движений. После этого следует разработать программное обеспечение и провести его апробацию.

### **Способы измерения степени схожести двух сигналов**

Можно выделить несколько способов оценки расстояния для заданного контекста.

1. Расстояние Минковского (как обобщение для трех указанных ниже):
  - (a) расстояние Евклида;
  - (b) расстояние городских кварталов;
  - (c) расстояние Чебышева.
2. Расстояние Левенштейна.
3. Косинусное расстояние.
4. Индекс Жаккара.

5. Расстояние Махаланобиса.
6. Глобально оптимальный алгоритм репараметризации<sup>1</sup>.
7. Алгоритм динамического искажения временной шкалы.

В данном исследовании сигналы не имеют фиксированного временного окна. Это означает, что одно и то же движение может быть выполнено как медленнее, так и быстрее записанного эталона. Кроме того, интенсивность различных этапов движения также может варьироваться. Таким образом, использование прямых метрик, таких как расстояния Махаланобиса, Минковского, Евклида, косинусное расстояние, расстояние городских кварталов и расстояние Чебышева, приведет к значительным ошибкам в случае сдвига по времени, растяжения или сжатия движения. Применение этих метрик непосредственно не подходит для сравнения двух сигналов. Расстояние Левенштейна имеет свою основную область применения в компьютерной лингвистике и биоинформатике, для сравнения слов некоторого алфавита (что не является актуальным для данного исследования). Индекс Жаккара применяется в случае работы с теорией множеств.

Остаются два подходящих метода для оценки расстояния между сигналами: алгоритм динамического искажения временной шкалы (DTW) и глобально оптимальный алгоритм репараметризации (GORA). DTW использует евклидово расстояние в своей логике работы, но учитывает особенности искажения данных на временной шкале. GORA был разработан как альтернатива DTW, работающая быстрее для данных о движении, но в данном исследовании данные получены [85] в виде траекторий в пространстве  $R^3$  из видео<sup>2</sup>.

Алгоритм динамического искажения временной шкалы широко применяется благодаря возможности настройки выбора конкретной метрики. В данном исследовании будет использован классический вариант алгоритма. Подробное описание алгоритма будет представлено далее.

---

<sup>1</sup>(англ.) Globally Optimal Reparameterization Algorithm

<sup>2</sup>Сигналы в виде реальных траекторий в пространстве  $R^3$  и векторизованных видеопоследовательностей с фиксированным числом фреймов данных. Источник: <https://arxiv.org/pdf/1807.05485.pdf>.

## Биомеханическое представление кисти человека

Простое биомеханическое представление кисти человека состоит из 16 твердых костей и связей между ними (анатомический аналог — суставы) [86]. Суставы представляют собой идеальные шарниры, моделирующие межкостные связи. Подробнее можно посмотреть на рис. 1.5. данной работе каждый шарнир декомпозируется по трем направлениям двух датчиков (гироскопа и акселерометра).

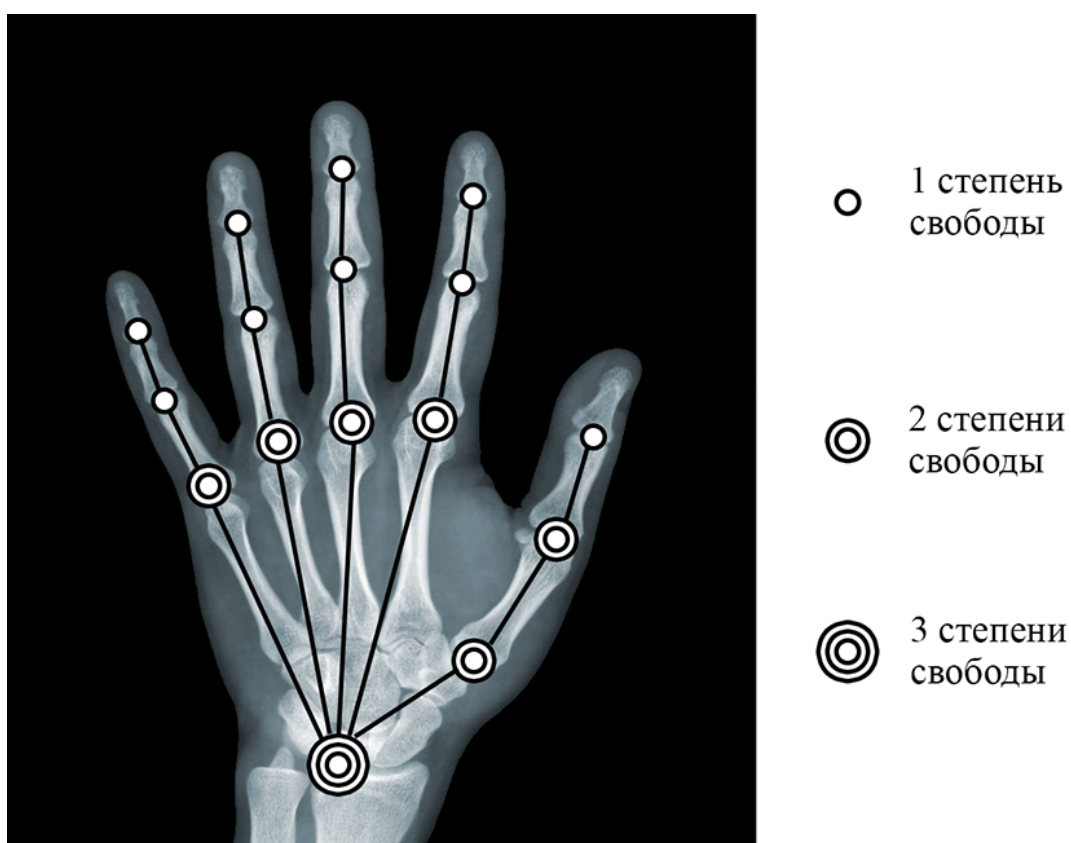


Рисунок 1.5 – Количество DoF вращательных движений для суставов кисти человека.

Это упрощение обосновано, так как оно позволяет объяснить большую часть возможных движений руки [87].

Из 48 теоретически возможных степеней свободы, биологически доступно лишь 24. Это подтверждает, что костная структура ограничивает половину потенциальных степеней свободы движений.

Важность этого наблюдения заключается в том, что в здоровом состоянии кисть не должна превышать эти ограничения. Более того, появление

движений в запрещенных направлениях указывает на физиологическую патологию.

Модель следует дополнить мышечными связями для получения ясного представления о дополнительных ограничениях на возможные движения кисти.

Дистальные<sup>3</sup> фаланги пальцев могут выполнять только сгибание и разгибание. Это также применимо к средним фалангам пальцев, за исключением большого. Проксимальные<sup>4</sup> фаланги II - V пальцев могут дополнительно выполнять отведение и приведение благодаря межкостным мышцам. Большой палец имеет больше дополнительных мышц. Из-за этого и особенностей костной структуры, он может быть противопоставлен остальным пальцам, что для нас означает большую DoF как в количестве возможных координат движения, так и в смысле диапазона самого движения.

Следует подчеркнуть, что кисть способна выполнять вращения по всем трем биомеханическим осям благодаря лучезапястному суставу. Для более детального изучения анатомии руки, включая потенциальный объем движений каждого сустава, можно обратиться к специализированной литературе, например, посвященной суставным заболеваниям [88]. В контексте обучения физическим упражнениям, как, например, в тхэквондо [89], следует также учитывать динамические характеристики, влияющие на параметры поворотов.

В рамках этого исследования мы исключаем некоторые связи из-за узкого спектра применения. Для полной модели интервал широк, но в мониторинге фокусируемся на определенных действиях, ожидая их наблюдение. Остальные движения считаем неиспользуемыми. [90].

**Определение 1.** Представим множество всех возможных DoF движения кисти  $Q$  как совокупность параметров  $N$ ,  $A$ ,  $\Omega$ , его характеризующих.  $N$  – множество DoF для суставов, в которых можно осуществлять нормальные действия.  $A$  – множество DoF для мышц и сухожилий, в нормальном рабочем диапазоне.  $\Omega$  – множество DoF, характерных

---

<sup>3</sup>Относительно центра масс кисти, это дальняя фаланга пальца.

<sup>4</sup>Относительно центра масс кисти, это ближняя фаланга пальца.



для патологических физических движений.

Тогда **минимальным набором параметров**  $PAR(Q)$  будем называть совокупность параметров, описывающих минимально необходимую совокупность DoF объекта  $Q$  с наименьшим числом мышечных, сухожильных и суставных и патологических движений, которые соответствуют целевым действиям по Н.А. Бернштейну.

$$PAR(Q) = \{Z \in N, H \in A, I \in \Omega, \},$$

где  $Z$  – множество DoF для суставов, в которых нужно осуществлять целевые действия;

$H$  – множество DoF для мышц и сухожилий, минимально необходимый для деятельности;

$I$  – множество DoF, характерных для патологических физических движений, должно быть пустым.

Стоит подробнее раскрыть смысл множества DoF для мышц и сухожилий. Движения одних частей кисти могут провоцировать перемещение других. Связано это с анатомическим строением кисти (рис.1.6).

Например, это можно проиллюстрировать для полного сгибания всех фаланг мизинца. В этом случае безымянный палец непременно будет согнут в некоторой степени вслед за мизинцем.

Пальцы I-IV руки связаны общим вложением [91]. При сгибании мизинца начинает натягиваться его сухожилие, влекущее за собой натяжение соответствующих сухожилий других пальцев. Однако, с увеличением расстояния от первичного источника натяжения, влияние этого эффекта уменьшается. Эту характеристику следует учитывать при разработке системы контроля движений, особенно с целью минимизации количества используемых датчиков. В этом контексте возможно усовершенствование методов обработки данных с привлечением биомеханических исследований, поскольку человеческое тело само по себе ограничивает некоторые виды движений (подробнее об этом принципе в идеологии физиологии активности описывает Н.А. Бернштейн [92]; эта идея впоследствии была развита в

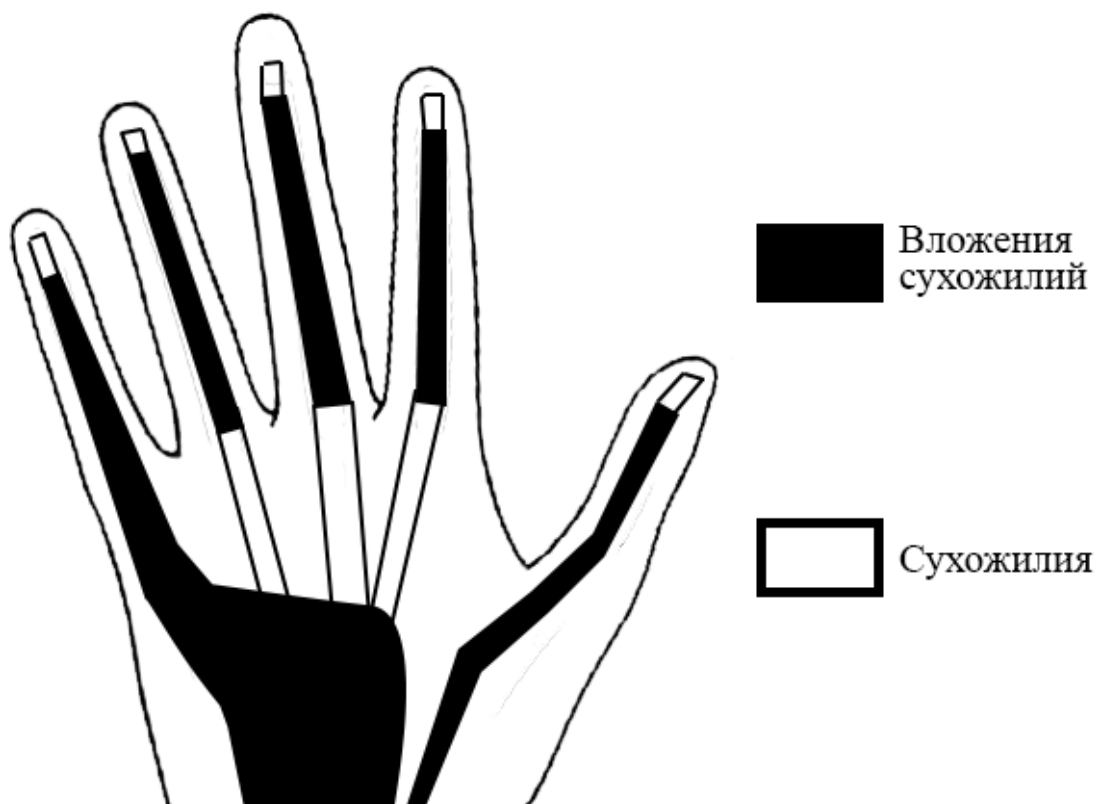


Рисунок 1.6 – Вложения сухожилий пальцев рук.

«теории деятельности») [93].

Если рассмотреть движения, которые не являются типичными для человека, они могут быть классифицированы следующим образом:

В первом случае мы имеем дело с различными вывихами, переломами, растяжениями. В зависимости от задачи, этот класс движений может быть принят во внимание или исключен из рассмотрения.

Второй случай включает в себя те движения, которые здоровый человек анатомически способен выполнить. Некоторые из них могут оказаться неактуальными, что приведет к недостаточной тренировке отдельных мышц. Примером может служить сгибание дистальных фаланг I-IV пальцев. Это движение в среднем люди выполняют редко, однако можно представить ситуацию, когда оно окажется необходимым.

В третьем случае будут затронуты движения, вызванные или поддерживаемые внешним воздействием [94]. Под это описание подходят нижеприведенные примеры.

- Сгибание фаланги пальца другой рукой. В этом случае можно расширить диапазон возможных углов сгиба в норме.
- Опора пальцами о плоскую поверхность.
- Фиксация отдельных частей руки.
- Использование различных инструментов. При физических упражнениях это означает, что человек выполняет упражнения не с собственным весом, а, например, с гантелями. В контексте обучения двигательным шаблонам это означает, что парикмахер взял ножницы, тем самым дополнительно ограничив перечень возможных действий.

В итоге можно утверждать, что для достижения результата необходимо провести 2 усечения:

- биомеханические ограничения;
- ограничения цели использования.

Идея заключается в предоставлении дополнительного канала ОС для сенсорной коррекции движения. Органы чувств сигнализируют мозгу о ходе движения, чтобы вносить коррекции. При нарушениях ОС или при освоении новых движений требуется моментальная коррекция, которую обеспечивает комплекс контроля движения. То же самое происходит, когда человеку приходится заново учиться какому-то движению: он вынужден вырабатывать новые и непривычные паттерны моментальной коррекции движения [92]

## 1.4 Постановка задач исследования

Анализ предметной области приводит к следующим заключениям:

Существует потребность в решении задачи соотнесения двух паттернов движения, которая, в отличие от задачи восстановления полной картины движения, может быть выполнена более простыми методами. В данном контексте мы имеем дело с ограниченным набором заранее известных классов движений и одним неизвестным движением, подлежащим классификации. Данная задача позволяет минимизировать избыточность в матрице данных о движении, исключая ненужные строки и столбцы.

По результатам анализа, наиболее подходящей технологией для решения задачи соотнесения движений являются МЭМС. Эта технология позволяет описывать движения в наиболее сжатой форме, легко адаптируется для мобильного использования, требует минимальной перекалибровки датчиков, предъявляет низкие требования к перекрытию конечностей и не сталкивается с проблемами освещения пространства. Аппаратная часть такого комплекса имеет много аналогов (например, XSens, Neuron) и требует лишь технической модификации для синхронного приема сигналов от датчиков, в то время как алгоритмическая часть требует усовершенствования в части уменьшения размерности матрицы данных о движении.

Традиционные алгоритмы обработки матрицы данных МЭМС о движении работают в условиях ограниченного числа датчиков, обеспечивая низкий уровень трансформации данных. В контексте задачи соотнесения двух цифровых паттернов движения это критично, так как в комплексе на первый план выходит общая вычислительная сложность. Время выполнения алгоритма сравнения цифровых паттернов движения очень чувствительно к размерности входящих паттернов движения (алгоритмическая сложность  $O(n^2)$ ). Поэтому важно разработать новые методы трансформации данных, целью которых будет сокращение размерности матрицы данных о движении, при этом необходимо доказать сохранение качества информации о действиях.

В этой связи цель идентификации движений при помощи разработки – программно-аппаратного комплекса движений кисти человека является актуальной, для решения нужно решить следующие задачи:

- Проанализировать особенности информационных сигналов, регистрируемых и выходных каналов датчиков комплекса контроля движений кисти человека.
- Разработать методики уменьшения размерности первичной информации без потери её качества.
- Провести экспериментальную апробацию разработанного программно-аппаратного комплекса.

## Глава 2

# Анализ особенностей информационных сигналов в комплексе контроля движений кисти человека

В современных системах контроля движения ключевую роль играет качество и точность сигнала, поступающего с датчиков. Понимание особенностей этого сигнала, его характеристик и потенциальных искажений является основой для эффективного анализа и последующей обработки данных. В данной главе акцентируется внимание на детальном анализе сигнала, полученного от акселерометра и гироскопа, что позволяет создать подходящую архитектуру программно-аппаратного комплекса для максимально точного отслеживания движений.

Важным этапом исследования является обоснование выбора конкретных моделей датчиков. Будет проанализирована степень влияния различных факторов на информационный сигнал, включая ошибки измерений, температурную зависимость и другие потенциальные искажения.

Заключительной частью главы станет демонстрация формирования матрицы с данными о движении. Этот этап позволит обосновать и продемонстрировать, как за методики хранения информации о движении можно учитывать дрейф нулевого уровня выходных каналов датчиков единожды.

## 2.1 Анализ основных характеристик информационных сигнала

### Исходные данные с акселерометра и гироскопа: особенности и параметры

Системы контроля движений основаны на сборе и анализе данных с различных датчиков. Акселерометры и гироскопы играют ключевую роль в измерении параметров движения. Целью данной работы является системный анализ и обработка информации, полученной от этих датчиков, для выявления характеристик движений и их коррекции.

Акселерометр измеряет линейное ускорение по каждой из координатных осей ( $m/c^2$ ). Данные с датчика можно представить в виде:

$$a(t) = [a_x(t), a_y(t), a_z(t)] \quad (2.1)$$

Гироскоп измеряет мгновенный вектор угловой скорости по координатным осям ( $^\circ/c$ ). Данные с гироскопа можно представить в виде:

$$\omega(t) = [\omega_x(t), \omega_y(t), \omega_z(t)] \quad (2.2)$$

Типичный сигнал, полученный с одного из выходов датчика генерирует сигнал вида (см. Рис. 2.1):

При хранении данных со всех выходов датчика (см. Таблицу 2.1), часто используется формат CSV. Типичный файл в этом формате для данных с акселерометра и гироскопа может выглядеть следующим образом:

Таблица 2.1 – Пример данных с акселерометра и гироскопа в формате CSV

Time	Accel_X	Accel_Y	Accel_Z	Gyro_X	Gyro_Y	Gyro_Z
0	0.001	0.002	9.81	0.01	0.02	0.03
0.01	0.002	0.003	9.80	0.02	0.03	0.04
...	...	...	...	...	...	...

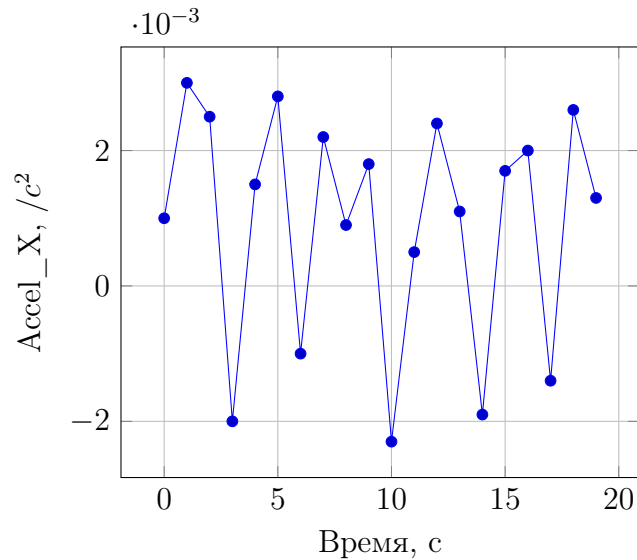


Рисунок 2.1 – Данные по Accel\_X

Где  $Time$  - это временные отметки,  $Accel\_X$ ,  $Accel\_Y$ ,  $Accel\_Z$  - данные акселерометра по трем осям, и  $Gyro\_X$ ,  $Gyro\_Y$ ,  $Gyro\_Z$  - данные гироскопа.

Анализ и обработка данных с акселерометров и гироскопов требуют системного подхода. Математическая постановка задачи позволяет формализовать процесс обработки, а использование стандартизированных форматов, таких как CSV, обеспечивает удобство и универсальность в программной обработке.

Для синхронизации момента начала движения и момента регистрации сигнала можно рассмотреть два основных подхода. Первый метод базируется на автоматическом определении активности на основе анализа пороговых значений сигналов датчика, в то время как второй метод использует внешние события, например, нажатие на кнопку, для начала и конца регистрации.

### **Синхронизация момента начала движения и сигнала**

Допустим,  $t_m$  - момент начала движения, а  $t_s$  - момент начала регистрации сигнала. Целью является минимизация разницы  $\Delta t = t_s - t_m$ . В идеальных условиях  $\Delta t = 0$ , но из-за различных задержек в системе, такое равенство редко достижимо.

Если рассматривать *кнопку начала движения* как механизм срабатывания, то можно установить пороговое значение  $\theta$ , таким образом, что при  $a(t) > \theta$ , где  $a(t)$  - ускорение в момент времени  $t$ , система начинает регистрацию сигнала.

$$\begin{aligned} t_s &= t \quad \text{если} \quad a(t) > \theta \\ t_s &\neq t \quad \text{иначе} \end{aligned} \tag{2.3}$$

Таким образом, событие срабатывания кнопки фиксируется при превышении установленного порога, и в этот момент блокируется часть памяти для сохранения данных.

Срабатывание по внешнему событию использует внешние стимулы, такие как нажатие на *кнопку начала движения*, для начала регистрации. В этом случае, момент регистрации  $t_s$  определяется внешним событием  $e(t)$ :

$$\begin{aligned} t_s &= t \quad \text{если} \quad e(t) = \text{событие активировано} \\ t_s &\neq t \quad \text{иначе} \end{aligned} \tag{2.4}$$

В целях упрощения задачи выбран второй вариант. Таким же образом завершается запись сигнала.

### **Распределение 16 датчиков на кисти руки**

На основе предыдущих исследований с одним датчиком, в реальной практике для создания полной скелетной модели кисти требуется 16 датчиков. Был проведен анализ биомеханических моделей кисти в первой главе диссертационной работы и было показано, что кисть можно описать в простом случае из 16 суставов, таким образом нужно 16 гироскопов и 16 акселерометров (в совмещенном датчике). Это минимальное количество датчиков, которое позволяет достоверно описать все степени свободы кисти человека.

1. **Датчики:** Пусть  $S = \{s_1, s_2, \dots, s_{16}\}$  представляет собой множество датчиков.



2. **Позиционирование:** Для каждого датчика  $s_i$ , его позиция в трехмерном пространстве задается как  $P(s_i) = (x_i, y_i, z_i)$ , где  $(x_i, y_i, z_i)$  — координаты в декартовой системе координат.
3. **Данные с датчика:** Для каждого датчика  $s_i$ , данные с акселерометра и гироскопа представлены в виде:
  - $A(s_i) = (a_{xi}, a_{yi}, a_{zi})$  для акселерометра.
  - $G(s_i) = (g_{xi}, g_{yi}, g_{zi})$  для гироскопа.
4. **Особенности размещения:** При размещении датчиков на кисти важно учитывать анатомические и функциональные особенности кисти, а также избегать возможного перекрытия датчиков.

При дальнейшем разработке системы, каждый из этих датчиков будет использован для сбора данных, которые затем будут обрабатываться для получения полной скелетной модели движения кисти.

**Тестовый сигнал** Тестовый сигнал представляет собой запись всех выходных данных со всех датчиков за определенный период времени — от момента нажатия кнопки старта до момента нажатия кнопки завершения записи. Эта запись содержит как данные с акселерометра, так и с гироскопа каждого из датчиков.

**Определение 2.1.1.** Для множества датчиков  $S = \{s_1, s_2, \dots, s_{16}\}$  и заданного временного интервала  $[t_{start}, t_{end}]$ , тестовый сигнал  $T$  определяется как:

$$T(s_i, t) = \begin{cases} A(s_i, t) & \text{для акселерометра} \\ G(s_i, t) & \text{для гироскопа} \end{cases} \quad (2.5)$$

где

$$t_{start} \leq t \leq t_{end}$$

Таким образом, тестовый сигнал содержит информацию о движении и ориентации кисти в заданном временном интервале.

**Эталонный сигнал** В системах анализа техники движения высокого уровня мастерства, таких как игра на пианино или другие сложные моторные задачи, важнейшую роль играет определение эталонных паттернов движений. Эти паттерны отражают наиболее типичные и оптимальные особенности движений специалиста, обладающего высокой степенью мастерства.

### Методика формирования эталонного сигнала

1. *Запись сигналов*: Серия измерений движений эксперта проводится с помощью специализированных датчиков. Каждое измерение представляет собой матрицу с данными о движении.
2. *Обработка данных и усреднение*: На основе тестовых сигналов формируется эталонный сигнал. Процедура усреднения может быть выполнена различными способами, в зависимости от конкретной задачи:
  - (а) **Простое арифметическое усреднение**: В этом случае для каждого временного интервала рассчитывается среднее значение по всем тестовым записям.

$$\mu(t) = \frac{1}{N} \sum_{i=1}^N x_i(t) \quad (2.6)$$

где  $N$  - количество записей,  $x_i(t)$  - значение  $i$ -ой записи в момент времени  $t$ .

- (b) **Взвешенное усреднение**: При этом методе каждому тестовому сигналу присваивается определенный вес, основанный на его важности или достоверности.

$$\mu(t) = \frac{\sum_{i=1}^N w_i x_i(t)}{\sum_{i=1}^N w_i} \quad (2.7)$$

где  $w_i$  - вес  $i$ -ой записи.

- (c) **Медианное усреднение**: Вместо среднего значения для каждого временного интервала выбирается медианное значение из всех тестовых записей. Этот метод устойчив к выбросам и может быть полезен при наличии аномальных записей.

- (d) **Геометрическое усреднение:** Используется в случаях, когда необходимо усреднить произведения значений.

$$\mu(t) = \sqrt[N]{\prod_{i=1}^N x_i(t)} \quad (2.8)$$

С помощью выбранной процедуры усреднения формируется эталонный сигнал, который представляет наиболее типичное поведение эксперта и может быть использован для последующего сравнения с другими сигналами.

### **Согласование длительности сигналов**

Часто встречающейся проблемой при обработке сигналов является различие в длительности. Такое несовпадение может возникнуть из-за разных факторов, включая разную продолжительность выполнения одного и того же движения разными исполнителями или даже одним и тем же исполнителем в разное время.

Для решения этой проблемы применяется следующий процесс:

1. *Определение максимальной длительности сигнала:* Из всех доступных записей выбирается сигнал с максимальной длительностью, который будет служить эталоном для последующей интерполяции.
2. *Интерполяция данных:* Остальные сигналы интерполируются так, чтобы их длительность соответствовала длительности максимально продолжительного сигнала. Интерполяция может осуществляться с использованием различных методов, например, линейной интерполяции. Результатом является набор сигналов одинаковой длительности.
3. *Применение метода усреднения:* После интерполяции, один из вышеупомянутых методов усреднения применяется к согласованным сигналам для получения эталонного сигнала.

В данной работе применяется интерполяция и простое арифметическое усреднение для максимальной простоты.

### **Цель обработки**

Основной целью обработки данных является получение четкой и надежной информации о характере движения кисти, минимизируя искажения, вызванные шумами и погрешностями датчиков. В этом контексте основной задачей системного анализа стоит определение оптимального баланса между объемом исходных данных для обработки и вычислительной эффективностью на этапе сравнения.

**Проблематика выбора объема данных** С одной стороны, качественное усреднение и, следовательно, более точное представление о типичном движении требует большого объема реальных данных. Это позволяет учитывать множество нюансов и вариаций движения, делая эталонный сигнал репрезентативным.

С другой стороны, алгоритмы сравнения, особенно в условиях реального времени, должны быть максимально быстрыми. Алгоритмы с вычислительной сложностью  $O(n^2)$  квадратично увеличивают время обработки с увеличением объема данных, что может стать критическим для систем, где требуется быстрый отклик.

**Поиск точки баланса** В данном контексте, системный анализ направлен на поиск оптимального компромисса между объемом данных для усреднения и вычислительной эффективностью на этапе сравнения. Такой баланс позволит получить максимально точное представление о движении при минимальных затратах на обработку в реальном времени.

Пусть  $N$  — количество данных,  $E(N)$  — ошибка усреднения на основе  $N$  данных, а  $C(N)$  — вычислительная сложность алгоритма сравнения для  $N$  данных.

Целью является минимизация ошибки усреднения  $E(N)$  и одновременно минимизация вычислительной сложности  $C(N)$ . Введем функцию потерь  $L(N)$ :

$$L(N) = \alpha E(N) + \beta C(N) \quad (2.9)$$

где:

- $\alpha$  и  $\beta$  — весовые коэффициенты, которые определяют относительную важность ошибки усреднения и вычислительной сложности, соответственно.

Таким образом, цель оптимизации:

$$\min_N L(N) \tag{2.10}$$

Для решения этой задачи оптимизации:

1. **Ошибка усреднения,  $E(N)$ :** Это функция, которая убывает с увеличением  $N$  (больше данных обычно приводит к меньшей ошибке). Она может быть представлена, например, гиперболической или экспоненциальной функцией, где при  $N \rightarrow \infty, E(N) \rightarrow 0$ .

2. **Вычислительная сложность,  $C(N)$ :** В условиях экспоненциальной сложности, это может быть функцией вида  $C(N) = 2^N$ .

## Влияние факторов на сигнал

Факторы, влияющие на данные указываются производителем датчиков. В начале мы рассмотрим их без привязки к конкретной модели датчика, потом выберем наиболее подходящий из них, а позже проверим самые значимые параметры.

**Дрейф нулевого уровня выходного значения датчика** Дрейф нулевого уровня выходного значения датчика определяет изменение выходного сигнала датчика при отсутствии внешних воздействий. Это ключевая характеристика, которая влияет на общую надежность и точность измерений. См. рис.2.2).

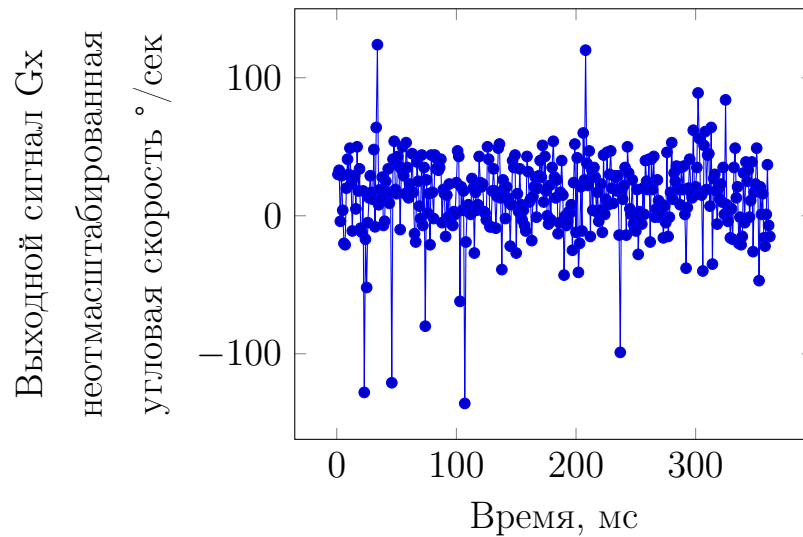


Рисунок 2.2 – Дрейф нулевого уровня выходного значения гироскопа Gx

Все МЭМС датчики имеют такую характеристику и возьмем датчик VM160 в качестве иллюстрации степени влияния этого параметра на выходные данные датчика.

В рассмотренных данных, представляющих собой выходной сигнал гироскопа Gx, было проведено статистическое исследование с целью определения дрейфа нулевого уровня. Для этой цели были рассчитаны следующие статистические показатели:

Для последовательности  $x_1, x_2, \dots, x_n$ , где  $n$  — общее число измерений, среднее значение  $\bar{x}$  определяется как:

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i \quad (2.11)$$

На основе данных было установлено, что  $\bar{x} \approx 13.98$ .

Стандартное отклонение  $s$  вычисляется по формуле:

$$s = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (x_i - \bar{x})^2} \quad (2.12)$$

На основе данных было установлено, что  $s \approx 28,49$ .

Исходя из полученных значений, был рассчитан 95% доверительный интервал для среднего значения. Границы доверительного интервала опреде-

ляются следующим образом:

$$\bar{x} \pm t \frac{s}{\sqrt{n}} \quad (2.13)$$

где  $t$  — критическое значение t-статистики Стьюдента на заданном уровне доверия (для 95% и  $n - 1$  степенями свободы).

Нижняя граница доверительного интервала 11,05, а верхняя граница - 16,92. Истинное значение равняется 0,00. Таким образом можно говорить, что данные смещены по значению при стабильных условиях без использования аппаратной части. Были использованы реальные данные с датчика в режиме покоя.

**Смещение.** Это стандартное отклонение или средняя ошибка измерения датчика при отсутствии внешнего воздействия. Например, акселерометр в состоянии покоя может показывать ненулевое значение из-за смещения.

Для акселерометра может составлять порядка  $\pm 40mg$ , а для гироскопа  $\pm 3^\circ/s$ .

**Пропуск данных** Проблемы с коммуникацией или низкие частоты дискретизации могут привести к пропускам данных (см. на Рис. 2.3), которые могут исказить исходный сигнал.

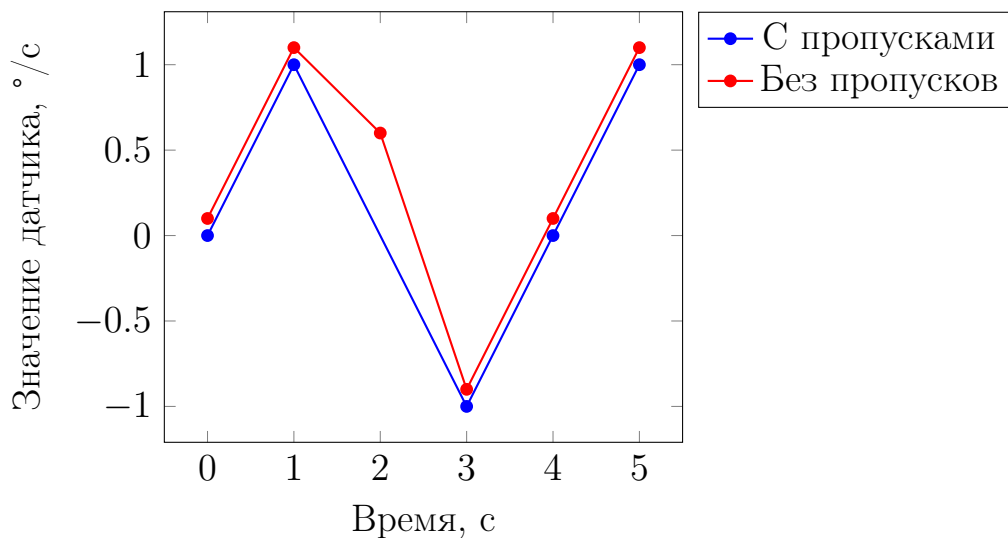


Рисунок 2.3 – Пропуски данных

На второй секунде виден пропуск данных, который искажает информацию во время формирования эталонного движения, что негативно скажется на сравнении эталонного и тестового сигналов в реальной работе.

**Нелинейность.** Этот параметр описывает степень, в которой фактическая характеристика датчика отклоняется от идеальной линейной характеристики. В контексте акселерометров и гироскопов это может быть важным показателем, так как идеальное устройство будет иметь 100% линейный отклик. Значение, указанное в процентах, показывает, насколько максимально значение может отклоняться от идеального линейного ответа в пределах диапазона измерений (FS - full scale, полный масштаб).

Для акселерометра может составлять порядка  $0.5\%FS$ , а для гироскопа  $0.1\%FS$ .

**Температурная зависимость.** Температурное смещение указывает, насколько значение смещения может изменяться с изменением температуры. Это особенно важно в приложениях, где датчик подвергается различным температурным условиям, так как изменение температуры может вызвать дополнительные ошибки в измерениях. Использование комплекса планируется в первую очередь в неподготовленном помещении, поэтому в работе принято решение использовать оборудование при стандартной температуре. При этом оценка влияния этого параметра может быть для акселерометра порядка  $\pm 1.0mg/K$ , а для гироскопа  $0.05^\circ/s/K$ .

**Плотность шума.** Этот параметр описывает уровень флуктуаций в выходном сигнале датчика. Он измеряется в единицах амплитуды на квадратный корень частотного диапазона.

Для акселерометра может составлять порядка  $180\mu g/\sqrt{Hz}$ , а для гироскопа  $0.007^\circ/s/\sqrt{Hz}$ .

Приведем пример для гироскопа на Рис. 2.4.



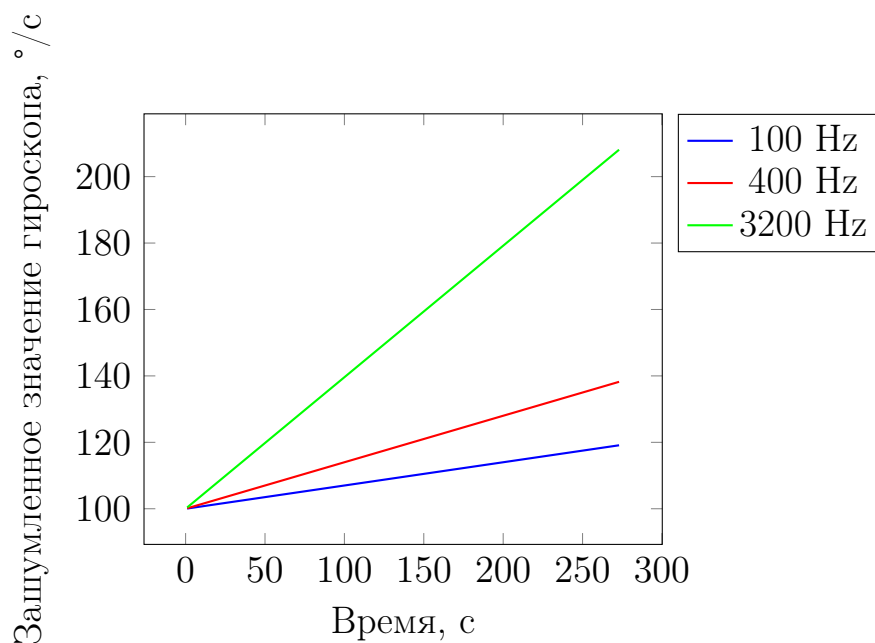


Рисунок 2.4 – Зависимость шума от частоты

Допустим, истинное значение равняется стабильным  $100^\circ$  в секунду, тогда при работе на разной частоте получатся вышеописанные данные на протяжении 200 секунд при соответствующей плотности шума.

#### Оценка шумов в данных

Для оценки шумов используются различные методики анализа [95,96] (см. подробное сравнение в Табл. 2.2).

Каждый из методов хорошо определяет наличие различных шумов в данных. Необходимо учесть особенности для применения к данной работе. В спектральном анализе требуется достаточно большая выборка для полного анализа и необходимо условие стационарности сигнала, что нельзя гарантировать.

Таблица 2.2 – Сравнение параметров датчиков

Название методики	Какие шумы идентифицирует	Необходима ли стационарность сигнала	Необходимая длина выборки	Позволяет ли оценить данные на разном временном масштабе
Fourier Spectral Analysis Спектральный анализ Фурье	1) Белый шум. Лучше всего определяет этот шум. 2) Фликкер шум. 3) Шум случайного блуждания.	Да	Большая	Нет
Bretthorst Periodogram (спектральная плотность мощности)	1) Белый шум. 2) Фликкер шум. 3) Шум случайного блуждания.  Точнее различает близкие частотные пики, чем спектральный анализ Фурье	Нет	Средняя	Нет
Вариация Аллана	1) Шумы, связанные с квантованием выходного сигнала. 2) Белый шум. 3) Фликкер-шум. 4) Винеровский процесс. 5) Линейный дрейф.	Нет	Маленькая	Да

Таким образом самым информативным способом анализа - будет вариация Аллана. Была произведена собственная оценка уровня шумов на одном из датчиков по выбранной шкале для исходных данных, фильтра Калмана, алгоритма Маджвика, фильтра плавающей средней, комплементарного фильтра первого порядка, алгоритма расчета угла только по гироскопу и алгоритма расчета угла только по акселерометру. На рис. 2.6 приведен график для исходных данных, а рис.2.5 содержит оценки для вышеупомянутых алгоритмов и фильтров.

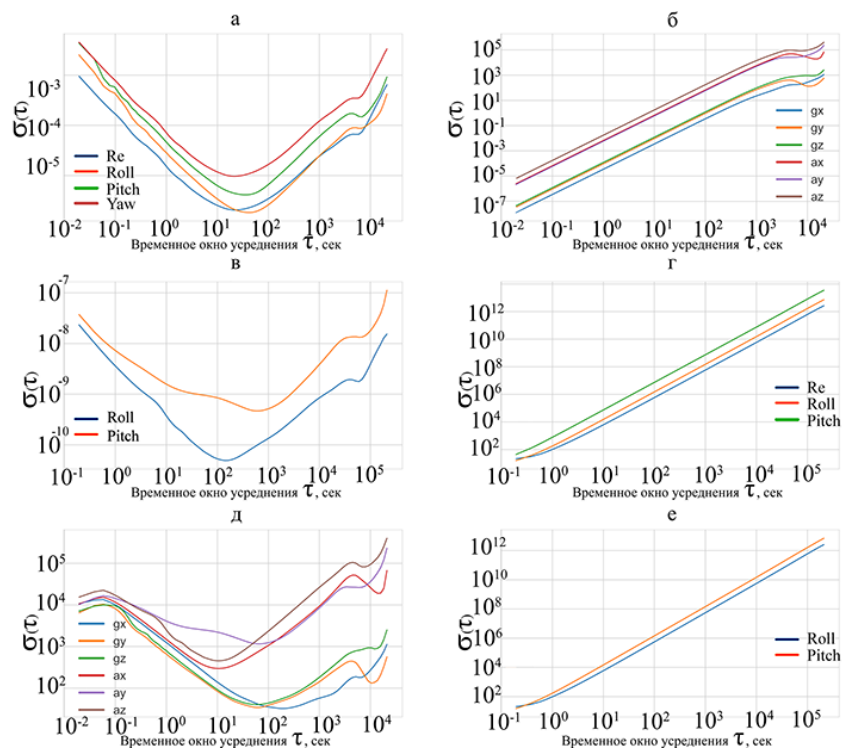


Рисунок 2.5 – Вариация Аллана для датчика ВМ160: а) алгоритм Маджвика; б) фильтр Калмана с коэффициентом 3,29; в) расчет только по акселерометру; г) расчет только по гироскопу; д) фильтр плавающей средней для окна в 3 значения; е) комплементарный фильтр.

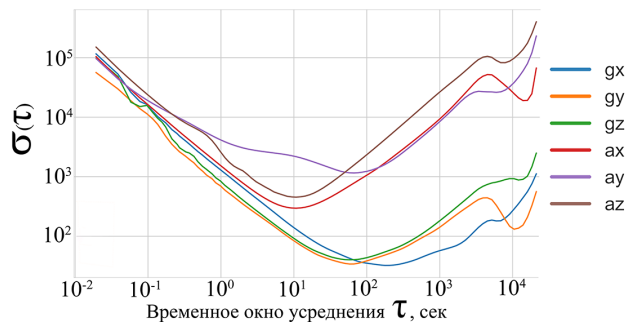


Рисунок 2.6 – Вариация Аллана для необработанных данных датчика ВМ160.

Сбор данных для сравнения осуществлялся в состоянии покоя на протяжении трех часов. В идеальном случае все показания для гироскопа должны быть нулевыми. На практике в таком режиме можно наблюдать интегральную шумовую компоненту в данных датчика, которая приводит к ошибке интегрирования, в чистом виде. Для акселерометра значения проекции на вектор гравитации не будут нулевыми, однако останутся стабиль-

ными. Используемые производные данные от акселерометра и гироскопа также должны быть стабильными.

Обозначения на графиках расшифровываются следующим образом.  $A_x$  – проекция на вектор гравитации для направления  $X$ .  $A_y$  – проекция на вектор гравитации для направления  $Y$ .  $A_z$  – проекция на вектор гравитации для направления  $Z$ .  $G_x$  – угловая скорость гироскопа для направления  $X$ .  $G_y$  – угловая скорость гироскопа для направления  $Y$ .  $G_z$  – угловая скорость гироскопа для направления  $Z$ .  $Re$  – действительная часть кватерниона.  $Roll$  – крен.  $Pitch$  – тангаж.  $Yaw$  – рыскание.

Лучшим алгоритмом по качеству является фильтр Калмана. Худшим алгоритм расчета угла по гироскопу. Однако фильтр Калмана является самым требовательным по вычислительным мощностям. Согласно литературным источникам данные результаты согласованы с мировой и отечественной практикой [78, 97–101].

Ни один из алгоритмов не подходит для решения поставленной задачи. В каких-то мы теряем информацию о повороте вокруг одной из пространственных осей, какие-то алгоритмы только в 2 и меньше раз уменьшают количество столбцов. Самый лучший фильтр Калмана - отлично справляется с подавлением шумов, однако является слишком вычислительно сложным для автономного устройства и не занимается уменьшением размерности даже столбцов. Половина алгоритмов показывает, что само усреднение данных - помогает снизить уровень ошибки в данных при временном окне усреднения до  $10^1$ с для акселерометра и до  $10^2$ с для гироскопа. Это можно будет использовать для разработки собственной методики трансформации матрицы с данными о движении как по столбцам, так и по строкам.

Имея ввиду все эти факторы, нужно такое алгоритмическое решение, которое будет значительно превышать (более 10 Дб в соотношении сигнал/шум) уровень всех имеющихся шумов для стабильного сравнения эталонного и тестового сигналов.

## **Обоснование выбора датчиков для решения задачи**

Основные требования к выбору датчиков:

- Частота дискретизации более 40 Гц. (Связано с правилом Найквиста и тем, что интересующий класс движений лежит в диапазоне от 0,2 до 20 Гц [102])
- На одном чипе должны быть реализованы как 3 осевой гироскоп, так и 3 осевой акселерометр. Это связано с тем, что при запросе данных надо делать 1 запрос, а не 2 разных. И один ответ кодируется компактнее двух из-за особенностей формирования пакетов с данными, согласно протоколам из документации датчиков.
- Дискретные выходные каналы сенсоров. Это связано с тем, что в противном случае придётся от каждого сенсора вести независимые линии вплоть до МК, вместо того, чтобы организовать общую шину передачи данных.
- Минимальное энергопотребление.
- Работа с протоколом SPI.
- Чувствительность гироскопа от  $\pm 125/s$ .
- Чувствительность акселерометра от  $\pm 2g$  до  $\pm 16g$ .
- Минимальный дрейф данных.
- Минимальная плотность шума.

В Табл. 2.3 акселерометры обозначены (акс.), а гироскопы (гиро.).

Таблица 2.3 – Сравнение параметров датчиков

Название датчика	Частота дискретизации акс., giro., Гц	На одном чипе акс., giro.	Дискретные выходные каналы датчика	Энергопотребление в активном режиме, ма	Протокол SPI	Чувствительность мин, акс. (g) giro. ( $^{\circ}/s$ )	Дрейф данных, акс. (mg), giro. ( $^{\circ}/сек$ )	Плотность шума акс., акс. ( $\mu g/\sqrt{Hz}$ ), giro. ( $^{\circ}/s/\sqrt{Hz}$ )
<b>BMI160</b>	<b>Акс.: до 3200 Гиро.: до 1600</b>	<b>Да</b>	<b>Да</b>	<b>0,925</b>	<b>Да</b>	<b>Акс.: 2 Гиро: 125</b>	<b>Акс.: 40 Гиро: 3</b>	<b>Акс.: 180 Гиро: 0,007</b>
MPU-9250	Акс.: 4000 Гиро: 4000	Да	Да	3,930	Да	Акс.: 2 Гиро: 250	Нет данных	Акс.: 300 Гиро: 0.01
ADXL335	Акс.: 1600 Гиро: Нет	Нет	Нет	0.35	Нет	Акс.: 3 Гиро: нет	Акс.: нет данных Гиро: нет	Акс.: 150 - 300 Гиро: нет

Выбраны датчики Bosch Sensortec BMI160<sup>5</sup>, которые использованы в

<sup>5</sup>Официальная документация на BMI160 [https://ae-bst.resource.bosch.com/media/\\_tech/](https://ae-bst.resource.bosch.com/media/_tech/)

количестве 16 штук.

Данные чипы используют и другие исследовательские группы [103–105]. Каждая плата представляет собой одновременно и трехосевой акселерометр, и трехосевой гироскоп. Гироскоп получает значения угловой скорости, а акселерометр углового ускорения — также в динамике.

Имеются и встроенные аналого-цифровые преобразователи. Выходной сигнал — цифровой 16 бит, один для гироскопа и еще один для акселерометра. Используемое напряжение питания — 3,3 В. Потребление каждого датчика — 925 мкА. Диапазон частоты работы гироскопа — от 25 до 3200 Гц. Диапазон частоты работы акселерометра — от 25 до 1600 Гц. Встроенный вычислительный блок может частично компенсировать дрейф нуля гироскопа в автоматическом режиме. Возможно по отдельности настраивать рабочие диапазоны работы устройства для повышения точности конечных данных. Возможно использование двух каналов передачи на МК: SPI и I2C. В данной работе используется первый вариант.

Комплекс имеет встроенный блок приведения аналоговых данных к цифровому виду, пригодному для дальнейшей передачи. Эта функция повышает помехоустойчивость блока. Также внизу слева можно заметить вторичный интерфейс, к которому можно дополнительно подключить магнитометр. В данной работе эта часть не используется. Связано это с тем, что, во-первых, магнитометры чувствительны к наличию поблизости электроприборов и других источников магнитных излучений, а во-вторых, частота их работы несопоставима с акселерометром и гироскопом. Например, у более оснащенной линейки BM055 магнитометр работает с максимальной частотой 30 Гц. Некоторые исследовательские группы работают на этих датчиках с частотой 10 Гц [106]. Также есть встроенный счетчик шагов, имеющий более низкое энергопотребление, чем требует стандартный режим работы. Присутствует работа с прерываниями. Данные функции также не используются ввиду отсутствия необходимости.

---

[media/datasheets/BST-BMI160-DS000.pdf](#)

## 2.2 Разработка алгоритмического обеспечения для анализа сигналов

### Выбор базового способа измерения двух сигналов

Комплекс контроля движений предполагает наличие нескольких предварительно записанных цифровых эталонных паттернов движений. В последующем, при использовании этого комплекса, новые данные сравниваются с эталонами, и требуется определить класс, к которому относится неизвестный цифровой паттерн движения. Для этого производится сравнение тестового сигнала со всеми эталонными с целью измерения степени их схожести.

Математическим образом можно представить это следующим образом:

Пусть у нас имеется множество  $E$  из  $n$  эталонных цифровых паттернов движений:

$$E = \{e_1, e_2, \dots, e_n\} \quad (2.14)$$

И пусть у нас есть тестовый цифровой паттерн движения  $x$ , который нужно классифицировать.

Тогда мы можем оценить степень сходства (similarity) между тестовым цифровым паттерном движений  $x$  и каждым эталонным цифровым паттерном движений  $e_i$ , используя некоторую функцию расстояния или схожести  $\mathcal{D}$ :

$$\text{similarity}(x, e_i) = \mathcal{D}(x, e_i), \quad i = 1, \dots, n \quad (2.15)$$

Задача заключается в том, чтобы найти класс  $C$ , к которому относится тестовый цифровой паттерн движения  $x$  на основе сравнения его со всеми эталонными цифровыми паттернами движений. То есть:

$$C = \text{argmax}(\text{similarity}(x, e_i)) \quad (2.16)$$

Таким образом, комплекс контроля движений позволяет определить класс неизвестного цифрового паттерна движения на основе анализа степени его схожести со всеми эталонными цифровыми паттернами движений.

Когда сигналы уже сохранены в специальном формате хранения, то они становятся уже цифровым паттерном движения.

Сигнал в системном анализе обычно представляет собой временной ряд или функцию, описывающую изменение некоторой величины во времени. Математически, непрерывный сигнал может быть представлен как функция  $(x(t))$ , где  $(t)$  - время, а значения  $x(t)$  определены для всех моментов времени из определенного интервала. Дискретный сигнал имеет значения только в конкретных дискретных моментах времени и может быть записан как последовательность значений:

$$x[n] = x(nT_s) \quad (2.17)$$

, где  $(n)$  - целочисленное значение индекса, а  $(T_s)$  - период дискретизации.

Цифровой паттерн обычно относится к конкретной последовательности дискретных значений, которая используется для описания информации или события. Он может быть связан со специальными алгоритмами или методами анализа данных. В контексте сохраненных значений сигнала в виде матрицы CSV, мы можем рассматривать эту матрицу как цифровой паттерн движения объекта или системы контроля на протяжении определенного периода времени.

Давайте представим это формально: Предположим, у нас есть сохраненные значения сигнала в виде матрицы  $(M)$ , где каждый элемент  $M_{ij}$  представляет значение датчика для конкретного момента времени и определенного датчика. Мы можем записать эту матрицу как:

$$M = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1m} \\ x_{21} & x_{22} & \dots & x_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{nm} \end{bmatrix} \quad (2.18)$$

Где  $n$  - количество моментов времени, а  $m$  - количество датчиков. Таким образом, мы можем рассматривать данную матрицу CSV как цифровой паттерн движения объекта или системы контроля на основании значений



датчиков в различные моменты времени. Этот паттерн может быть проанализирован и использован для изучения свойств движения объекта или системы контроля при помощи соответствующих методов анализа данных.

Каждый цифровой паттерн движения представляет собой последовательно измеренные данные, записанные через определенные временные промежутки, или временной ряд в трактовке Давида Бриллинджера [107] (в переводе под редакцией А.Н. Колмогорова) и Джорджа Бокса [108].

Экспериментальные данные относительно величины промежутков между измерениями были получены ранее (см. рис.5.8). Наиболее распространенной частотой получения данных было 212 Гц. Интервалы измерений имеют почти равномерный характер: отклонение около моды незначительное, что позволяет продолжать рассматривать имеющиеся данные о движении как подходящие временные ряды.

Ранее была проведена оценка стабильности частоты поступления данных от системы контроля движений в состоянии покоя (см. рис.2.5) с использованием методики анализа дисперсии Аллана. Были предоставлены временные оценки дрейфа нулевых значений MEMS.

Далее в главе примем следующие обозначения:

1.  $y$  – эталонный цифровой паттерн движения. Представляет из себя временной ряд, который заранее записан с помощью системы контроля движений.
2.  $x$  – исследуемый цифровой паттерн движения. Представляет из себя неизвестный по классу временной ряд, который фиксирует система контроля движений система.
3.  $N$  – длина временного ряда  $y$  ряда.
4.  $M$  – длина временного ряда  $x$  ряда.

Для начала сравним сопоставления цифровых паттернов с помощью евклидовой метрики и алгоритмом DTW (см. на рис.2.7) в метрическом пространстве  $\mathbb{R}^1$  как самый наглядный пример. Сравниваются две разные записи одного движения.

Во время сопоставления с применением евклидовой метрики каждый

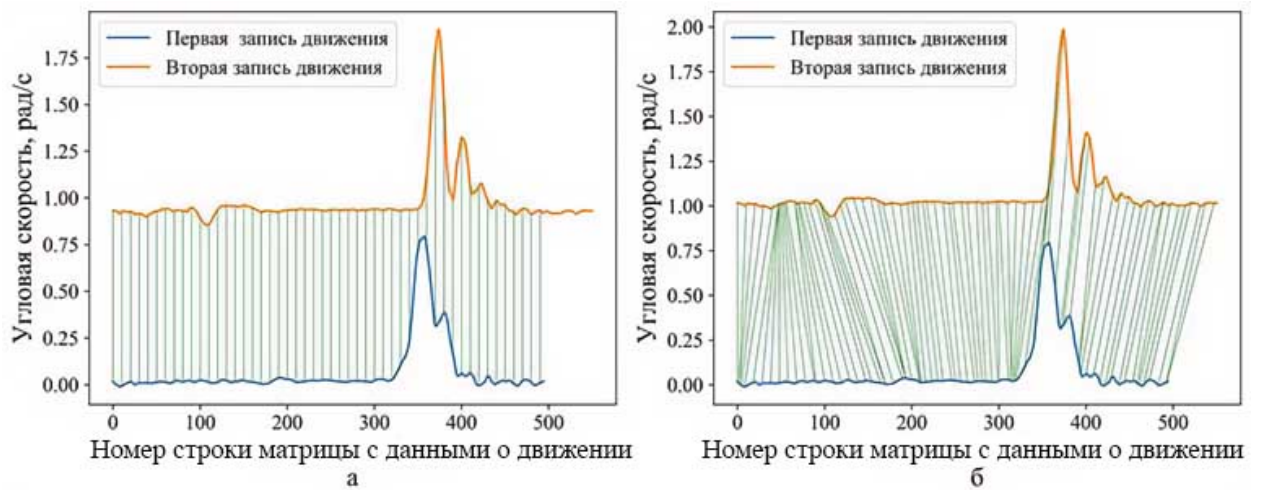


Рисунок 2.7 – Сопоставление цифровых паттернов: (а) по евклидовой метрике; (б) по DTW.

цифровой паттерн сравнивается с другим в этот же момент времени. Сопоставление производится один к одному. Это ведет к нескольким следствиям.

1. Существует чувствительность к сдвигу по времени.
2. Существует чувствительность к временным искажениям. Например, когда движение выполняют медленнее или быстрее эталона  $y$ .
3. Края цифровых паттернов могут не совпадать друг с другом.

В случае алгоритма DTW края двух цифровых паттернов всегда совпадают. Возможны сопоставления «один ко многим». Это позволяет сравнивать разные по продолжительности паттерны движения.

В алгоритме DTW вычисляются попарные расстояния в  $n$ -мерном арифметическом евклидовом пространстве  $\mathbb{R}^n$  между всеми элементами обоих цифровых паттернов в смысле определения А.Н. Колмогорова и С.В. Фомина [109]. Визуально этот процесс для пространства  $\mathbb{R}^1$  представлен на рис. 2.8 и рис. 2.9.

Таким образом, построчно составляется матрица локальных расстояний.

Необходимая размерность исходного пространства выбирается, исходя из размерности сравниваемых цифровых паттернов. Так, в случае изначального количества датчиков и их выходных каналов для системы контроля движений это число равняется 96. Каждый столбец матрицы движе-

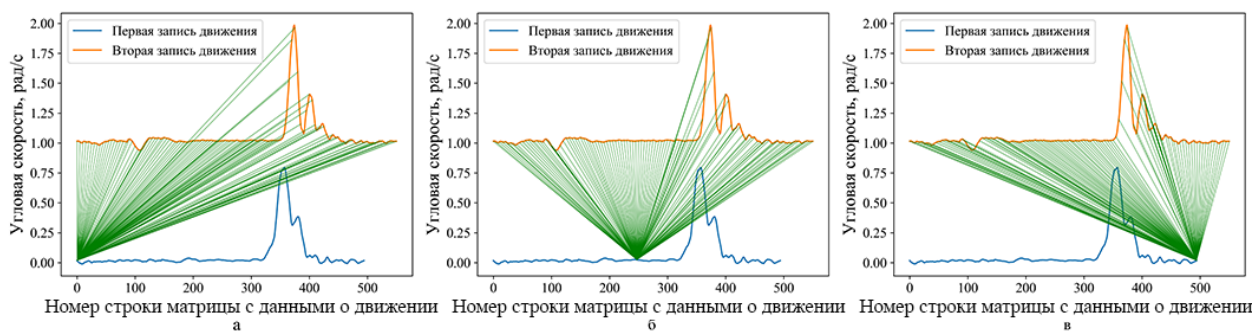


Рисунок 2.8 – Вычисление евклидова расстояния от: а) начального элемента первого паттерна до всех элементов второго паттерна; б) срединного элемента первого паттерна до всех элементов второго паттерна; в) последнего элемента первого паттерна до всех элементов второго паттерна.

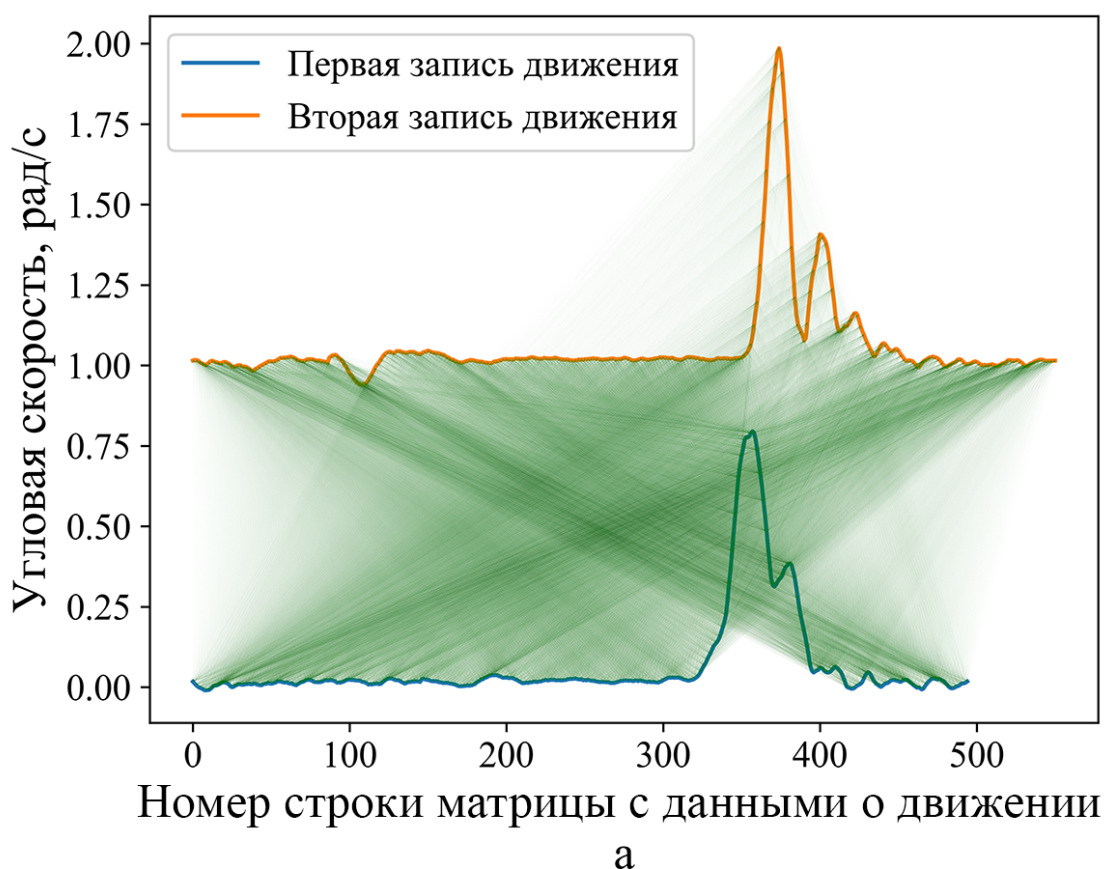


Рисунок 2.9 – Все взаимные расстояния, рассчитанные для двух паттернов.

ния работает независимо от остальных. В данной работе применяется идея сравнения цифровых паттернов движений между собой, поэтому количество сравниваемых столбцов матрицы движения прямо влияет на размерность пространства, в котором работает алгоритм DTW. Та же рассмот-

рены следующие методики понижения размерности данных: с помощью вейвлет-преобразования и биомеханической избыточности.

**Определение 2.** *Матрицей локальных расстояний* размерностью  $N \times M$  назовем такую матрицу, элементы которой  $[n, m]$  являются евклидовым расстоянием между элементами временных рядов  $x[n]$  и  $y[m]$ ,

Далее в матрице локальных расстояний необходимо найти наименьший путь<sup>6</sup> в смысле оригинального исследования алгоритма динамического программирования по выравниваю временных рядов В. Н. Juang 1984 года [110]. Выбор звеньев пути составляется согласно следующим условиям:

1. **Граничные условия.** Путь должен начинаться, заканчиваться в противоположных концах диагонали матрицы локальных расстояний  $d$ .

$$\phi[1] = (1, 1)$$

$$\phi[K] = (N, M)$$

$$\min(N, M) \leq K \leq N + M - 1 \quad (2.19)$$

где  $\phi$  – функция искажения временной шкалы,

$K \in \mathbb{N}$  – последний шаг пути,

$\phi[1] \in \mathbb{Q}_0^+$  – первый элемент пути,

$\phi[K] \in \mathbb{Q}_0^+$  – последний элемент пути.

2. **Непрерывность.** Продолжение пути возможно только через соседние клетки матрицы локальных расстояний.

$$\phi[k] = (n, m) \Rightarrow \phi[k + 1] = (n', m')$$

$$n' - n \leq 1$$

$$m' - m \leq 1$$

, где  $k \in \mathbb{N}$  – промежуточный номер шага пути,

---

<sup>6</sup>англ. оригинальный термин – *path*

$\phi[k] \in \mathbb{Q}_0^+$  –промежуточный элемент пути,  
 $n \in \mathbb{N}$  –промежуточный номер элемента временного ряда  $x$ ,  
 $m \in \mathbb{N}$  –промежуточный номер элемента временного ряда  $y$ ,  
 $n' \in \mathbb{N}$  –соседний с  $n$  номер элемента,  
 $m' \in \mathbb{N}$  –соседний с  $m$  номер элемента.

3. **Монотонность.** Продолжение пути назад по времени невозможно.

$$\phi[k] = (n, m) \Rightarrow \phi[k + 1] = (n', m'),$$

$$n' \geq n$$

$$m' \geq m$$

Для приведенного выше примера была построена матрица локальных расстояний с нанесенным поверх наименьшим путем на рис. 2.10.

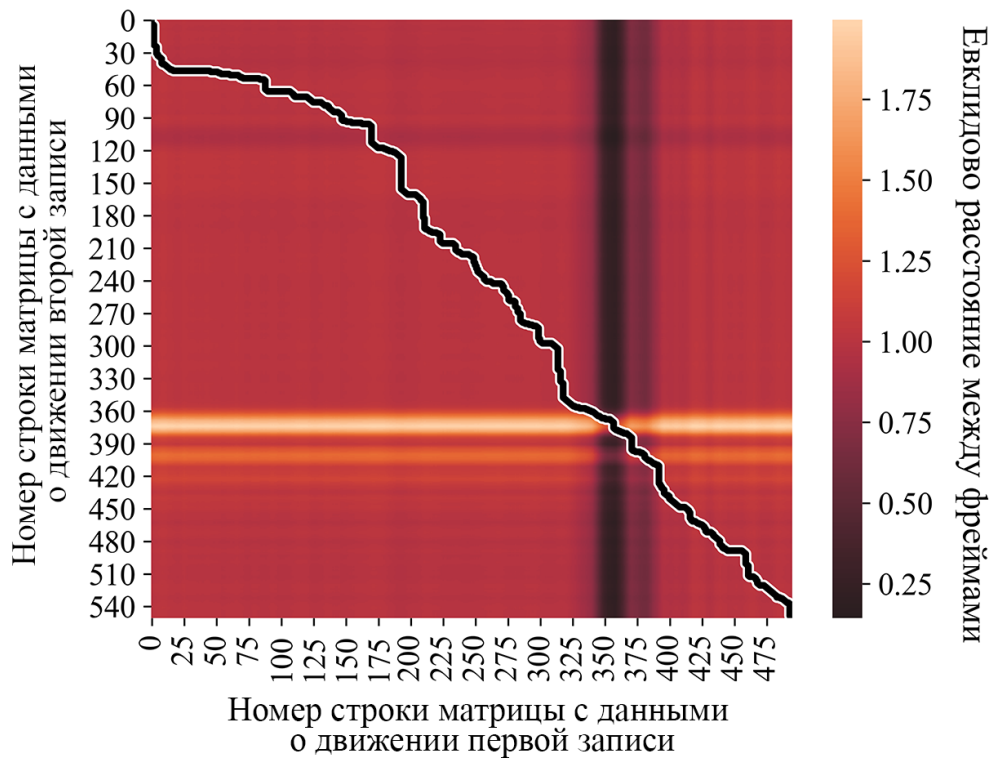


Рисунок 2.10 – Матрица локальных расстояний с наименьшим путем.

Минимальный путь демонстрирует наименьшую величину трансформации временной оси, которая требуется для того, чтобы один цифровой паттерн движения стал идентичен другому. Длина такого пути будет точно

равна 0, если цифровые паттерны совпадают. Верхнее значение не ограничено.

В реальных условиях работы системы возникновение идентичного цифрового паттерна маловероятно из-за шумовой компоненты MEMS и сенсорных корректировок в биомеханической модели поведения человека. Следовательно, дизайн эксперимента основывается на средних значениях расстояний DTW для каждого движения. Для этого выполняется серия записей одного и того же движения, после чего производится усреднение паттерна движения. Затем этот усредненный эталонный паттерн сопоставляется с каждой отдельной записью движения с использованием алгоритма DTW. Получаются диапазоны значений DTW-расстояний, в которых предполагается наличие заданного паттерна. Для проверки качества полученного эталона его следует сопоставить с другим эталоном при помощи того же DTW-алгоритма. Если значение расстояния выходит за пределы допустимого диапазона, можно утверждать о различимости эталонов цифровых паттернов движения.

Каждый элемент пути представляет собой пару чисел, соответствующих координатам в матрице локальных расстояний. В идеале полного совпадения двух цифровых паттернов маршрут будет представлять диагональ в квадратной матрице.

Значение элемента пути  $(1, 1)$  указывает на то, что первому элементу цифрового паттерна  $x$  соответствует первый элемент цифрового паттерна  $y$ .  $(1, 2)$  обозначает, что первому элементу паттерна  $x$  соответствует второй элемент паттерна  $y$ . Другими словами, горизонтальные и вертикальные перемещения по матрице локальных расстояний обозначают соответствие нескольких точек паттерна одной точке паттерна и наоборот. Присутствует отображение «один к многим».

Таким образом, мы полагаем: чем больше усилий надо приложить для искажения временной шкалы по преобразованию одного цифрового паттерна в другой, тем больше вероятность того, что это разные цифровые паттерны.

**Определение 3.** *Путь DTW – такой путь, который идет через области наименьших локальных расстояний в двумерном представлении матрицы локальных расстояний [111].*

$$\phi[k] = (\phi_x[k], \phi_y[k]) \quad (2.20)$$

где  $\phi$  – функция искажения временной шкалы,

$k \in N$  – промежуточный номер шага пути,

$\phi_x[k] \in \mathbb{Q}_0^+$  – проекция  $\phi[k]$  на выходной канал паттерна  $x$ ,

$\phi_y[k] \in \mathbb{Q}_0^+$  – проекция  $\phi[k]$  на выходной канал паттерна  $y$ ,

Функция  $\phi$  для вычисления минимального расстояния:

$$\phi(x, y) = \min \left( \frac{\sqrt{\sum_{k=1}^K (x(\phi_x[k])y(\phi_y[k]))^2}}{K} \right) \quad (2.21)$$

Согласно учебнику «Элементы теории функций и функционального анализа» А.Н. Колмогорова и С.В. Фомина [109] – метрическим пространством называется пара  $(X, \rho)$ , состоящая из некоторого множества (пространства)  $X$  элементов (точек) и расстояния, т.е. однозначной, неотрицательной, действительной функции  $\rho(x, y)$ , определенной для любых  $x, y$  из  $X$  и подчиненной аксиомам тождества, симметрии и треугольника.

**Определение 4.** *Метрическим пространством DTW назовем такое пространство в  $\mathbb{R}^2$ , которое состоит из множества элементов матрицы локальных расстояний и пути, который определен в формуле 2.21.*

**Утверждение 2.2.1.** *Пусть определены матрица локальных расстояний и функция для вычисления расстояния на данном множестве элементов однозначная, неотрицательная, действительная и удовлетворяющая аксиомам тождества, симметрии и треугольника. Тогда определено метрическое пространство DTW, которое обозначим  $\Omega$ .*

*Доказательство.* Проверим однозначность функции 2.21. Неопределенность выбора следующего шага снимается тремя ограничениями на построение пути и выбором паттерна шага. Возможные варианты приведены на рис. 2.11. Самыми популярными являются symmetric1 и symmetric2 [112].

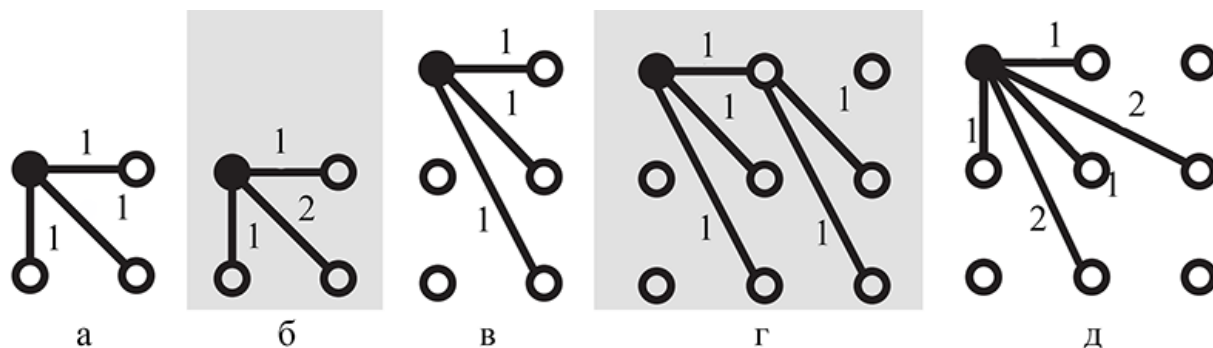


Рисунок 2.11 – Паттерны шага для алгоритма DTW: а) symmetric1; б) symmetric2; в) asymmetric; г) rabinerJuang; д) symmetric5.

Нейтральный путь искажения временной шкалы — это прямая линия с наклоном  $45^\circ$  через матрицу локальных расстояний. Если временной ряд  $x$  короче, то наклон минимального пути искажения времени будет больше  $45^\circ$ , а если он длиннее, то наклон минимального пути искажения времени будет меньше  $45^\circ$ . Следующие параметры можно менять для управления паттернами шага.

1. Явно указать минимальный и максимальный наклон пути.
2. Явно указать максимальную полосу для пути искажения времени.
3. Предпочитать путь из более коротких, либо более длинных шагов.

Проверим неотрицательность. Справедливость неотрицательности очевидна из того, что числитель и знаменатель неотрицательны. Числитель неотрицательный, так как состоит из квадрата разности, а знаменатель находится в указанном диапазоне 2.19.

Проверим действительность. Справедливость очевидна из накладываемых условий на введенные обозначения в данной главе.

Проверим три необходимых аксиомы для доказательства. Между всеми обобщенными цифровыми паттернами была произведена оценка степени схожести с помощью алгоритма DTW. Доказательство верности выраже-



ния 2.21 приведено в [109]. Формула отличается только константным действительным делителем, что не влияет на ход самого доказательства.

**1. Аксиома тождества.**

$$d(x, y) = 0 \Leftrightarrow x = y$$

Все диагональные значения равны 0. Таким образом, расстояние паттерна до самого себя равно 0, что подтверждает данную аксиому.

**2. Аксиома симметрии.**

$$d(x, y) = d(y, x)$$

Все симметричные значения относительно диагонали имеют равное значение, что подтверждает вторую аксиому.

**3. Аксиома треугольника.**

$$d(x, z) \leq d(x, y) + d(y, z)$$

Расстояние от  $x$  до  $z$  через  $y$  не меньше чем прямо от  $x$  до  $z$ , что тоже подтверждается приведенными тепловыми картами.

Необходимые свойства функции и три аксиомы подтверждаются.

□

Алгоритм DTW удобен для сравнения разных по продолжительности временных рядов, однако не учитывает по умолчанию допустимый уровень разброса значений цифровых паттернов движений. Что затрудняет классификацию.

Исходя из проведенного анализа, стандартные методы оказываются неспособными вычислить эталонный цифровой паттерн движения с учетом большой вариабельности движений. Одной из основных проблем является неспособность данных методов корректно оценивать уровень нормы отклонения значений сигнала в каждом элементе матрицы с данными о движении. Более того, данные методы также не способны компенсировать дрейф нулевого уровня сигнала.

Для решения данной проблемы предлагается использование новых подходов и алгоритмов, которые бы могли обеспечить более точное вычисление эталонного цифрового паттерна движения и учет большой вариабельности движений. Эти подходы должны быть способными корректно оценивать и контролировать уровень отклонения значений сигнала в каждом элементе матрицы данных о движении. Кроме того, разработанные алгоритмы должны быть способными компенсировать дрейф нулевого уровня сигнала.

Предложенные новые подходы и алгоритмы будут основываться на принципах системного анализа и математического моделирования. Они позволят учесть сложность движений, включая большую вариабельность, и обеспечить более точные результаты вычисления эталонного цифрового паттерна движения. Кроме того, эти подходы должны быть способными не просто компенсировать дрейф нулевого уровня сигнала для достижения более надежных результатов, а ещё и исключить регулярность во время самого вычисления.

### **Количество вычислительных операций**

Исходя из поставленной задачи, имеются два противоречивых требования: на одной стороне необходимо обеспечить максимально возможное количество реальных значений сигнала, получаемых в течение одной секунды времени. Это связано с желанием достичь высокой точности и детализации данных о движении.

Однако, на другой стороне нужно обеспечить минимально компактное представление этих данных при их сопоставлении с эталонными записями движения. Требуется найти баланс между сохранением информативности и экономичностью хранения данных для последующего анализа.

Для решения данного противоречия необходимо разработать новые методы кодирования и компрессии данных о движении. Они должны быть способными представлять значения сигнала таким образом, чтобы максимизировать использование доступного пространства памяти или передачи

данных без потерь важных деталей о движении.

Вычислительные требования:

У нас имеется входной набор данных, состоящий из 16 датчиков, каждый из которых обладает 6 выходными каналами. Длительность движений может достигать до 30 секунд, и система должна быть способна обрабатывать данные со скоростью порядка 200 значений в секунду. Эти требования являются приемлемыми как для тестовых цифровых паттернов движений, так и для эталонных записей.

Для создания эталонных записей необходимо минимум 8 различных паттернов движения. При использовании алгоритма сравнения сложность операций возрастает пропорционально квадрату числа данных:  $O(N^2)$ , где  $N$  представляет собой количество операций для каждого сравнения.

Таким образом, чтобы определить принадлежность исследуемого движения к одному из эталонных паттернов, потребуется выполнить определенное количество операций. Общая формула будет следующей:

Для анализа и сопоставления движений в системе рассматривается входной набор данных, содержащий информацию от  $M = 16$  датчиков, каждый из которых порождает  $C = 6$  выходных каналов. Продолжительность движения ограничивается временным интервалом  $T = 30$  секунд, а система способна обрабатывать данные с частотой приблизительно  $F = 200$  измерений в секунду.

Пусть  $N = 7$  представляет собой количество эталонных паттернов движения, используемых для создания эталонных записей. При использовании алгоритма сравнения, вычислительная сложность операций увеличивается пропорционально квадрату  $N$ , что обозначается как  $O(N^2)$ , где  $N$  представляет количество операций для каждого сравнения.

Таким образом, для определения принадлежности исследуемого движения к одному из эталонных паттернов, необходимо выполнить определенное количество операций. Общее выражение для вычисления общего числа сравнений имеет вид:

$$KVO = E \times (M \times C \times F \times T)^2 \quad (2.22)$$

$$KVO = 7 \times (16 \times 6 \times 200 \times 30)^2 = 2\,322\,432\,000\,000$$

Где  $KVO$  - количество вычислительных операций,  $E$  - количество подготовленных эталонных паттернов движений,  $M$  представляет количество датчиков,  $C$  - количество каналов,  $F$  - частоту измерений,  $T$  - продолжительность движения, а  $N$  - количество эталонных паттернов движения. Такая формула отражает общее количество операций, необходимых для проведения всех сравнений между данными движения и эталонами. Ввиду такой вычислительной сложности, важно разрабатывать эффективные алгоритмы и методы для оптимизации процесса сопоставления движений и повышения эффективности работы системы.

Полученная формула отражает количество операций необходимых для выполнения всех сравнений между данными о движении и эталонными записями. Учитывая данную сложность, требуется разработать эффективные алгоритмы и методы для оптимизации процесса сопоставления движений и ускорения работы системы.

Влияние возможно на параметры  $M$ ,  $C$ ,  $F$ . Для этого нужно определенным образом структурировать сигналы для формирования цифрового паттерна движений в виде матрицы для последующей трансформации.

При снижении входящего количества информации в 4 раза, количество вычислительных операций будет снижаться на 93,75 процентов. Если снизить в 16 раз, то теоретически должна снизиться на 99,6 процентов. Помимо снижения количества данных, надо удостовериться, что нет потерь значимой информации о движении.

### **Формирование матрицы, включающей выходные сигналы**

Все движения кисти человека можно описать как поворотами суставов [113, 114], так и созданием уникального и компактного описания движения, на их основе [115]. Будем это использовать в реализации комплекса контроля движений, так как МЭМС оцифровывают именно угловые параметры, на основе которых можно создать уникальную сигнатуру движения.

## Уровень корреляции столбцов матрицы с данными о движении

Далее будет происходить работа с уже полученными матрицами с данными о движении. На данном этапе размерность составляет 96 столбцов (6 значений каждого сенсора умножить на количество сенсоров 16). Сравнить в таком виде матрица - трудоёмкая задача, так как небольшие движения могут доходить по времени от 1 секунды и больше. В таком случае размерность матрица с данными о движении составит не менее  $96 \times 215$ . Чтобы убедиться, что это излишнее и избыточное описание движения - проведем корреляционный анализ, который продемонстрирует, что некоторые столбцы являются линейной комбинацией других столбцов. А это значит, что их можно убрать без потери информативности описания сформированных классов движений.

Для этого в тестовой выборке будут оценены сильные линейные связи с помощью нахождения корреляции столбцов матрицы движения между собой. Далее из наборов данных будут удалены дублирующие столбцы матрицы движения. Для удаления линейной избыточности были отсечены все связи, которые нельзя было отнести в сильным (со значением корреляции более 0,9 по модулю).

Применение корреляционного анализа, показывает, что число избыточных столбцов матрицы движения для наборов данных (НД) равняется:

- 41 для НД, зарегистрированный при движении всей кисти во время проведения прямой линии при помощи карандаша;
- 26 для НД, зарегистрированный при движении указательного пальца при движении в горизонтальной плоскости;
- 28 для НД, зарегистрированный при сгибе безымянного пальца;
- 72 для НД, зарегистрированный при движении ладонью от себя;
- 70 для НД, зарегистрированный при движении ладонью влево;
- 65 для НД, зарегистрированный при движении ладонью на себя;
- 71 для НД, зарегистрированный при движении ладонью вправо.

Этот список дает весомые основания полагать, что существует некоторое число столбцов матрицы движения, которые можно будет исключить из

анализа.

Хотя начальные НД имеют строгие границы появления значений (от  $-32768$  до  $32767$ , или двухбайтовое число типа `int`), однако наиболее вероятно встретить разные распределения значений для гироскопа и акселерометра — по их физической и биомеханической природе. Так, по собранной статистике максимальные значения, которые были зафиксированы во всем НД, для гироскопа и акселерометра составляют  $345,6$  °/сек (моментальная угловая скорость) и  $2,5$  g соответственно. Учитывая что рабочий диапазон составляет  $\pm 500$  °/сек и  $\pm 8$ g - в первом грубом приближении уже можно утверждать, что гироскоп не задействует более 30% своего диапазона, а акселерометр - более 68%. Зная эту информацию можно изначально настроить работу сенсоров на выявленные диапазоны работы. Из этого также вытекает предположение, что разное распределение данных приведет и к разному уровню разброса данных, — а это, в свою очередь, повлияет на корреляционный анализ. Для проверки предположения контрольная и тестовая выборка была дополнительно поделена по ненормированным значениям (рис. 2.13) и нормированным (рис. 2.14). Нормировка (биекция) проводилась в диапазоне  $\pm 1$ . Для иллюстраций применяется один и тот же НД, обработанный разными способами.

Предположение о наличии сильно коррелирующих столбцов матрицы движения подтверждается матрицей взаимных корреляций столбцов, где присутствуют явные линейные связи (рис. 2.12). Взаимозависимость некоторых столбцов видно как на рис. 2.13, так и рис. 2.14.

Теоретически это объясняется биомеханической связностью кисти человека, помимо костных взаимосвязей, присутствуют мышечные и сухожильные. Движение некоторыми мышцами провоцирует движение связанных мышц по определенным закономерностям, которые в итоге выливаются во взаимосвязанные показания датчиков.

В матрице с данными о движении это выражается в виде столбцов, которые являются линейной комбинацией других столбцов. В общем виде это нельзя заранее определить, необходимо проводить анализ под конкретно

заданный набор движений.

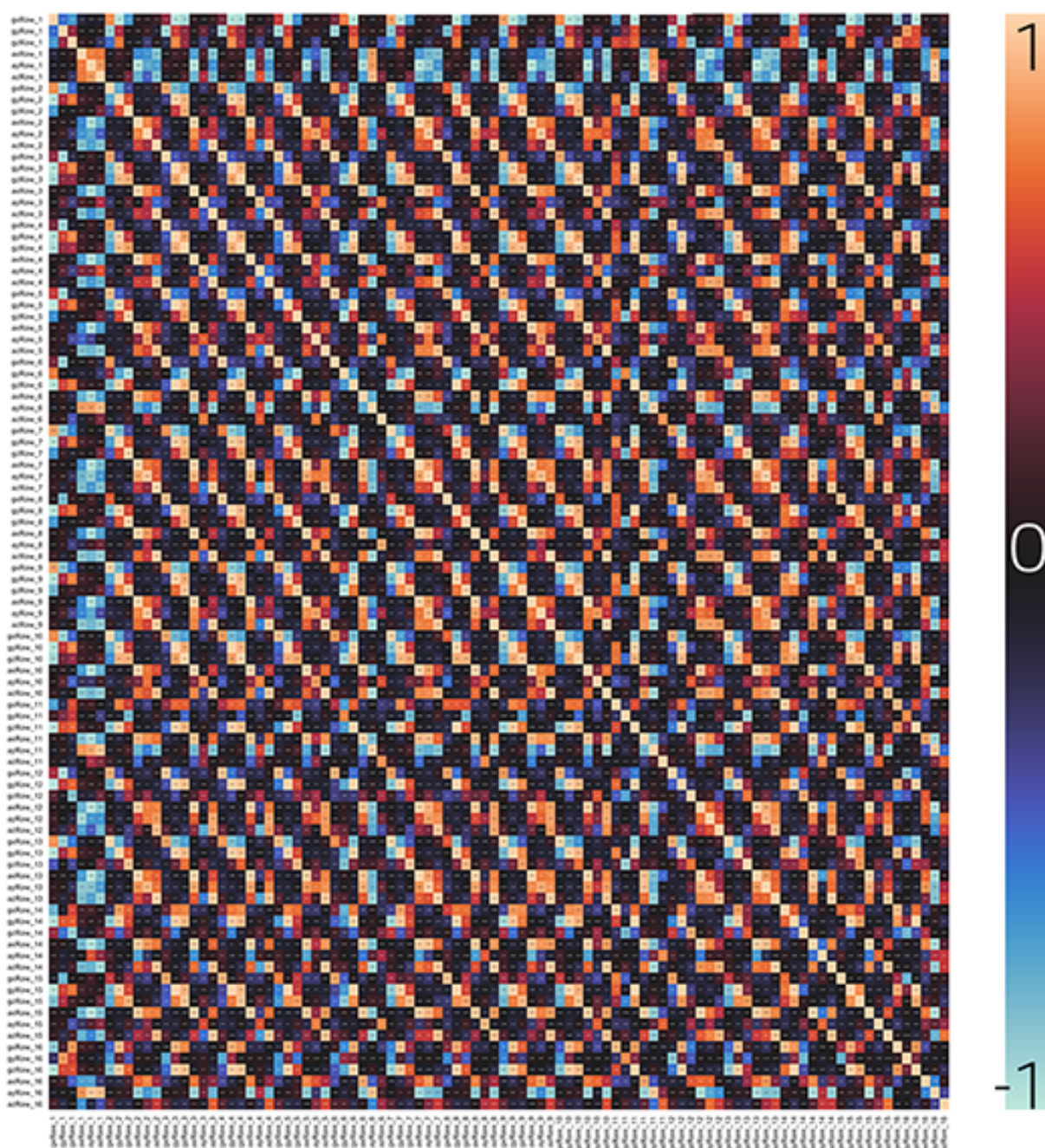


Рисунок 2.12 – а) Матрица взаимных корреляций НД, зарегистрированного при движении ладонью вперед; б) матрица взаимных корреляций для НД, зарегистрированного при движении ладонью вправо.

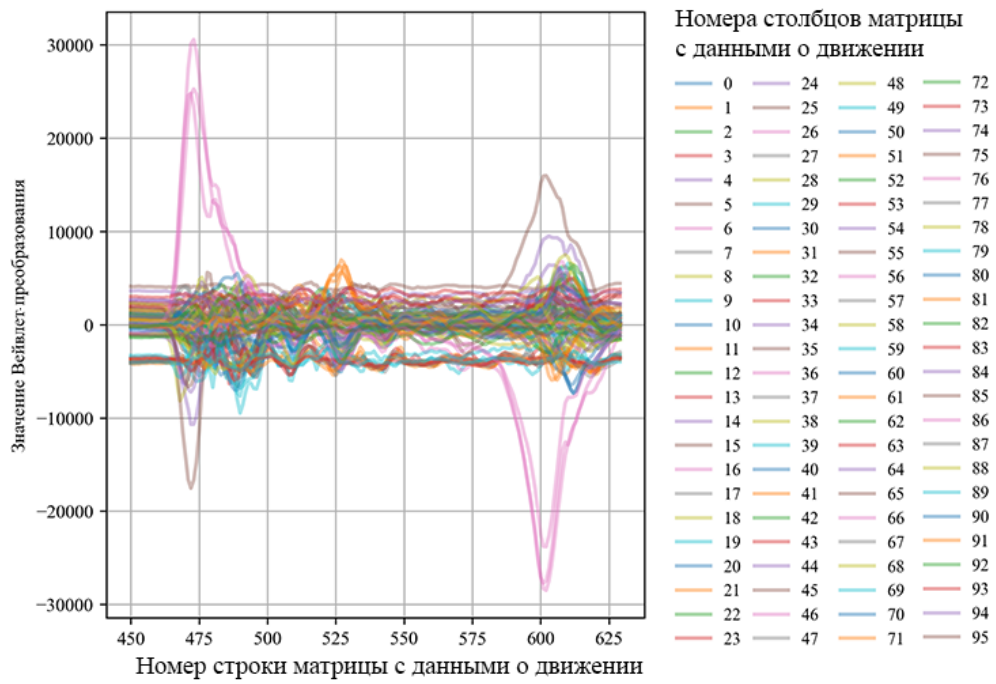


Рисунок 2.13 – НД, зарегистрированный при движении указательного пальца при движении в горизонтальной плоскости.

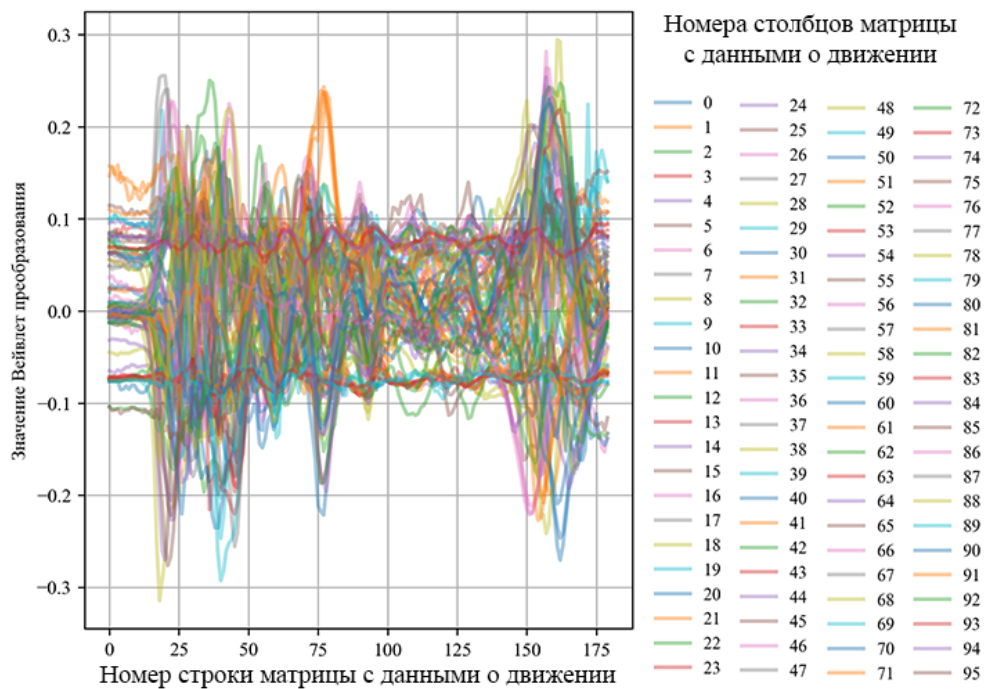


Рисунок 2.14 – НД, зарегистрированный при движении указательного пальца при движении в горизонтальной плоскости. Начальные значения с нормированными значениями в диапазоне  $\pm 1$ .



Фактические данные указывают на то, что мы можем исключить часть информации без потери общей информативности цифрового паттерна движения.

### Учёт дрейфа нулевого уровня выходных каналов датчиков при помощи метаданных эталонного паттерна движений

Для каждого набора данных сформирован обобщенный цифровой паттерн, описывающий конкретное движение. Для каждого из 7 типов действий было записано по 8 вариаций, которые сохраняют крайние состояния выполнения движений по скоростям и амплитудам.

Из этих вариаций эталонного выполнения действия можно сформировать одну эталонную матрицу по представленному алгоритму. В каждом значении выходного уровня сигнала каждой вариации движения на гироскоп действует дрейф нулевого уровня с  $\pm 3^\circ/\text{сек}$  и на акселерометр  $\pm 40\text{mg}$ .

Существующие сенсорные коррекции движения (по Бернштейну Н.А.) имеют больший уровень разрешенной зоны изменений.

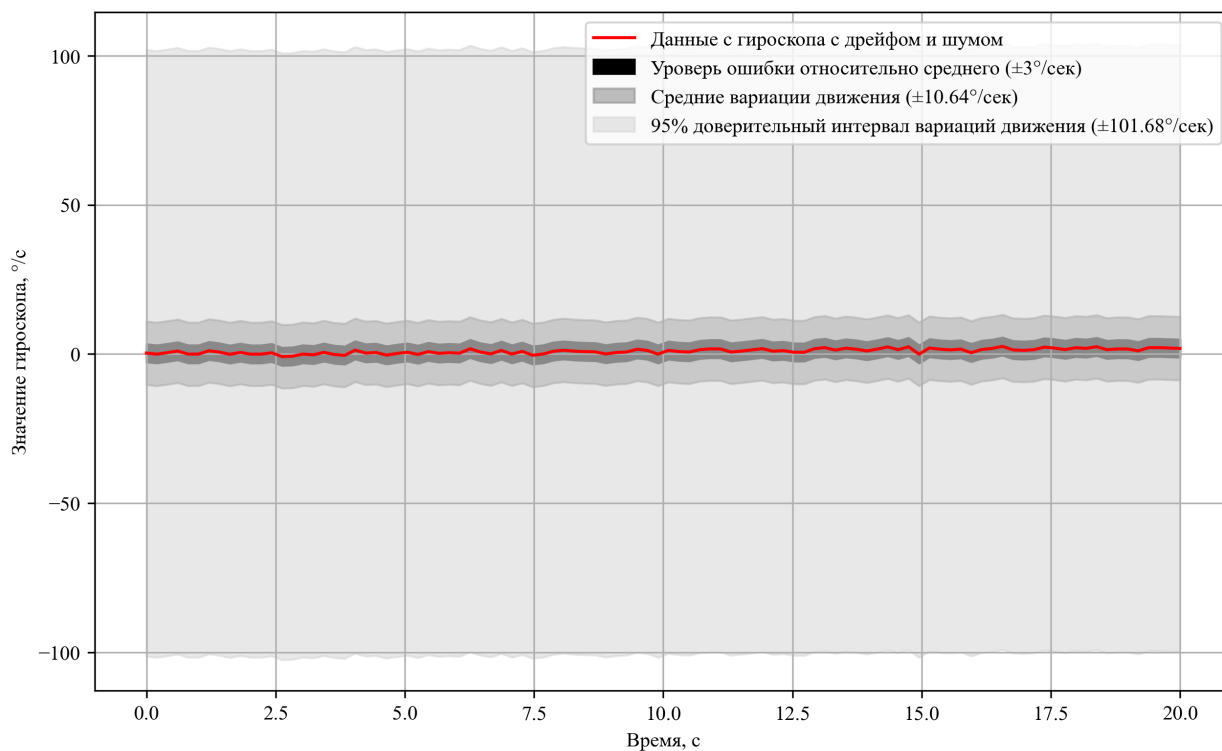


Рисунок 2.15 – График дрейфа нулевого значения гироскопа вместе с доверительными интервалами нормальных вариаций значений датчика

Для гироскопа среднее значение вариации для точки является  $\pm 10.64$ , максимальное  $\pm 101.68$ . Для акселерометра среднее значение вариации:  $\pm 140\text{mg}$ , а максимальное -  $\pm 386\text{mg}$ .

Таким образом строя специальные доверительные интервалы (95% по генеральной совокупности) около среднего значения - можно выстраивать в более чем 3 раза более широкие коридоры значений, которые будут уже по своему дизайну учитывать типичные для датчиков ошибки. Такая разрешенная зона позволяет доводить соотношение сигнал/шум около каждой точки от 11 до 30 Дб (см. Рис. 2.15).

В итоге получается, что имея средний уровень значений нескольких эталонных вариаций действий - можно выстраивать широкую разрешенную зону отклонений. Такая зона перекрывает существующие постоянные ошибки и позволяет убрать из фокуса вниманий дрейф нулевого уровня выходных сигналов датчиков. Уровень ошибок при этом не будет превышать 5 процентов для каждого типа движений.

Для каждого класса движений характерен свой уникальный уровень среднего значения в каждой точке матрицы данных о движении, и эти значения не коррелируют между собой на статистически значимом уровне (коэффициент корреляции менее 0,7) при анализе различных движений. Это обусловлено разной последовательностью техники действий при выполнении этих движений.

В результате получается, что на входе мы осуществляем отбор данных с минимальным уровнем шума и максимальным количеством релевантной информации о движении. Эти данные, хотя и содержат избыточность, представляют дополнительную ценность для эталонного цифрового паттерна движения. Они обладают широкой зоной изменчивости движения по сравнению с нулевым уровнем дрейфа, что позволяет его учитывать единожды на эта формирования цифровых паттернов движений.

Таким образом, можно говорить о разработке специального метода для сокращения размерности матрицы данных о движении путем трансформации как столбцов, так и строк. Это позволит достичь более компактного

и информативного представления матрицы данных о движении за счет уменьшения излишнего описания без потери информационной ценности.

## Общая схема

Общая схема трансформации данных до сравнения цифровых паттернов представлена на Рис.2.16.



Рисунок 2.16 – Способы формирования цифровых паттернов движения: эталонные и исследуемый

Для записи эталонных движений проводится предварительный анализ, общая схема которого выглядит следующим образом.

1. Выделение целевых движений вместе с экспертом.
2. Установка полного набора датчиков.
3. Выполнение всех вариаций целевых действий.
4. Получение матриц с данными.
5. Формирование такой эталонной матрицы с данными о движении, которая будет учитывать, но не будет фокусироваться на компенсации дрейфа нулевого уровня выходных сигналов.

## 2.3 Выводы по главе

1. Описаны основные характеристики сигнала, полученного из комплекса контроля движений, с акцентом на особенности первичной информации с акселерометра и гироскопа.
2. Выделены проблемы, возникающие при стандартных методах обработки сигналов, подчеркнута необходимость в увеличении количества датчиков для точной интерпретации движения.
3. Проведено измерение степени схожести двух цифровых паттернов движений, основанное на статистических методах, что позволило уточнить оценку корреляции между данными о движении и выходными сигналами.
4. Разработано алгоритмическое обеспечение для анализа сигналов, включая формирование матрицы, которая содержит выходные сигналы. Данный подход обеспечивает учет дрейфа нулевого уровня выходных каналов датчиков.
5. Подробно рассмотрена необходимая общая архитектура программно-аппаратного комплекса контроля движений и принципы его работы.
6. Акцентируется внимание на ошибке измерений, возникающей из-за погрешности сигнала и влияющей на конечное качество интерпретации движения. Обсуждена методика оценки этой погрешности и ее влияния на исследование.

## Глава 3

# Методики уменьшения размерности первичной информации, собираемых с инерционных датчиков

В начале данной главы представляется информация о формировании набора данных (НД), который подвергается всем последующим обработкам.

Далее исследуется методика преобразования матрицы с данными о движении, основанная на вейвлет-преобразовании, но не являющаяся его полным аналогом. Описывается метод правильной сортировки столбцов в матрице данных о движении. Обосновывается выбор масштаба вейвлет-преобразования и типа применяемого вейвлета.

Также представляется методика исключения столбцов матрицы с данными о движении, что позволяет уменьшить уровень избыточности в описании движения. В результате получаются компактные матрицы с данными о движении, которые могут быть сравнены на автономном вычислительном устройстве.

### 3.1 Формирование набора данных о движении кисти

#### План сбора данных

В качестве демонстрации работы выбраны следующие движения:

- Обездвиженное состояние перчатки.
- Произвольные движения кистью руки.

- Движение только лучезапястным суставом.
- Жесты.
  - Сгибание безымянного пальца.
  - Отведение в сторону указательного пальца в горизонтальной плоскости.
  - Провести прямую линию на листе бумаги, взяв предварительно карандаш.

Следует отдельно оцифровать отобранные движения, выполняя 8 записей для каждого указанного пункта.

Состояние перчатки в покое необходимо для учета специфики работы сенсоров. В данном случае, все столбцы матрицы движения должны быть независимыми, поскольку между ними отсутствуют как механические, так и биомеханические связи. Это состояние эквивалентно независимому расположению датчиков. Данное предположение подверглось последующей проверке.

Случайные движения кисти позволяют оценить общие взаимосвязи, характерные для всех биомеханических DoF.

Движения, выполняемые исключительно лучезапястным суставом, дают возможность рассмотреть специфический случай, в котором комплекс идеально описывается одним датчиком. Подобные решения применяются некоторыми исследовательскими группами для управления либо отдельным беспилотным летательным аппаратом<sup>7</sup>, либо группой беспилотников<sup>8</sup>.

Сгибание безымянного пальца должно показать связи между пальцами, обусловленные общим расположением сухожилий. При этом следует избегать отвода проксимальной фаланги пальца, так как в этом случае возможно согнуть остальные фаланги без участия других пальцев.

Отведение указательного пальца должно демонстрировать взаимосвязи пальцев с другой стороны ладони. Этот палец менее связан с остальными, поэтому выбрано наиболее иллюстративное движение, которое непременно

---

<sup>7</sup>Нолан Мур использует контроллер-перчатку Nintendo Power Glove <https://nplus1.ru/news/2016/06/07/power-glove>

<sup>8</sup>Дмитрий Тетерюков <https://events.yandex.ru/lib/talks/6263/>

вызовет движения и в других пальцах.

Вариант с проведением прямой линии на листе бумаги должен показать, что в матрице движения присутствуют неиспользуемые столбцы даже при выполнении сложных действиях.

В качестве факторов эксперимента по исследованию режимов работы комплекса могут выступать такие параметры:

- Количество используемых столбцов матрицы движения.
- Расположение датчиков.
- Временное окно поиска цифрового паттерна.
- Частота работы акселерометра.
- Частота работы гироскопа.
- Автокомпенсация дрейфа нуля.
- Интервал работы акселерометра.
- Интервал работы гироскопа.

Несмотря на отдельную значимость первых двух компонентов данного исследования, их взаимосвязь имеет немалую важность. В представленном материале нам предстоит оптимизировать количество используемых столбцов матрицы движения данных, но при этом обеспечить сохранение качества выводов при выполнении определённого набора задач. Однако, важно отметить, что при сокращении числа столбцов матрицы движения, следует осознанно подходить к вопросу оставления наиболее значимых и исключения менее важных для финального анализа.

В отношении временного окна поиска цифрового паттерна движения, наиболее эффективным подходом является применение специализированных алгоритмов, общепринятых в данной дисциплине. К примеру, метод Прони [116] или Mueen–Keogh [117]. Следует отметить, что возможно и определение временного окна эмпирическим путем, путём прямого измерения каждого цифрового паттерна на используемом оборудовании. Исходя из этого, можно сделать вывод о возможности исключения временного окна поиска как отдельного параметра.

Схожую ситуацию можно наблюдать и в отношении параметра частоты

ты работы гироскопа и акселерометра. С учётом того, что временная дискретизация составляет 400 Гц при рабочей частоте комплекса 212,38 Гц, мы обладаем адекватной частотой обновления показаний датчиков. При снижении частоты дискретизации ниже частоты работы всего комплекса, мы столкнемся с эффектами дублирования данных. Дополнительные подробности можно изучить в Табл. 3.1, где приведены значения отклонений. Схожую ситуацию можно наблюдать и для гироскопа.

Таблица 3.1 – Уровень погрешности акселерометра в зависимости от частоты дискретизации

Частота работы, Гц	25	50	100	200	400	800	1600
Уровень погрешности, милли g	0,6	0,7	1,0	1,5	2,2	2,8	4,3

## 3.2 Методика трансформации данных о движении с использованием вейвлет-преобразования

Представим столбцы матрицы движения в форме временного ряда. Для заданного сигнала мы можем применить дискретное вейвлет-преобразование. На первом этапе сигнал проходит через низкочастотный фильтр, характеризующийся импульсной реакцией. В результате мы получаем следующую свертку:

$$y[n] = (x * g)[n] = \sum_{k=-\infty}^{\infty} x[k]g[n - k] \quad (3.1)$$

где  $x$  – заданный сигнал.

$y$  – свертка сигнала.

$n$  – число уровней разложения.

$g$  – импульсный отклик.

Так же сигнал раскладывается при помощи высокочастотного фильтра. Получается два набора коэффициентов: аппроксимации и детализации. Де-



тализирующие коэффициенты получаются после высокочастотного фильтра. Коэффициенты аппроксимации после низкочастотного фильтра.

$$y_{\text{low}}[n] = \sum_{k=-\infty}^{\infty} x[k]g[2n - k] \quad (3.2)$$

$$y_{\text{high}}[n] = \sum_{k=-\infty}^{\infty} x[k]h[2n - k] \quad (3.3)$$

где,  $h$  - высокочастотный фильтр.

### Комбинация столбцов матрицы движения

Последовательность расположения столбцов в матрице движения имеет значительное влияние на итоговое вейвлет-преобразование. Рассмотрим два примера: набор данных без предварительной сортировки и набор, где столбцы матрицы движения отсортированы в соответствии с направленностью имеющихся столбцов. То есть на начало ставится выходной канал  $gx$  первого столбца, затем  $gx$  второго и так далее до шестнадцатого одноименного. Далее следуют типы выходных каналов:  $gy$ ,  $gz$ ,  $ax$ ,  $ay$ ,  $az$ . Для сортировки используется исходный набор данных. Затем к обоим примерам применяется одинаковое вейвлет-преобразование. Оба результата [2] представлены на рис. 3.1.

Здесь (см. рис 3.1) можно увидеть как влияет порядок столбцов матрицы движения на конечный результат. В случае неотсортированных столбцов идет смешение разнотипных датчиков, что приводит к тому, что большинство графиков схожи друг с другом. Применение аппроксимирующих коэффициентов приводит к быстрому усреднению значений датчиков между собой в случае, если рядом находятся датчики разного типа.

В данном контексте предпочтительнее отсортировать столбцы матрицы движения в соответствии с соответствующими выходными каналами датчиков. Так, выходной канал  $X$  всех гироскопов должен идти первой

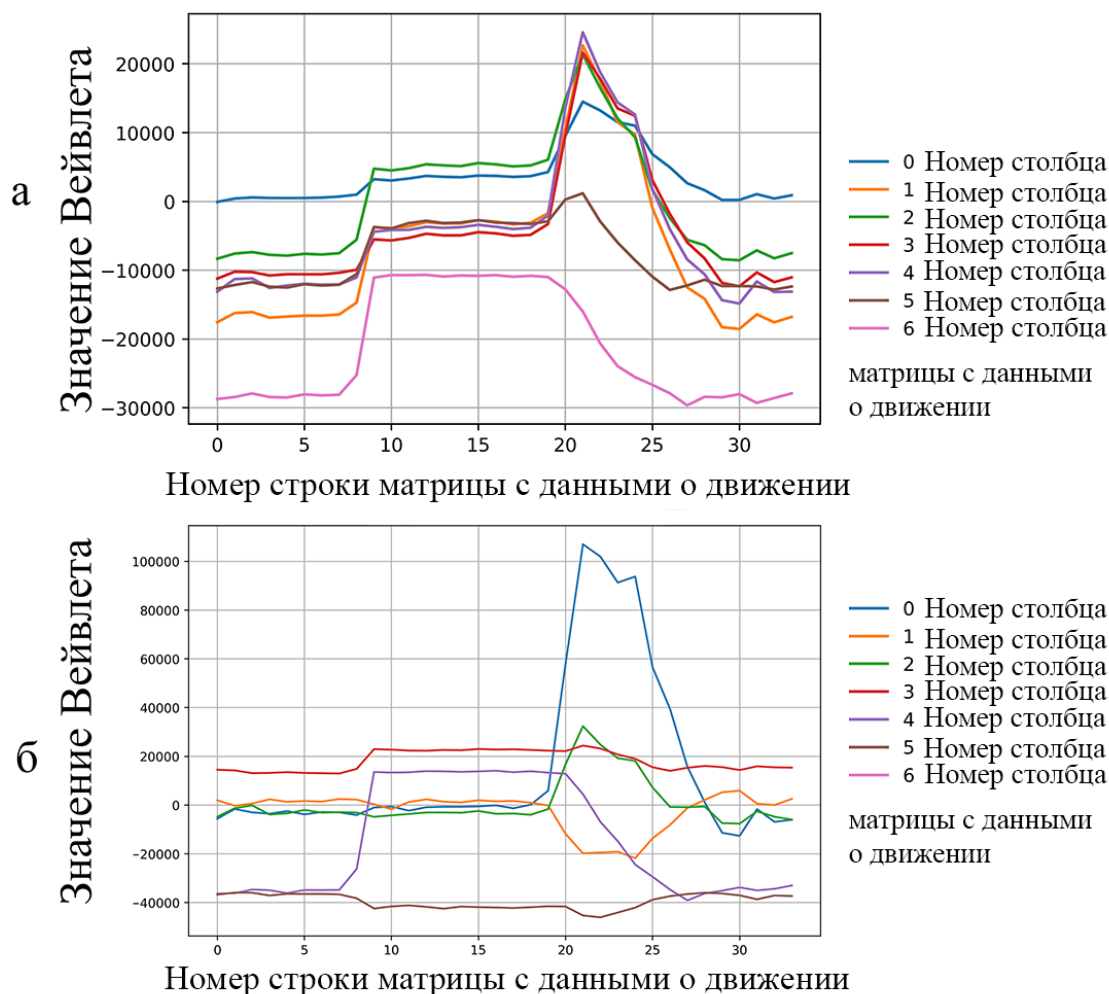


Рисунок 3.1 – Результаты вейвлет-преобразования с (а) неотсортированными столбцами матрицы движения и (б) отсортированными.

группой, выходной канал  $Y$  всех гироскопов - второй группой и т.д.

Значительный эффект от сортировки можно наблюдать при сжатии данных с использованием  $\text{tar}^9$ . Это связано с тем, что в наборе данных при раз-ной организации упорядочивания встречаются значения, либо с близкими значениями (значения датчиков обычно не имеют большого разброса из-за биомеханических ограничений на угловую скорость и ускорения суставов), либо с различными. В общем, можно утверждать, что данные, предвари-тельно отсортированные по содержимому, более подходят для сжатия лю-бым алгоритмом.

<sup>9</sup>Почему  $\text{tar.xz}$ -файлы, созданные с Python  $\text{tar}$ , оказались в 15 раз меньше, чем у macOS  $\text{tar}$  <https://habr.com/ru/company/flant/blog/548428/>

## Выбор типа коэффициентов для вейвлет-преобразования

Необходимо выбрать тип используемых коэффициентов вейвлет-преобразования. Всего возможно применение четырех типов коэффициентов<sup>10</sup>: аппроксимирующий и три детализирующих. Детализирующие могут быть вертикальными, горизонтальными и вертикальными (См. рис. 3.2). Стоит отметить, что изображения используют цветовую палитру не в градациях серого, а с применением цветной схемы библиотеки `matplotlib - viridis`. Сделано это из эстетических соображений. Данная схема подходит для отображения изображений из пространства градаций серого в цветное пространство<sup>11</sup>.

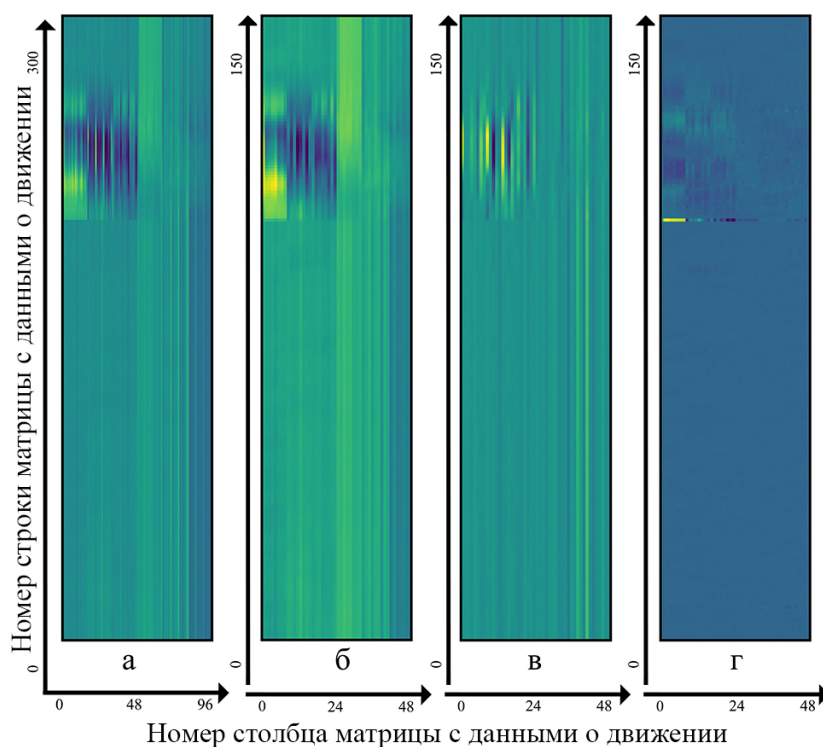


Рисунок 3.2 – Представление набора данных в виде изображения: (а) оригинальный НД, зарегистрированный при движении вперед рукой; (б) вейвлет-преобразование с применением одного уровня аппроксимирующего коэффициента; (в) вейвлет-преобразование с применением одного уровня вертикального коэффициента; (г) вейвлет-преобразование с применением одного уровня горизонтального коэффициента.

Центральная задача заключается в трансформации данных таким об-

<sup>10</sup>PyWavelets - Wavelet Transforms in Python <https://pywavelets.readthedocs.io/en/latest/>

<sup>11</sup>Choosing Colormaps in Matplotlib <https://matplotlib.org/3.2.1/tutorials/colors/colormaps.html>

разом, чтобы основная информация об изображении оставалась неповрежденной. Кроме того, возможно применение такой стратегии вейвлет-преобразования, при которой на каждом уровне используются различные типы коэффициентов преобразования. Хотя в данном контексте это не требуется, возможность существует. Для сохранения основной информации детализирующие коэффициенты не подойдут, поэтому наиболее подходящим является аппроксимирующий коэффициент, который мы будем применять итеративно несколько раз в зависимости от выбранного уровня вейвлета. Это подтверждается результатами анализа матрицы данных о движении с использованием вариации Аллана из предыдущей главы. Результаты итеративного вейвлет-преобразования с применением одного типа коэффициентов можно увидеть на рис. 3.3 для третьего уровня вейвлета и на рис. 3.4 для четвертого.

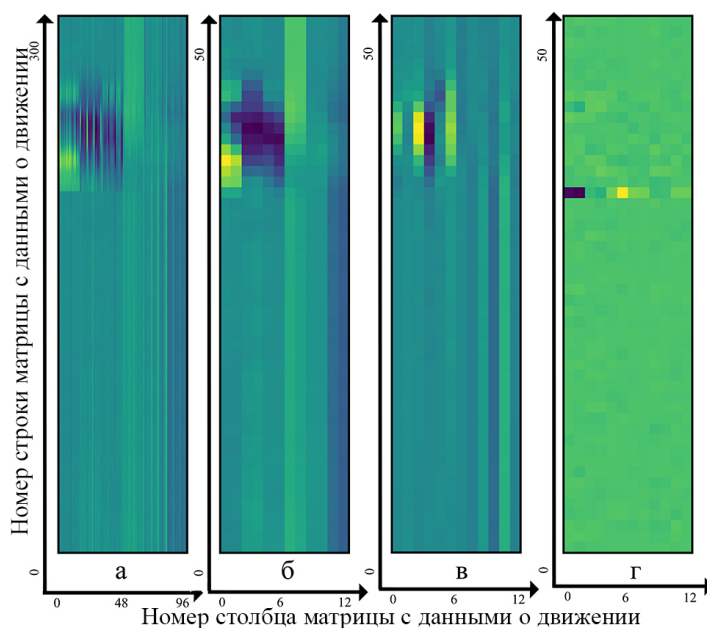


Рисунок 3.3 – Представление НД в виде изображения: (а) НД, зарегистрированный при движении вперед рукой; (б) вейвлет-преобразование с 3 уровнями аппроксимирующего коэффициента; (в) вейвлет-преобразование с 3 уровнями вертикального коэффициента; (г) вейвлет-преобразование с 3 уровнями горизонтального коэффициента.

Посмотрим 4 и 5 уровни масштаба при использовании только аппроксимирующих коэффициентов (рис. 3.5). На рисунке можно проверить выбор

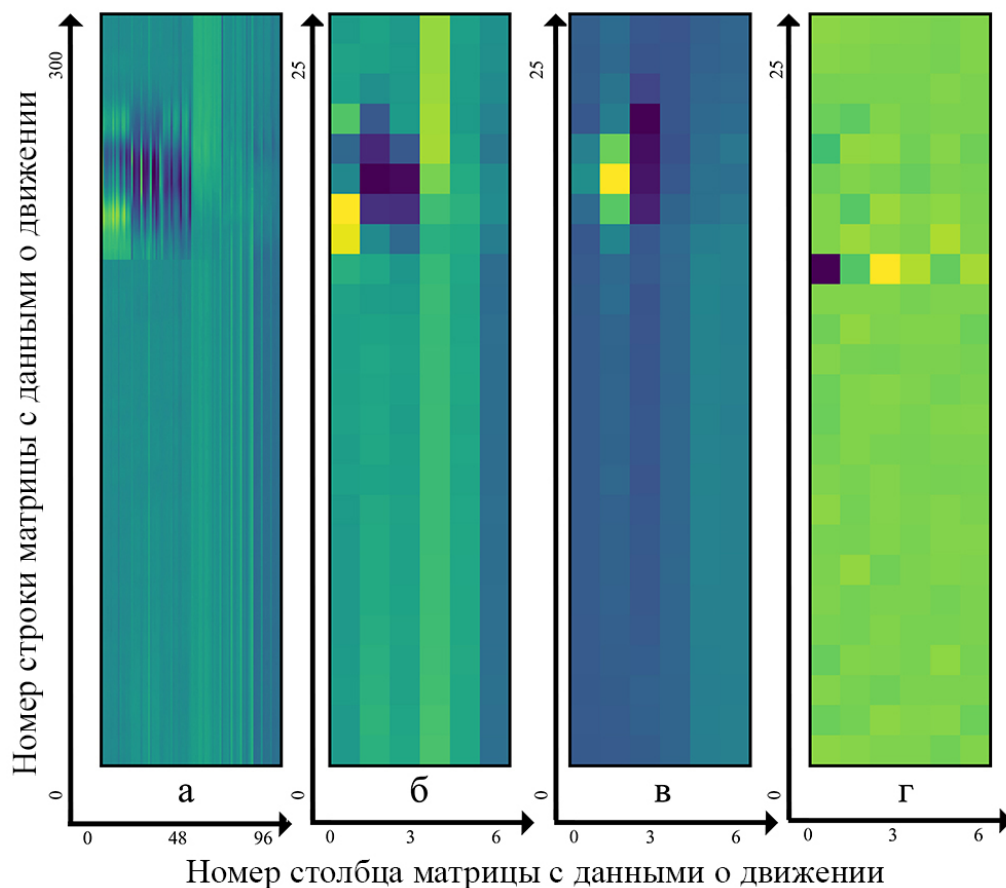


Рисунок 3.4 – Представление НД в виде изображения. (а) НД, зарегистрированный при движении вперед рукой. (б) вейвлет-преобразование с 4 уровнями аппроксимирующего коэффициента. (в) вейвлет-преобразование с 4 уровнями вертикального коэффициента. (г) вейвлет-преобразование с 4 уровнями горизонтального коэффициента.

масштаба на наборе данных в состоянии покоя.

Заранее известно, что в состоянии покоя значения для всех выходных каналов гироскопа будут колебаться около нуля, а для акселерометра будет около постоянного значения. В случае с 4 уровнем масштаба мы сохраняем данную информацию, а в более высоких уровнях уже нет. Значения гироскопа составляют половину от всех данных, значит на картинке должна быть горизонтальная площадка, занимающая половину площади графика.

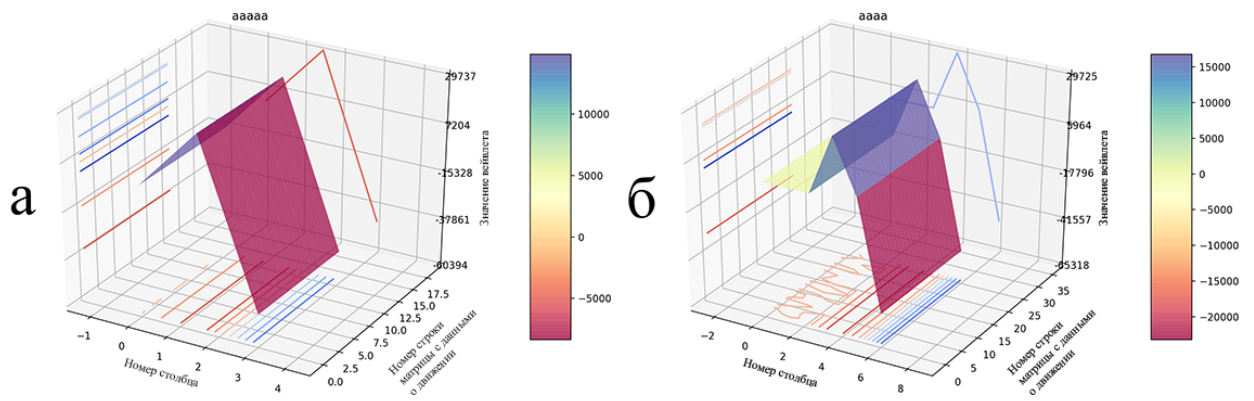


Рисунок 3.5 – Представление набора данных в виде поверхности. (а) Для масштаба 5 уровня. (б) Для масштаба 4 уровня.

## Выбор уровня вейвлет-преобразования и визуализация данных

**Первое ограничение** уровня вейвлет-преобразования вытекает из выражения  $\log_2 N$ , где  $N$  - размер исходного сигнала (96). Таким образом, для этого набора датчиков **уровень вейвлета не может превышать 6, 585**. Поскольку уровень вейвлета должен быть целым числом для дискретного вейвлет-преобразования, максимальный уровень вейвлет-преобразования составляет 6.

Далее, согласно принципу Найквиста, с каждым уровнем преобразования количество столбцов матрицы движения уменьшается вдвое. Так, имея на входе 96 столбцов матрицы движения, после первого уровня преобразования остается 48. На втором уровне - 24 столбца матрицы движения. В данном случае количество различных выходных каналов равно 6, и смешение столбцов разных семантических значений недопустимо. **Следовательно, верхней границей для этого набора данных будет 4 уровень вейвлета** (вейвлет-преобразование с выбранными коэффициентами можно применить четыре раза подряд) для обеспечения возможности полного восстановления исходного набора данных. Эта оценка не учитывает частоту самого движения и основывается только на критерии смешения колонок матрицы движения, имеющих одно направление.

В соответствии с правилом Найквиста, если исходный сигнал имеет са-

мую высокую частоту  $f_{max}$ , он требует выборки частоты  $f_s = 2f_{max}$ . Следовательно, на каждом уровне разложения и каждая информационная выходной канал рекурсивно разделяется пополам на идеальной частоте среза. Поскольку сигналы, полученные от МЭМС отбирались на частоте 212 Гц, диапазон действительных частотных составляющих сигнала находится между 0 и 106 Гц для первого уровня вейвлет-преобразования. В таких исследованиях основной энергетический диапазон для повседневной деятельности варьировался от 0,2 до 20 Гц, что позволяет по временной шкале **дважды применять вейвлет-преобразование** при сохранении возможности восстановить исходный сигнал. Объем данных в этом случае уменьшается в  $2^2$  раза по столбцам или в 16 раз с учетом строк (двумерное дискретное вейвлет-преобразование). Таким образом из изначальных 96 столбцов получается 24 столбца.

### Алгоритм принятия решений по уровню вейвлет-преобразования

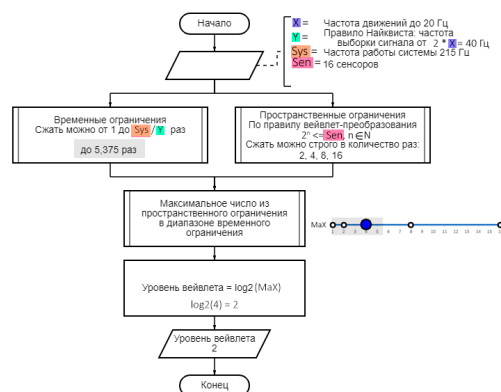


Рисунок 3.6 – Алгоритм принятия решений по уровню вейвлет-преобразования

С увеличением количества датчиков возникает проблема перегруженности визуализации сгруппированных цифровых паттернов. Попытка представить каждый столбец матрицы движения в форме временного графика затрудняет восприятие общей картины. Поэтому столбцы матрицы движения можно визуализировать в виде пиксельного изображения в градациях серого. Это означает, что у нас будет 96 пикселей, соответствующих столбцам матрицы движения, и каждый пиксель имеет уровень яркости от 0

до 65536. Вертикально представлены временные ряды выходных каналов датчиков (рис. 3.7).

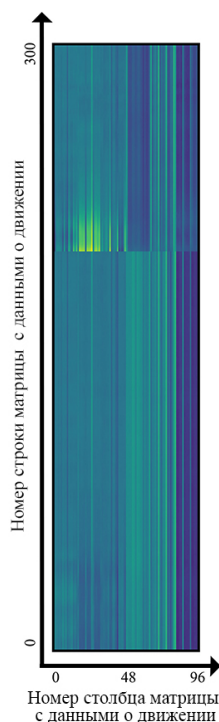


Рисунок 3.7 – Представление движения в виде изображения.

### **Анализ особенностей полученных данных после вейвлет-преобразования**

В течение 6-ти часов данные перчатки фиксировались в статическом положении. Затем была вычислена матрица корреляции каждого столбца матрицы движения с каждым другим, которая демонстрировала полное отсутствие связей между столбцами (см. источник [6]). По вертикали и горизонтали расположены номера столбцов матрицы движения. Из-за небольшого масштаба каждой ячейки, отмечены только крайние и центральные значения ячеек. Справа представлена тепловая шкала, соответствующая степени корреляции данных. Как видно, значимая зависимость присутствует только у столбца матрицы движения самого с собой.

В качестве иллюстрации матрицы взаимных корреляций для реального движения, рассмотрим набор данных, характеризующий движение ладони руки как плоскости с наклоном в сторону (рис. 3.8).



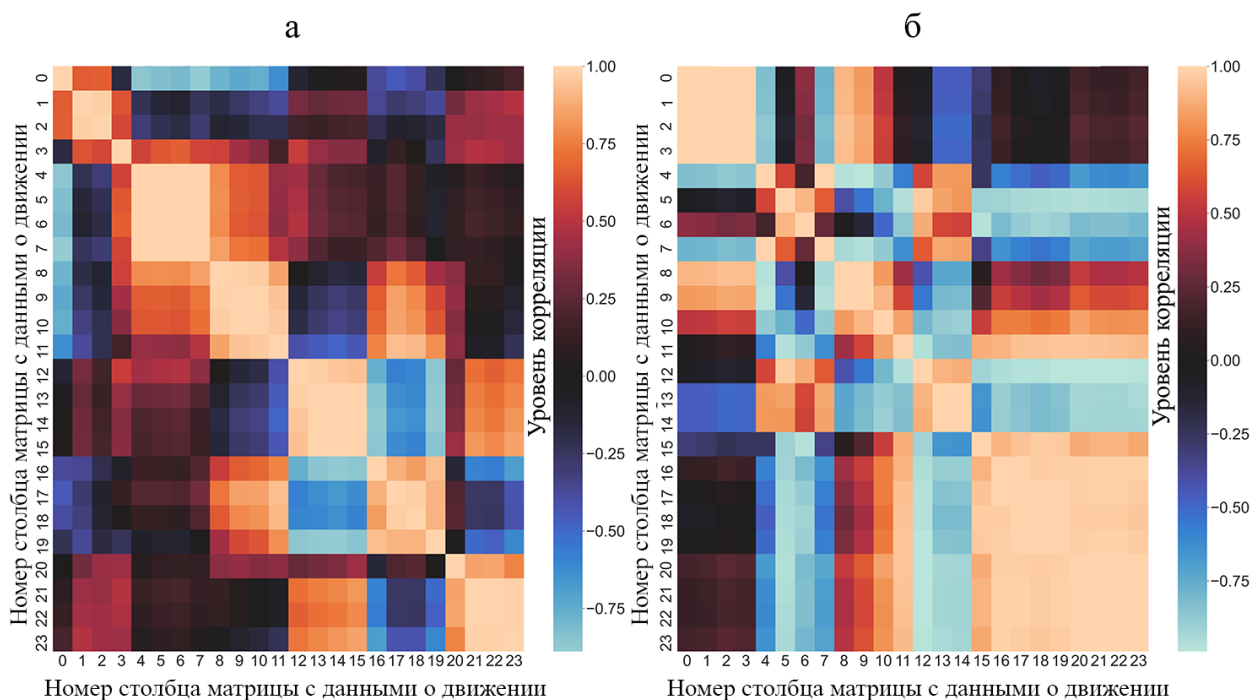


Рисунок 3.8 – Матрица взаимных корреляций между столбцами.

Ситуация изменяется, когда для аналогичного представления выбирается реальное движение. Итоговое представление взаимных корреляций столбцов матрицы движения данных можно увидеть на рис. 3.9 для всех доступных наборов данных.

Было отдельно проверено предположение о том, что столбцы матрицы движения для акселерометра и гироскопа не связаны линейно даже для наборов данных во время движения. Для этого было проведено попарное сравнение всех значений акселерометров с данными от гироскопов для каждого записанного набора данных. Для 61 набора данных только в 4 случаях (см. рис. 3.10) была обнаружена корреляция выше 0,8 (общее количество зависимостей - 10 из возможных 15616). И эти случаи были для направления  $Z$  акселерометра. Известно, что выходной канал  $Z$  у акселерометра дает большие погрешности, чем другие из-за конструктивных особенностей. Таким образом, можно утверждать, что линейная связь между данными акселерометров и гироскопов отсутствует.

Полученный датасет не позволял провести вычисления в оперативной памяти. Поэтому данные были разбиты на 384 блока по 10 000. Для всех

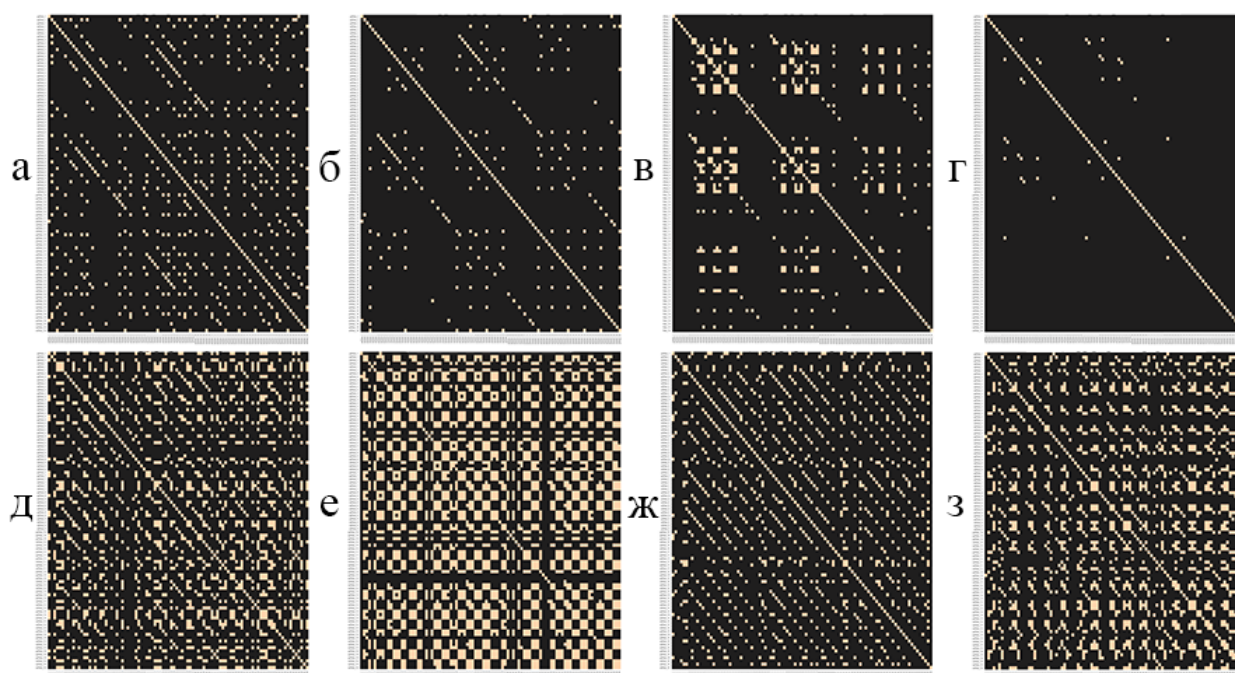


Рисунок 3.9 – Матрица взаимных корреляций между столбцами на уровне более 0,9 по модулю. (а) Набор данных, зарегистрированный при движении всей кисти во время проведения прямой линии при помощи карандаша. (б) Набор данных, где отводится указательный палец. (в) Набор данных, зарегистрированный при сгибе безымянного пальца. (г) Набор данных, где осуществляются случайные движения. (д) Набор данных, где движение осуществляется ладонью как плоскостью от себя. (е) Набор данных, где движение осуществляется ладонью как плоскостью влево. (ж) Набор данных, где движение осуществляется ладонью как плоскостью на себя. (з) Набор данных, где движение осуществляется ладонью как плоскостью вправо.

значений были построены диаграммы размаха, в которых замечена одну особенность работы ВМ160: каждый столбец матрицы движения гироскопа одной платы ВМ160 незначительно влияет на соседние столбцы. Для анализа данных эти влияния несущественны.

### **Сравнение спектра мощности сигнала в исходном и сжатом наборе данных о движении**

Для проверки предположения о сохранении информации при сжатии данных на 2 уровне вейвлет-преобразовании с применением аппроксимирующих коэффициентов (АК) необходимо сравнить спектр исходного набора данных о движении и его же сжатую версию (рис. 3.11 и рис. 3.12).

Сравнительный анализ спектра мощности до и после сжатия данных с

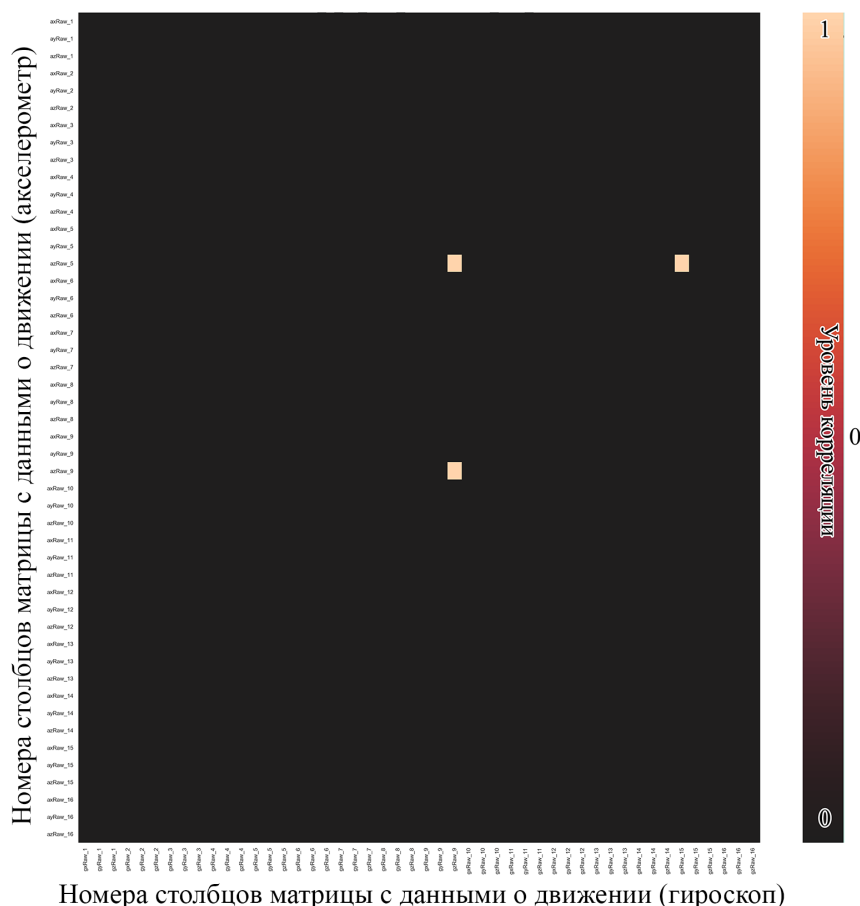


Рисунок 3.10 – Пример попарного сравнения значений акселерометров и гироскопов для НД, зарегистрированного при движении назад.

использованием вейвлет-анализа выявил, что изменения, наблюдаемые в спектре мощности, ограничиваются исключительно процессом децимации, при этом остальные основные характеристики сигнала остаются практически неизменными. (Форма сигнала, амплитуда, отношение сигнал-шум).

Для сравнения посмотрим спектр частот для детализирующих коэффициентов (ДК) (рис. 3.13). Заметна разница распределения частот в обоих случаях. В первом наибольшее число сфокусировано в низкочастотной области, во втором - в высокочастотной. Высокочастотная часть не несет в себе значимой информации о движении.

Так же рассмотрим спектры для более высоких уровней вейвлет-преобразования (ВП) с применением аппроксимирующих коэффициентов. В частности возьмем третий, четвертый и пятый уровни (рис. 3.14). Спек-

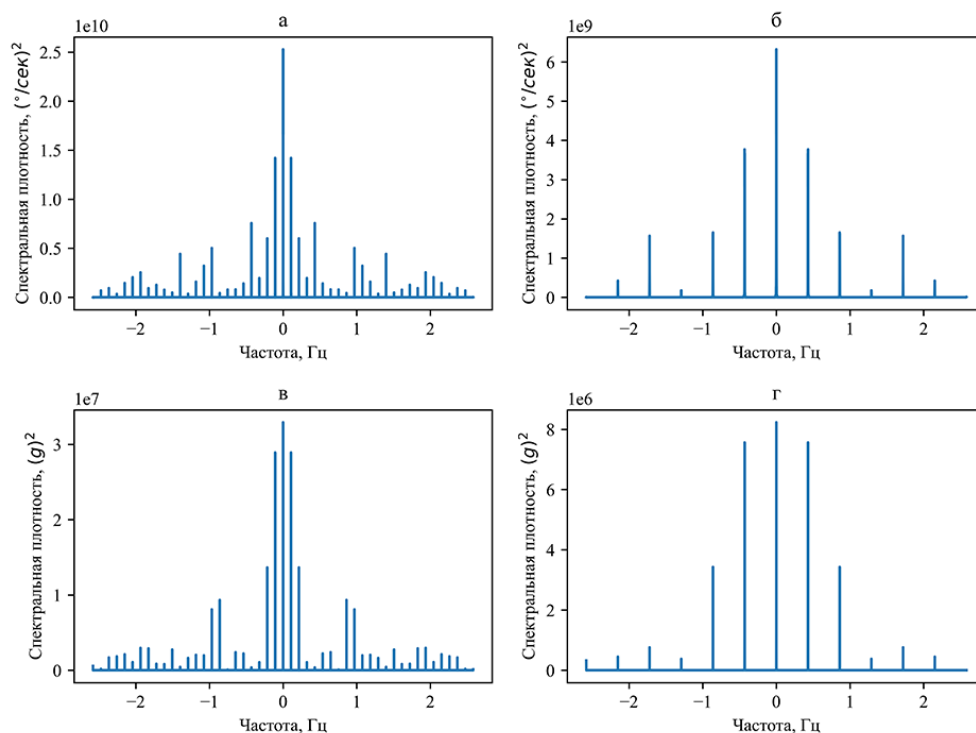


Рисунок 3.11 – Спектр мощности движения ладонью влево. (а) Исходный набор данных гироскопа. (б) Набор данных гироскопа, 2 уровня вейвлет-преобразования с АК. (в) Исходный набор данных акселерометра. (г) Набор данных акселерометра, 2 уровня вейвлет-преобразования с АК.

тры мощности при этом отличаются от референтного примера.

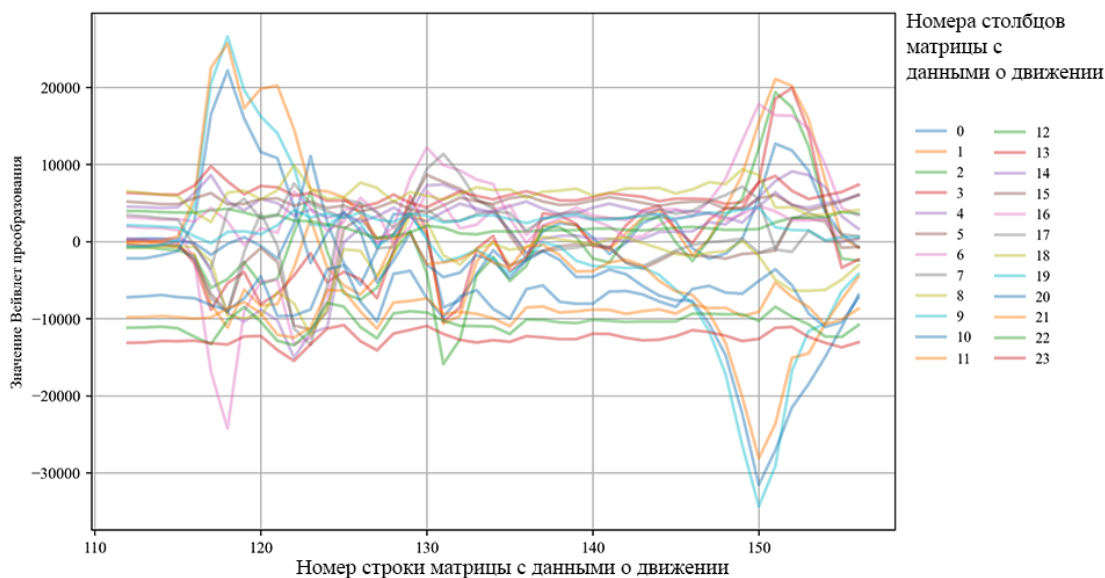


Рисунок 3.15 – НД, зарегистрированный при движении указательного пальца при движении в горизонтальной плоскости. вейвлет-преобразование второго уровня без нормирования.

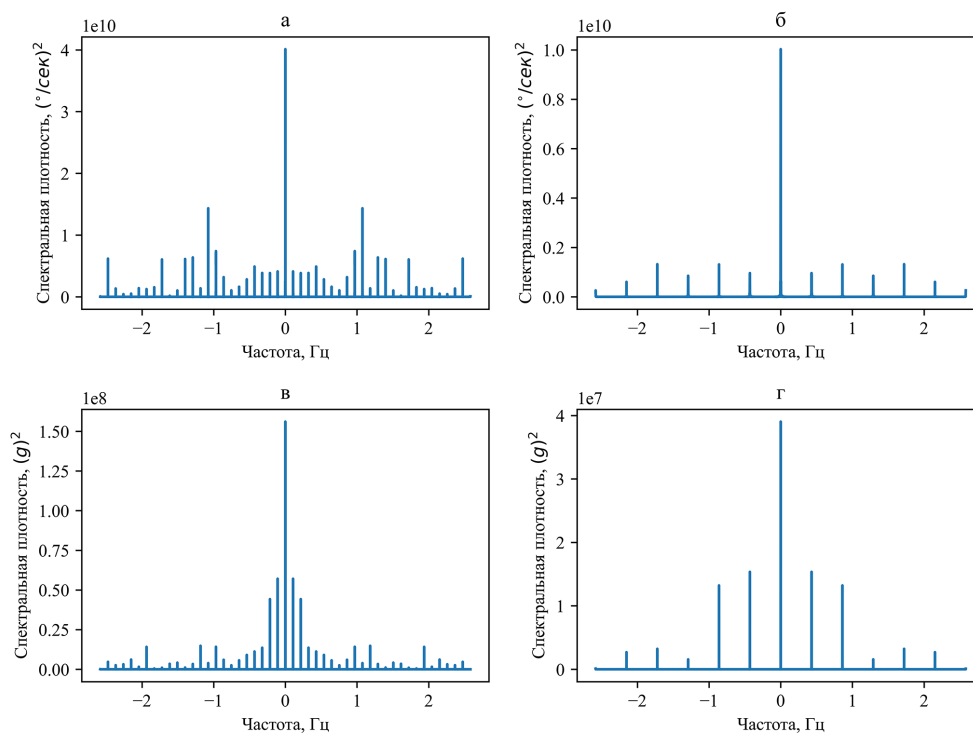


Рисунок 3.12 – Спектр мощности сгиба безымянного пальца. (а) Исходный набор данных гироскопа. (б) Набор данных гироскопа, 2 уровня вейвлет-преобразования с АК. (в) Исходный набора данных акселерометра. (б) Набор данных акселерометра, 2 уровня вейвлет-преобразования с АК.

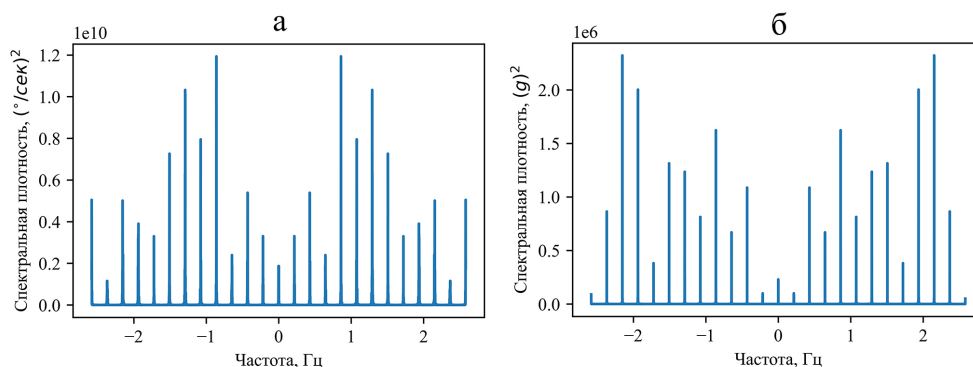


Рисунок 3.13 – Спектр мощности движения ладонью влево. (а) Набор данных гироскопа, 1 уровень вейвлет-преобразования с ДК. (б) Набор данных акселерометра, 1 уровень вейвлет-преобразования с ДК.

Наблюдается значительное количество взаимосвязанных столбцов матрицы движения в данных без применения вейвлет-преобразования. При применении вейвлет-преобразования на втором уровне количество таких столбцов уменьшается, однако они по-прежнему присутствуют. Сама процедура вейвлет-преобразования занимает в среднем 0,0022 секунды машинного времени; максимальное время составляет 0,015 секунды, минимальное

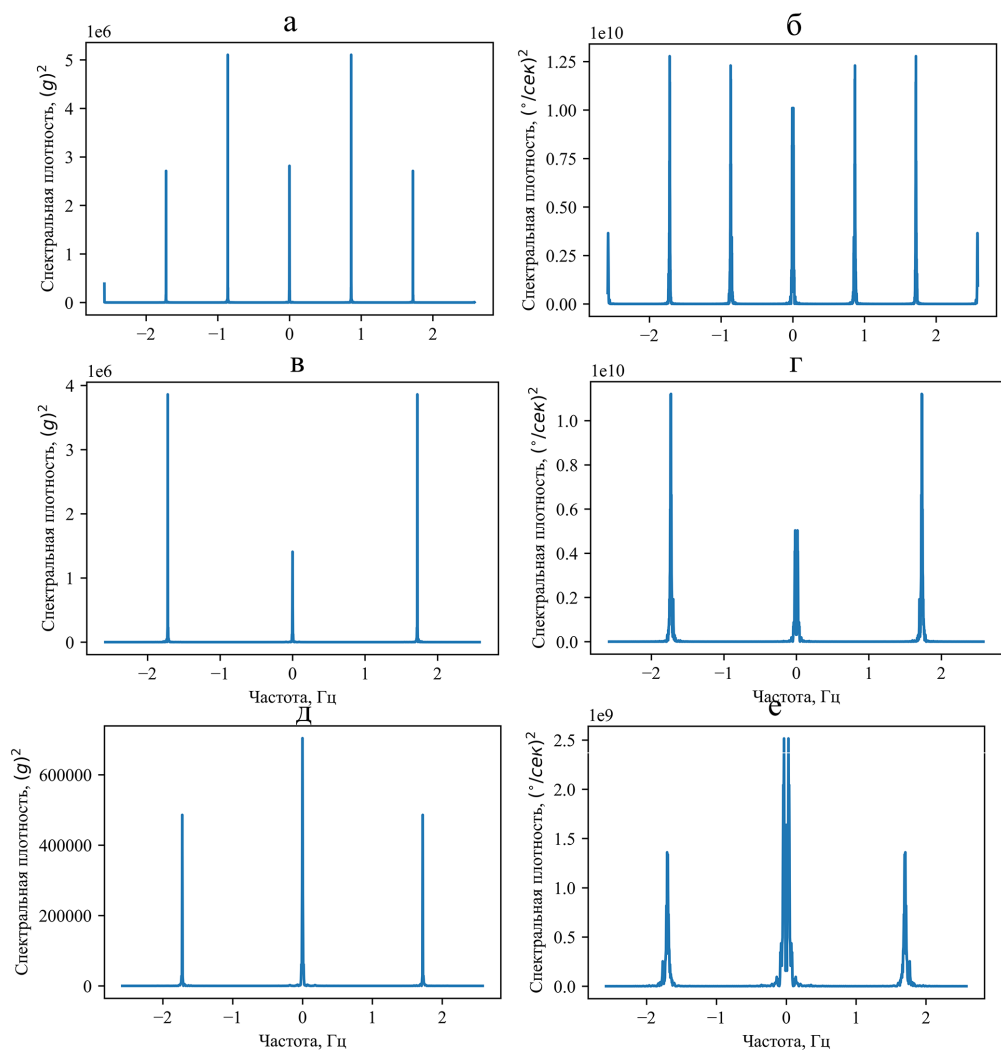


Рисунок 3.14 – Спектр мощности движения ладонью влево. (а) Акселерометр, 3 уровень ВП. (б) Гироскоп, 3 уровень ВП. (в) Акселерометр, 4 уровень ВП. (г) Гироскоп, 4 ВП. (д) Акселерометр, 5 уровень ВП. (е) Гироскоп, 5 уровень ВП.

- 0,00045 секунды.

После вейвлет-преобразования были сформированы обобщенные, эталонные цифровые паттерны движения на основе части набора данных. Записи движений были приведены к единой (максимальной) размерности путем растяжения более коротких записей. Более подробно о реализованном алгоритме смотрите в Приложении I. Листинг подпрограммы формирования общего цифрового паттерна движения. Далее в иллюстративных целях приводятся графики только для тестовой выборки с нормировкой значений для иллюстрации значимых нюансов метода обработки данных движения. Для каждого из 7 сформированных эталонных цифровых паттернов была

создана матрица взаимных корреляций столбцов.

### **3.3 Методика снижения количества столбцов в матрице с данными о движении**

С учетом вейвлет-преобразования, применение корреляционного анализа показывает, что избыточны из 24 столбцов матрицы движения:

- 14 для НД, зарегистрированный при движении всей кисти во время проведения прямой линии при помощи карандаша;
- 9 для НД, зарегистрированный при движении указательного пальца при движении в горизонтальной плоскости;
- 8 для НД, зарегистрированный при сгибе безымянного пальца;
- 14 для НД, зарегистрированный при движении ладонью от себя;
- 16 для НД, зарегистрированный при движении ладонью влево;
- 14 для НД, зарегистрированный при движении ладонью на себя;
- 16 для НД, зарегистрированный при движении ладонью вправо.

В результате сохраняются номера столбцов матрицы движения с наименьшим порядковым номером. Если учесть предварительную сортировку столбцов матрицы движения и тенденцию столбцов матрицы движения коррелировать с сонаправленными выходными каналами, можно предположить, что таким образом можно не только исключать определенные столбцы матрицы движения, но и полностью отключать некоторые датчики. Итоговые цифровые паттерны движений представлены на рис. 3.16 -3.17.

Из представленных изображений видно, что разнообразие столбцов матрицы движения увеличилось. Есть потенциал для дальнейшего уменьшения признакового пространства, однако, чем больше мы отходим от сильных линейных связей, тем ближе подходим к нелинейным связям. Для задачи вычитания линейных связей корреляционного анализа достаточно для уменьшения избыточности на уровне снижения общего числа столбцов с 33,3% до 66,6%. Для выявления нелинейных связей необходимо отдельное исследование с применением других подходов (например, нейросетей).

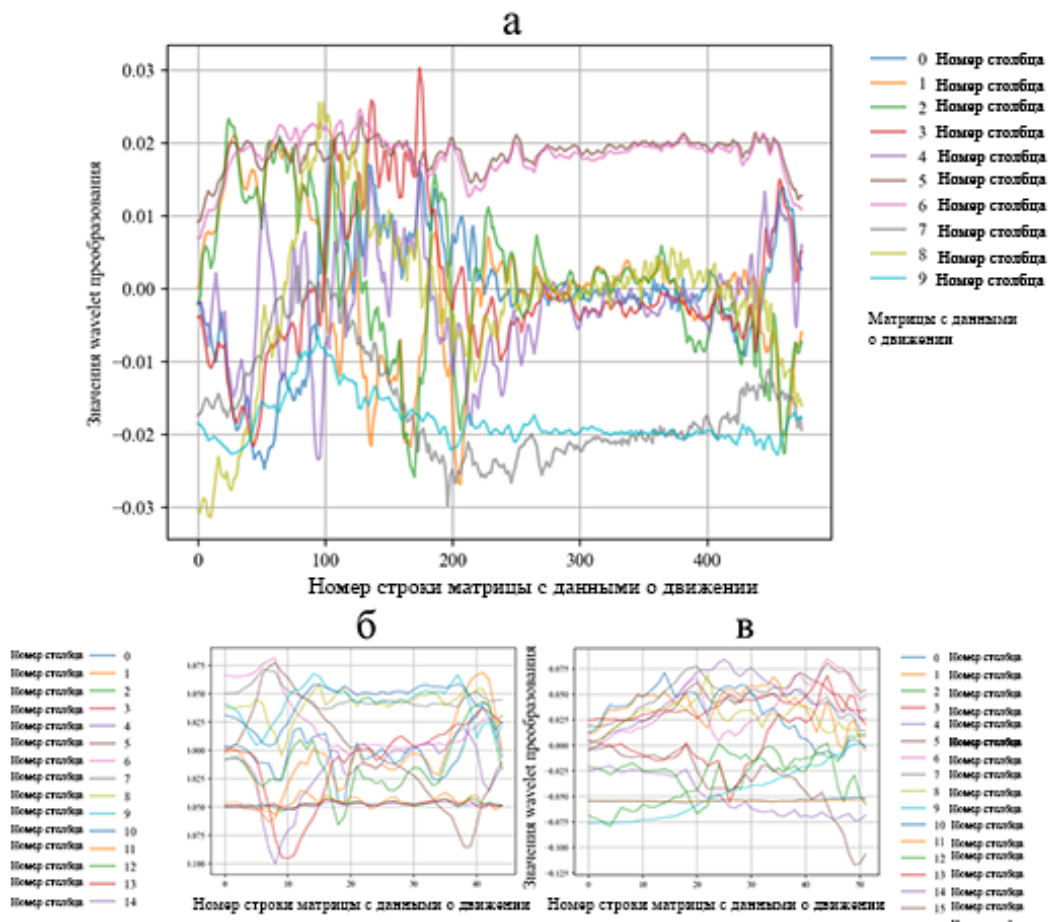


Рисунок 3.16 – а) Паттерн, зарегистрированный при движении всей кисти во время проведения прямой линии при помощи карандаша; б) паттерн, зарегистрированный при движении указательного пальца при движении в горизонтальной плоскости; в) паттерн, зарегистрированный при сгибе безымянного пальца.

### Алгоритм принятия решений исключения столбцов из расчетов

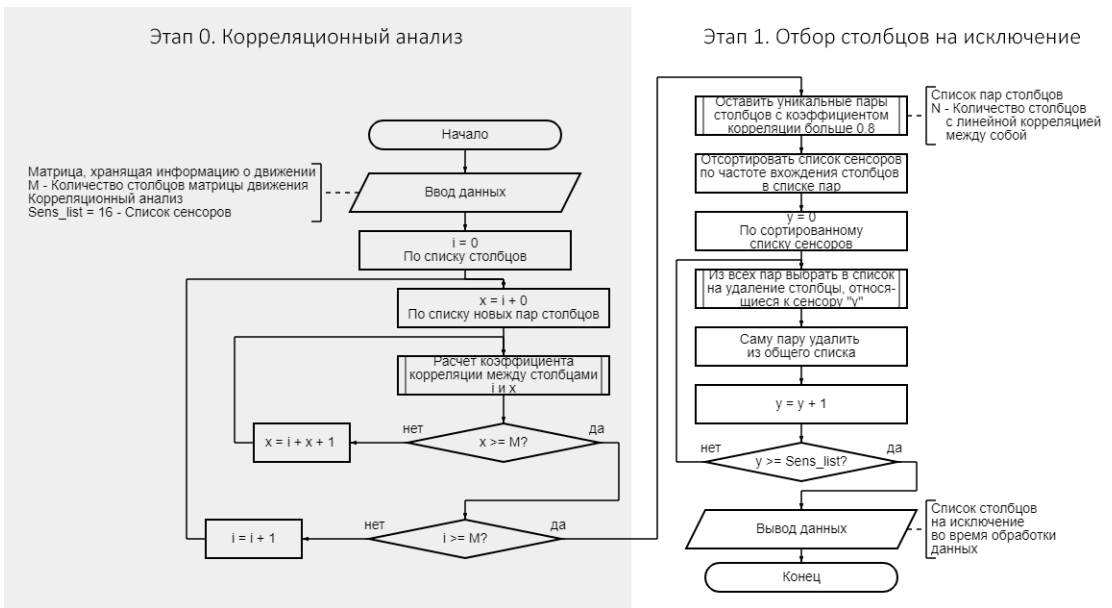


Рисунок 3.18 – Алгоритм принятия решений исключения столбцов из расчетов



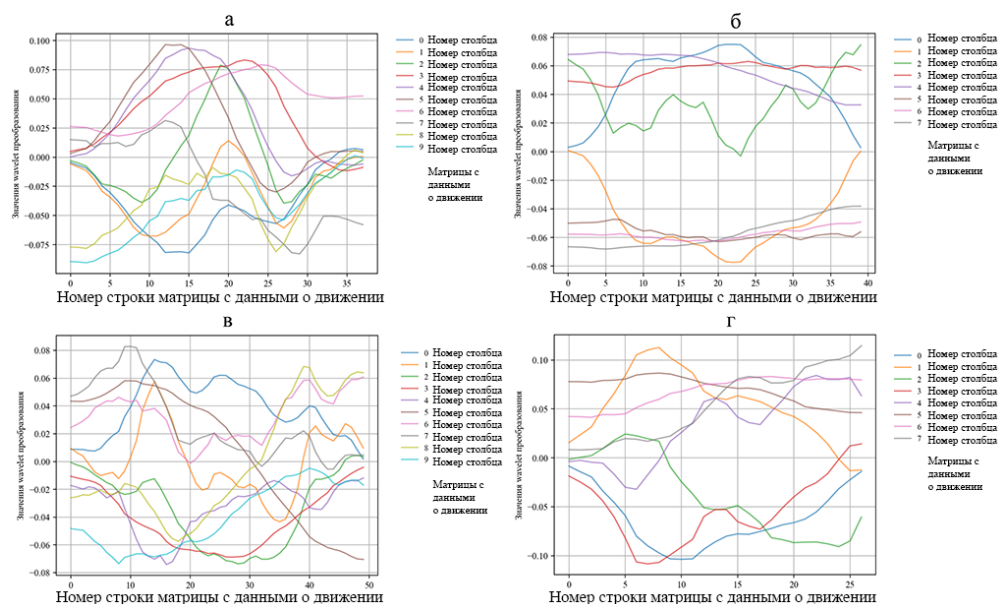


Рисунок 3.17 – а) Паттерн, зарегистрированный при движении ладонью от себя; б) паттерн, зарегистрированный при движении ладонью влево; в) паттерн, зарегистрированный при движении ладонью на себя; г) паттерн, зарегистрированный при движении ладонью вправо.

Алгоритм отбора признаков строится следующим образом:

1. На вход функции отбора идет матрица взаимных корреляций. Столбцы и строки соответствуют номеру столбцов матрицы движения.
2. В начале стоп-лист столбцов матрицы движения пустой.
3. По мере прохождения по столбцам отбираются столбцы матрицы движения с корреляцией больше или равно 0,9. Они же добавляются в стоп-лист.
4. Если номера столбца нет в стоп-листе, то переходим к следующему столбцу и далее возвращаемся к шагу 3. Таким образом действуем, пока не закончатся столбцы матрицы движения.

Подробнее в Приложении К. Листинг подпрограммы отбора признаков.

### Общая схема обработки данных

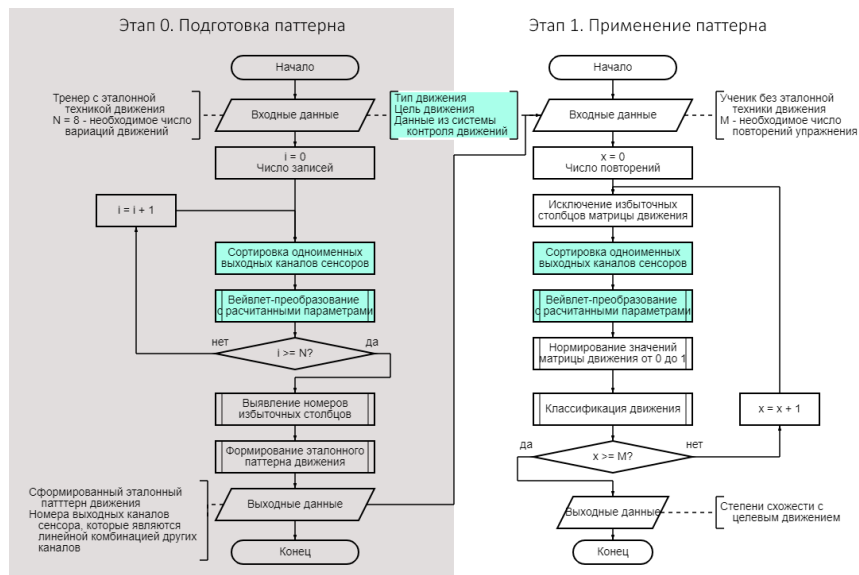


Рисунок 3.19 – Общая схема обработки данных

### 3.4 Выводы по главе

1. Разработан план сбора данных о движениях, описывающих различные биомеханические связи в кисти человека.
2. Предложена методика трансформации данных о движении с помощью вейвлет-преобразования.
3. Предложена методика выбора уровня вейвлет-преобразования.
4. Предложена методика комбинации столбцов матрицы движения, основанный на их корреляционной близости.
5. Обосновано применение аппроксимирующих коэффициентов при вейвлет-преобразовании.
6. Проведено сравнение спектров в исходном и сжатом наборе данных о движении.
7. Продемонстрирована методика отбора столбцов в матрице с данными о движении, являющимися линейной комбинацией других столбцов.
8. Суммарное уменьшение объема данных составило от 24 до 48 раз по сравнению с начальным объемом.

Процесс сжатия данных, связанных с движением, позволяет определить оптимальный уровень вейвлет-преобразования, при котором сохраняется возможность полного восстановления информации о самом движении, в то

время как остальная информация исключается. В контексте цифровизации человеческого движения кисти с использованием 16 датчиков и 96 столбцов матрицы движения, можно применить двухуровневое вейвлет-преобразование, что сокращает объем данных в 16 раз. Данные выводы справедливы при работе с комплексом цифровизации движения на уровне 212 Гц и при основном диапазоне повседневных движений, не превышающем 20 Гц.

## Глава 4

# Разработка алгоритмической и программной части системы контроля движений кисти человека

Уменьшение размерности матрицы с информацией о движении обеспечивает решение глобальной задачи компактного хранения данных, что приводит к снижению требуемых вычислительных ресурсов при сравнении цифровых паттернов на устройствах с ограниченной мощностью. Основным методом определения расстояния между двумя цифровыми паттернами является алгоритм динамического искажения временной шкалы, который обладает вычислительной сложностью  $O(n^2)$ .

В данном разделе иллюстрируется функционирование комплекса контроля движений кисти человека. Были проведены оценки, подтверждающие точность идентификации движений, включая оценки вероятности ошибок первого и второго рода. Были получены оценки расстояний с использованием метрики, основанной на алгоритме динамического искажения временной шкалы, для каждого класса движений. Эти оценки отражают степень отличия различных классов движений друг от друга.

Реализация алгоритма в программном обеспечении была продемонстрирована, и было показано, как увеличилась эффективность комплекса контроля движений в ходе экспериментального использования. Было рассчитано соотношение затраченного машинного времени для тестовых и исследуемых наборов данных.

## 4.1 Учет алгоритмических особенностей в аппаратной части комплекса

### Общая схема работы комплекса

Рассмотрим этапы работы комплекса. После подачи питания на микроконтроллер запускается точка доступа Wi-Fi. Затем инициализируются последовательно все датчики. Далее комплекс начинает отправлять UDP-пакеты на IP-адрес, который заранее известен. Клиентское приложение подключается к точке доступа для приема. Основной режим работы комплекса — непрерывная передача данных. Сбор данных осуществляется путем последовательного опроса 16 датчиков, а затем формируется UDP-пакет.

Используется общая шина SPI. Мультиплексор необходим, так как данный МК не имеет достаточно цифровых выводов для всех линий выбора микросхемы (CS). Однако как показал логический анализатор, это не сказывается на производительности комплекса. Время переключения данного мультиплексора — 9 нс. МК также хватает запаса производительности для управления ножками.

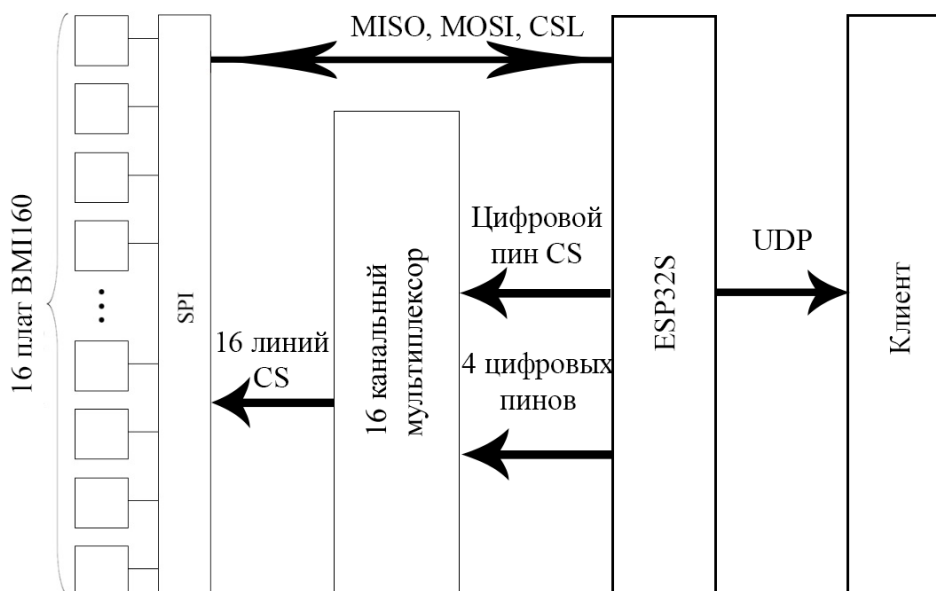


Рисунок 4.1 – Архитектура аппаратной части комплекса.

МК выполняет несколько функций. Первая — поддержание беспровод-

ного протокола для передачи данных. Вторая — инициализация всех датчиков через драйвер. Третья — формирование максимального числа пакетов данных в единицу времени.

Главной задачей работы комплекса является оцифровка движения человека для последующей отправки результата обработки на устройство графического вывода в режиме реального времени или с минимальной задержкой. Для этого требуется высокая производительность каждого компонента в отдельности и всего комплекса в целом. Комплекс должна работать в режиме непрерывной передачи данных, без отправки CPU в режим сна.

Комплекс, построенная на МЭМС, может функционировать в заявленных выше условиях: на датчики не влияют уровень освещенности и перекрытие конечностей, объем данных минимален из имеющихся аналогов, датчики могут быть достаточно миниатюрными для крепления на всех фалангах пальцев, а беспроводная передача данных возможна с необходимой скоростью.

Аппаратная часть комплекса (рис. 4.1) состоит из четырех блоков:

1. датчики BMI160;
2. модифицированная шина SPI;
3. 16 канальный мультиплексор;
4. МК ESP32S;

## **4.2 Разработка программного обеспечения комплекса**

### **Прошивка ESP32S**

Прошивка МК выполняет три основные функции:

1. Поддержание протокола Wi-Fi.
2. Управление 16 канальным мультиплексором.
3. Управление датчиками BMI160.

Прошивку ESP32S можно посмотреть в алгоритме 1 в виде псевдокода:

```

1 Function setup
2   start_Wi_Fi_access_point() // Запускаем Wi-Fi точку доступа.
3   setup_multiplexer_init_position()
4   // Устанавливаем мультиплексор в начальное положение.
5   initial_every_bmi160() // Инициализируем все датчики.
6 end
7 Function loop
8   beginPacket(udpAddress, udpPort)
9   // Открываем пакет для записи и указываем адрес доставки.
10  for one_sensor in sensors do
11    change_channel() // Меняем канал мультиплексора.
12    one_sensor.readMotionSensor() // Считываем данные
13    датчика.
14    write_data() // Записываем данные в пакет.
15  end
16  endPacket // Отправляем пакет.
17 end

```

**Алгоритм 1:** Псевдокод для прошивки ESP32S

Среда разработки — Arduino IDE 1.8.9. Используются как стандартные библиотеки, так и специализированные драйверы. Например, WiFi.h, ESPmDNS.h и WiFiUdp.h позволяют настраивать точку доступа и отправлять UDP пакеты. А специально разработанный драйвер<sup>12</sup> позволяет настраивать датчики.

Для управления мультиплексором не требуется специально разработанной библиотеки. Переключение каналов происходит при подаче различных комбинаций уровней сигналов на управляющие 4 линии. Например, при подаче на каждую линию логического нуля — происходит переключение на 0 канал.

Функция *setup* выполняется только один раз, *loop* — бесконечно. Она аналогична выражению *while(true)*. Передача пакетов происходит асин-

<sup>12</sup>[hanyazou BMI160 driver https://github.com/hanyazou/BMI160-Arduino](https://github.com/hanyazou/BMI160-Arduino)

хронно.

## Клиентское приложение

Клиентское приложение служит для удаленного получения данных. Это более быстрый способ, чем проводной вариант через виртуальный COM-порт. Для разработки использовали среду Spyder 3.3.4 и язык Python 3.7.

Всего в ходе разработки требовалось реализовать две функции от клиента. Первая — подсчет частоты прихода пакетов данных. Вторая — сохранение самих данных.

Сохранение данных можно представить в виде алгоритма 2.

```
1 Function Data receiving
2   open SOCK()
3   // Открываем сокет, на который отправляются данные.
4   open_output_csv_file(mode = a)
5   // Открываем файл в режиме добавления строк в конец файла.
6   write_column_names() // Записываем названия выходных
   каналов датчиков.
7   while key_board_interrupt do
8     | data_receive() // Получаем фрейм данных.
9     | data_save() // Сохраняем один фрейм данных в конец файла.
10  end
11  compression(bz2)
12  // Сжать CSV.
13 end
```

**Алгоритм 2:** Алгоритм приема данных.

Данные в UDP пакет добавляются последовательно, согласно включенному датчику. Один датчик добавляет 6 значений. В итоге для 16 датчиков получается 96 столбцов матрицы движения. Состав данных для одного датчика:

1.  $G_x$  – выходной канал X гироскопа.
2.  $G_y$  – выходной канал Y гироскопа.



3.  $G_z$  – выходной канал  $Z$  гироскопа.
4.  $A_x$  – выходной канал  $X$  акселерометра.
5.  $A_y$  – выходной канал  $Y$  акселерометра.
6.  $A_z$  – выходной канал  $Z$  акселерометра.

### 4.3 Анализ результатов использования программно - аппаратного комплекса

#### Применение алгоритма DTW для определения доверительных интервалов расстояний между цифровыми паттернами

При сравнении цифровых паттернов использовался алгоритм динамического искажения временной шкалы, который может сопоставлять их между собой.

Данная функция была применена для четырех наборов данных. Для двух контрольных и двух тестовых. Между собой они отличаются только нормированием значений. Всего было произведено 224 сравнения паттернов для одного запуска. В целях получения более объемной статистики количество запусков было равным 50.

Функцию сравнения цифровых паттернов в общем виде можно посмотреть в алгоритме 3 в формате псевдокода.

Для реализации этого программного модуля были задействованы сторонние открытые библиотеки `pandas`, `numpy`, `statsmodels` и `dtaidistance`<sup>12</sup>. Последняя используется научной группой Declarative Languages and Artificial Intelligence (DTAI<sup>13</sup>). Там реализованы две версии алгоритма: с применением чистого языка `python 3` и с расширениями на языке `C`.

---

<sup>12</sup>Library for time series distances (e.g. Dynamic Time Warping) <https://github.com/wannesm/dtaidistance>

<sup>13</sup>Declarative Languages and Artificial Intelligence section <https://dtai.cs.kuleuven.be/>

```

1 Function DTW pattern comparison
2   get_list_of_reference_patterns() // Получить список эталонных паттернов.
3   get_list_of_recorded_patterns() // Получить список записанных паттернов.
4   for one_ref_pattern in reference_patterns do
5     load_list_of_forbidden_columns() // Список запрещенных столбцов.
6     get_list_of_whole_columns() // Получаем список всех столбцов.
7     get_list_of_permitted_columns()
8     // Вычитаем из списка всех столбцов список запрещенных столбцов.
9     load_reference_pattern_with_permitted_columns()
10    // Загружаем эталонный паттерн с учетом разрешенных столбцов.
11    for one_recorded_pattern in recorded_patterns do
12      load_recorded_pattern_with_permitted_columns()
13      // Загружаем записанный паттерн с учетом разрешенных столбцов.
14      calculate_DTW_distance()
15      // Рассчитываем расстояние DTW для многомерного случая.
16      get_metadata() // Собираем метаданные.
17    end
18  end
19 end

```

**Алгоритм 3:** Псевдокод расчета расстояния между многомерными цифровыми паттернами.

Вторая версия быстрее, однако требует дополнительной поддержки OpenMP. В данной работе применяется первая реализация, так как цель работы не в настройке комплекса на режим работы в реальном времени, а в демонстрации того, как работа исключительно с данными может помочь ускорить обработку для любых реализаций выбранного алгоритма сравнения цифровых паттернов.

Для увеличения быстродействия работы также возможно применить ряд дополнительных улучшений, которые планируются в дальнейшей работе. Подробнее см. в заключении.

## Статистика сравнение контрольной и тестовой выборки

Для оценки эффектов от оптимизации сложности системы будут применяться следующие метрики: время выполнения выбранного алгоритма обработки данных и объем памяти, необходимый для полного цикла обработки данных, начиная с приема исходного пакета данных о движении и заканчивая сравнением полученного паттерна движения с эталонными. Анализ выполнялся в среде PyCharm 2021.2, с использованием Python версии 3.9. Время измерялось с помощью стандартной функции Python `time.perf_counter()`, а память определялась с помощью функции `sys.getsizeof()`.

После измерения всех расстояний между цифровыми паттернами движения необходимо убедиться, что цифровой паттерн действительно способен качественно отличать схожие образцы. Для этого для каждого эталонного цифрового паттерна каждого типа движения был построен 95% доверительный интервал расстояний.

Алгоритм построения Доверительного Интервала Расстояний (ДИР) на основе метода динамической трансформации временной шкалы (DTW) может быть описан следующим образом:

- Берем первый эталонный паттерн, состоящий из усреднения 8 записей.
- Для первого эталона вычисляем расстояние от всех его исходных записей до него самого и строим 95% доверительный интервал вокруг полученной статистики.

$$\text{ДИР}_{mean1} = \frac{1}{8} \sum_{j=1}^8 DTW(\text{Eta}_1, \text{Ishod}_j) \quad (4.1)$$

где  $\text{ДИР}_{mean1}$  - среднее значение доверительного интервала расстояний для первого эталонного цифрового паттерна движений,

DTW - алгоритм динамической трансформации временной шкалы,

$\text{Eta}_1$  - первый эталонный цифровой паттерн движений,

Ishod - исходная запись первого цифрового паттерна движения,

$i$  - номер исходной записи эталонного примера движений.

$$STD_1 = \sqrt{\frac{1}{8} \sum_{j=1}^8 DTW(Eta_1, Ishod_j) - ДИР_{mean1}}^2 \quad (4.2)$$

где  $STD_1$  - стандартное отклонение для выявленной статистики расстояний первого цифрового эталонного паттерна движений к его исходным записям,

$ДИР_{mean1}$  - среднее значение доверительного интервала расстояний для первого эталонного цифрового паттерна движений,

$DTW$  - алгоритм динамической трансформации временной шкалы,

$Eta_1$  - первый эталонный цифровой паттерн движений,

$Ishod$  - исходная запись первого цифрового паттерна движения,

$i$  - номер исходной записи эталонного примера движений.

$$ДИР_1 = [ДИР_{mean1} - V_{0.975} \times \frac{STD_1}{\sqrt{n}} ДИР_{mean1} + V_{0.975} \times \frac{STD_1}{\sqrt{n}}] \quad (4.3)$$

Где:

$STD_1$  - стандартное отклонение для выявленной статистики расстояний первого цифрового эталонного паттерна движений к его исходным записям,

$ДИР_{mean1}$  - среднее значение доверительного интервала расстояний для первого эталонного цифрового паттерна движений,

$V_{0.975}$  - значение квантиля нормального распределения, соответствующее 95% доверительному интервалу,

$n$  - число исходных примеров.

- Считаем такую же статистику для всех остальных исходных записей по отношению к первому эталону.
- Повторяем шаги для каждого из оставшихся эталонных паттернов, чтобы получить соответствующие средние расстояния и доверительные интервалы. Для каждого класса будут значения для попадания в класс и для его не попадания (если это другое известное движение)
- Проверяем, пересекается ли ДИР для расстояний до цифровых пат-

тернов схожих движений с доверительными интервалами других типов движений. Этого не должно быть.

- Если происходит пересечение, это указывает на то, что значения расстояния лежат в диапазоне, где невозможно однозначно классифицировать движение.
- Используя ДИР, мы можем отнести неизвестное исследуемое движение к определенному классу на основе того, попадает ли его расстояние в доверительный интервал этого класса или нет.
- Если нет попадания ни в один из классов - значит это неизвестное движение.

Таким образом, ДИР позволяет нам принять решение о принадлежности движения к определенному классу на основе анализа расстояний и пересечений с доверительными интервалами для различных типов движений.

Для расстояний до образцов схожих движений, ДИР не должен пересекаться с доверительными интервалами других типов движений. Если такое пересечение происходит, это означает, что существует диапазон значений, в котором невозможно достоверно отнести движение к определенному классу.

В таб. 4.1 - 4.7 представлены все доверительные интервалы в статистическом смысле (95% доверительные интервалы относительно выборочного среднего).

Таблица 4.1 – Исходный набор данных без применения вейвлет-преобразования и нормирования данных.

	Ср. значение расстояния того же типа движений	Ср. значение расстояния для остальных движений	ДИР для того же типа движений	ДИР для остальных движений	Есть ли пересечение с другими диапазонами
<b>Линия</b>	1118425,6	798362,0	От 1029928,1 до 1206923,1	От 775866,6 до 820857,5	Да
<b>Сгиб</b>	321576,1	559439,9	От 272840,5 до 370311,8	От 477673,1 до 641206,6	Нет
<b>Отведение</b>	314996,4	545015,0	От 273576,9 до 356415,9	От 463736,3 до 626293,7	Нет
<b>От себя</b>	469632,5	513080,9	От 355417,7 до 583847,4	От 429689,7 до 596472,0	Да
<b>Влево</b>	330652,2	540882,3	От 303852,7 до 357451,6	От 457860,1 до 623904,5	Нет
<b>На себя</b>	427546,8	533546,7	От 339398,4 до 515695,1	От 451225,0 до 615868,3	Да
<b>Вправо</b>	449845,9	508813,3	От 387224,3 до 512467,5	От 422814,2 до 594812,5	Да

В случае прямой линии (см. таб. 4.1), формально нет пересечения диапазонов расстояний, однако стоит отметить, что без нормализованных значений общий цифровой паттерн движений выдает большие расстояния для своего типа движений, чем для остальных. Это противоречит ожидаемому, так как в идеальном случае распознавания цифрового паттерна расстояния для своего типа движений должны варьироваться от 0 до некоторого значения, а все остальные расстояния должны быть строго больше этого значения. В данном случае получилось обратное, поэтому в столбце с пересечениями указан положительный результат.

Таблица 4.2 – Ошибки первого и второго рода в датасете.

Ошибок первого рода (ложноположительные), % от датасета	Ошибок второго рода (ложноотрицательные), % от датасета
17,36	14,28

В сумме 31,64% данных нельзя использовать. Данный набор данных не

проходит по качественному наполнению.

Таблица 4.3 – Исходный набор данных без применения вейвлет-преобразования, но с нормированием данных.

	Ср. значение расстояния того же типа движений	Ср. значение расстояния для остальных движений	ДИР для того же типа движений	ДИР для остальных движений	Есть ли пересечение с другими диапазонами
<b>Линия</b>	10,3	27,3	От 9,83 до 10,88	От 26,02 до 28,59	Нет
<b>Сгиб</b>	11,1	11,0	От 10,14 до 12,09	От 10,76 до 11,38	Да
<b>Отведение</b>	9,7	10,6	От 9,40 до 10,13	От 10,31 до 10,90	Нет
<b>От себя</b>	9,7	10,1	От 9,26 до 10,22	От 9,93 до 10,40	Да
<b>Влево</b>	9,5	10,2	От 9,21 до 9,84	От 9,98 до 10,58	Нет
<b>На себя</b>	10,06	10,8	От 9,29 до 10,83	От 10,53 до 11,22	Да
<b>Вправо</b>	9,6	9,6	От 9,26 до 10,04	От 9,49 до 9,75	Да

Анализ представленной таблицы позволяет сделать вывод, что наиболее долгие движения - взятие ручки, проведение прямой линии и бросание ручки - с использованием нормированных данных проявляются не только качественно, но и с существенным отрывом. Если бы было больше длительных движений, этот эффект, вероятно, был бы менее ярко выражен или вовсе нейтрализован - как это происходит с другими цифровыми паттернами. Для остальных цифровых паттернов нормирование данных не приносит выгоды. Далее мы рассмотрим еще две таблицы с метаданными.

Таблица 4.4 – Ошибки первого и второго рода в датасете.

Ошибок первого рода (ложноположительные), % от датасета	Ошибок второго рода (ложноотрицательные), % от датасета
28,91	0

Количество ошибок незначительно снизилось.

Таблица 4.5 – Исходный набор данных с применением вейвлет-преобразования и без нормированием данных.

	Ср. значение расстояния того же типа движений	Ср. значение расстояния для остальных движений	ДИР для того же типа движений	ДИР для остальных движений	Есть ли пересечение с другими диапазонами
<b>Линия</b>	501615,8	396468,8	От 463274,3 до 539957,4	От 371904,1 до 421033,6	Да
<b>Сгиб</b>	160064,4	404508,7	От 129722,1 до 190406,7	От 340376,1 до 468641,3	Нет
<b>Отведение</b>	141233,7	341342,7	От 128672,4 до 153795,0	От 286096,2 до 396589,3	Нет
<b>От себя</b>	182486,4	286189,5	От 129047,1 до 235925,8	От 236013,3 до 336365,7	Нет
<b>Влево</b>	109509,9	235868,3	От 103430,9 до 115588,9	От 204763,0 до 266973,6	Нет
<b>На себя</b>	138772,8	291540,2	От 97982,7 до 179562,9	От 241397,7 до 341682,6	Нет
<b>Вправо</b>	125100,7	241074,4	От 100602,7 до 149598,7	От 203626,1 до 278522,6	Нет

Таб.4.5 показывает, что вейвлет-преобразование с последующим отбором параметров помогает разделить простые движения, однако проблема с макро- и составными движениями остается. Стоит обратить внимание на движение «от себя»: хотя формально оно не имеет пересечений доверительных диапазонов, расстояние между ними близко. Ранее мы говорили о построении 95% доверительного интервала. Если установить более строгие критерии, например, построить 99% интервал, то даже без дополнительных расчетов можно утверждать, что данный цифровой паттерн не пройдет тест на пересечение диапазонов. Это можно было бы отнести на счет неточности сформированного цифрового паттерна, но в журнале эксперимента есть антагонистические движения. Противоположное действие - движение «на себя» - должно было бы показать аналогичный результат. Если посмотреть на эталонные цифровые паттерны (рис.4.2), то в целом они представляют собой инвертированные версии друг друга.



Таблица 4.6 – Ошибки первого и второго рода в датасете.

Ошибок первого рода (ложноположительные), % от датасета	Ошибок второго рода (ложноотрицательные), % от датасета
2,31	14,23

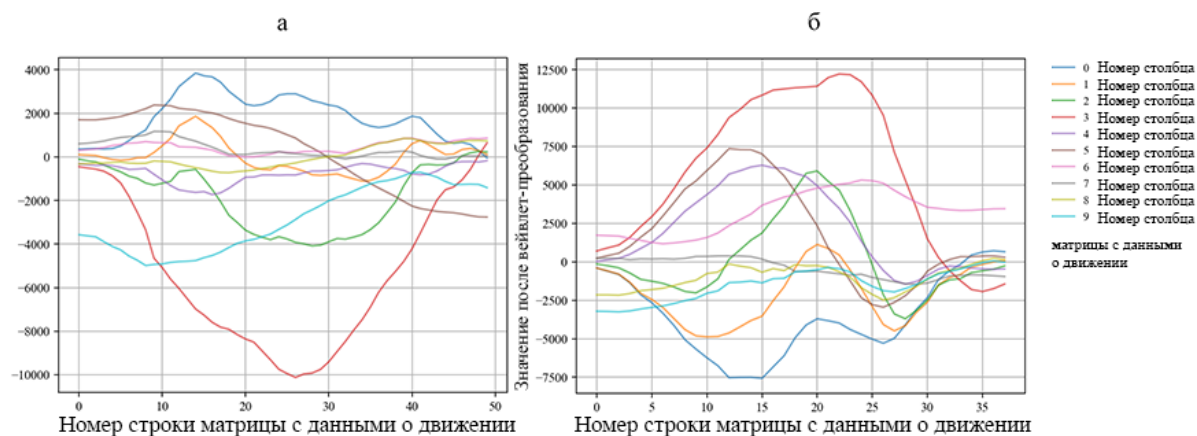


Рисунок 4.2 – Общий цифровой паттерн для движения: а) от себя; б) на себя.

Таблица 4.7 – Исходный набор данных с применением вейвлет-преобразования и нормированием данных.

	Ср. значение расстояния того же типа движений	Ср. значение расстояния для остальных движений	ДИР для того же типа движений	ДИР для остальных движений	Есть ли пересечение с другими диапазонами
<b>Линия</b>	2,8	8,2	От 2,72 до 3,03	От 7,76 до 8,79	Нет
<b>Сгиб</b>	3,7	4,8	От 3,45 до 4,12	От 4,76 до 5,02	Нет
<b>Отведение</b>	3,2	4,0	От 3,06 до 3,42	От 3,96 до 4,19	Нет
<b>От себя</b>	2,5	3,2	От 2,41 до 2,77	От 3,12 до 3,29	Нет
<b>Влево</b>	2,0	2,7	От 1,95 до 2,16	От 2,62 до 2,87	Нет
<b>На себя</b>	2,7	3,5	От 2,49 до 3,06	От 3,45 до 3,60	Нет
<b>Вправо</b>	2,1	2,8	От 2,01 до 2,23	От 2,81 до 2,94	Нет

Таблица 4.8 – Ошибки первого и второго рода в датасете.

Ошибок первого рода (ложноположительные), % от датасета	Ошибок второго рода (ложноотрицательные), % от датасета
2,33	2,61

Как и предполагалось ранее, нормирование данных помогает решить проблему со сложными движениями. Исходя из результатов, можно утверждать, что отбор признаков - это не только часть оптимизации алгоритмов, но и необходимость. Простое увеличение числа датчиков (и, как следствие, выходных каналов) не всегда позволяет добиться цели — более точного распознавания паттернов, а иногда даже уводит от нее. Это контринтуитивный вывод, полученный в ходе рассмотрения четырех по-разному подготовленных наборов данных.

Давайте теперь рассмотрим временные эффекты от разработанной схемы работы с данными. Общая схема движения информации выглядит следующим образом:

1. АЦП преобразование данных на IMU BMI160.
2. Формирование пакета данных на МК для 16 датчиков BMI160.
3. Запись пакета данных в формат CSV с разделителями типа запятая.
4. Запись серии данных для каждого типа движений.
5. Вейвлет-преобразование.
6. Нормализация значений в диапазон  $\pm 1$ .
7. Формирование эталонного движения.
8. Снижение признакового пространства с помощью корреляционного анализа.
9. Сравнение эталонного цифрового паттерна и исследуемого.

Из анализа выполненных преобразований данных, контрольная и тестовая выборки различаются существенно только в пятом пункте. Следовательно, к времени определения расстояния в этом контексте будет добавлено константное время (среднее) на выполнение вейвлет-преобразования. Среднее время выполнения вейвлет-преобразования составляет 2,2 мс, максимальное - 15,07 мс, минимальное - 0,45 мс. Нормализация значений бу-

дет происходить быстрее, если будет принято меньшее количество столбцов матрицы движения. Уменьшение размерности признакового пространства - это однократная задача, которую необходимо выполнить на начальном этапе проектирования обобщенного цифрового паттерна движения. Далее для практической работы достаточно либо игнорировать определенные выходные каналы, либо отключить их передачу на микроконтроллере. Это сделано для объективной оценки времени работы с данными в общем.

Сравнение проведено по четырем наборам данных для контрольной и тестовой выборки (таб. 4.9).

Таблица 4.9 – Затраченное машинное время на расчет расстояния между паттернами, разделенное по классам движений и набору данных: а) набор данных для начальных значений датчика без нормализации; б) набор данных для начальных значений датчика с нормализацией; в) набор данных с вейвлет-преобразованием без нормализации; г) набор данных с вейвлет-преобразованием и нормализацией.

	<b>а, Среднее, сек 95% ДИ, сек</b>	<b>б Среднее, сек 95% ДИ, сек</b>	<b>в Среднее, сек 95% ДИ, сек</b>	<b>г Среднее, сек 95% ДИ, сек</b>
<b>Линия</b>	4,063 2,428 - 5,698	3,791 2,268 - 5,314	0,238 + 0,0022 0,142 - 0,333	0,235 +0,0022 0,141 - 0,330
<b>Сгиб</b>	0,444 0,265 - 0,623	0,413 0,247 - 0,579	0,026 +0,0022 0,015 - 0,036	0,026 +0,0022 0,015 - 0,036
<b>Отведение</b>	0,389 0,232 - 0,545	0,362 0,215 - 0,508	0,022 +0,0022 0,013 - 0,031	0,022 +0,0022 0,013 - 0,031
<b>От себя</b>	0,324 0,194 - 0,454	0,301 0,180 - 0,421	0,019 +0,0022 0,011 - 0,026	0,019 +0,0022 0,011 - 0,026
<b>Влево</b>	0,345 0,206 - 0,484	0,316 0,189 - 0,442	0,020 +0,0022 0,012 - 0,028	0,020 +0,0022 0,012 - 0,028
<b>На себя</b>	0,431 0,258 - 0,605	0,402 0,241 - 0,564	0,025 +0,0022 0,015 - 0,035	0,025 +0,0022 0,015 - 0,035
<b>Вправо</b>	0,238 0,142 - 0,334	0,221 0,132 - 0,310	0,013 +0,0022 0,008 - 0,019	0,013 +0,0022 0,008 - 0,019

В дополнение к средним оценкам также приведены 95% доверительные интервалы (ДИ). +0.0022 - это время, которое нужно добавить для полного сравнения, учитывая время, необходимое для выполнения вейвлет-преобразования.

Среднее отношение тестовых к контрольным выборкам составляет 15,87 (таб. 4.10) при измерении времени работы для всех цифровых паттернов движений. Это указывает на линейную зависимость скорости расчетов от вейвлет-преобразования и дополнительное ускорение благодаря уменьшению размерности признакового пространства через корреляционный анализ. Дополнительное время на вейвлет-преобразование незначительно снижает этот показатель. Было подтверждено, что использование данных с удаленной избыточностью увеличивает эффективность работы алгоритма.

Таблица 4.10 – Затраченное общее машинное время на расчет расстояния между цифровыми паттернами: а) набор данных для начальных значений датчика без нормализации; б) набор данных для начальных значений датчика с нормализацией; в) набор данных с вейвлет-преобразованием без нормализации; г) набор данных с вейвлет-преобразованием и нормализацией.

	<b>а,</b> Среднее, сек 95% ДИ, сек	<b>б</b> Среднее, сек 95% ДИ, сек	<b>в</b> Среднее, сек 95% ДИ, сек	<b>г</b> Среднее, сек 95% ДИ, сек
<b>Общее время по всем цифровым паттернам</b>	0,8910 -0,4042 , 2,1864	0,8299 -0,3793, 2,0392	0,0521 +0,0022 -0,0237, 0,1281	0,0518 +0,0022 -0,0232, 0,1270
<b>Общее время по всем паттернам кроме проведения линии</b>	0,36235 0,2820, 0,4426	0,3363 0,2608, 0,4117	0,0211 +0,0022 0,0163, 0,0260	0,0211 +0,0022 0,016, 0,0260

Дополнительно были представлены временные параметры, исключая паттерн движения «проведение линии», поскольку это макродвижение, состоящее из нескольких простых движений. Остальные паттерны движения представляют собой отдельные простые, атомарные движения. Более точ-

ная информация о временных соотношениях иллюстрируется набором из шести цифровых паттернов, учитывающих общее время всех паттернов, кроме проведения линии. Наблюдается отрицательное время в нижнем диапазоне доверительного интервала, если учесть проведение линии. В данном случае время не может быть нулевым, не говоря уже о его отрицательном значении. Выполнение алгоритма потребует некоторого, хоть и минимального, вычислительного времени.

Как показано на рисунке 5.5, при использовании 96 столбцов матрицы движения данные поступают с интервалом в 4,7 миллисекунды. Таким образом, если алгоритм будет обрабатывать пакет данных с меньшим интервалом, возможна обработка в реальном времени. В настоящее время и в рамках данного проекта режим реального времени недоступен, однако существуют определенные улучшения, позволяющие достичь этой цели. В заключительной части работы представлены возможные направления для достижения данной цели.

## 4.4 Выводы по главе

Быстродействие комплекса увеличено от 14 до 17 раз по сравнению с аналогичными решениями без сжатия матрицы с данными о движении.

В этой главе были представлены следующие темы:

1. Проверено влияние методик уменьшения размерности данных на производительность системы контроля движений.
2. Проведены исследования с использованием четырех наборов данных для оценки качества модели движения человека.
3. Применен алгоритм DTW для расчета расстояний между эталонными цифровыми паттернами и записанными действиями.
4. Оценена дискриминируемость классов движений с использованием доверительных интервалов на основе четырех наборов данных.
5. Изучен эффект уменьшения признакового пространства модели движения.
6. Реализована программная составляющая комплекса.

На основе исходного набора данных, содержащего записи семи типов движений, были сформированы четыре дополнительных набора данных, учитывающих нормализацию данных, применение вейвлет-преобразования и использование исходных данных. Эти данные были получены с помощью программно-аппаратного комплекса, описанного во второй главе данной работы. Метод вейвлет-преобразования подробно описан в третьей главе.

Для проверки дискриминируемости эталонных цифровых паттернов движений были построены доверительные интервалы расстояний между ними. Чтобы эталонный паттерн считался корректным, его расстояние до паттернов своего типа должно находиться в пределах 95% доверительного интервала расстояний и не пересекаться с паттернами других типов движений. Только один из четырех исследуемых наборов данных, использующих вейвлет-преобразование и нормализацию данных, удовлетворяет этому условию. В остальных случаях более половины паттернов имеют серые зоны, где их классификация неоднозначна, а без нормализации данных границы доверительных интервалов расположены настолько близко друг к другу, что применение 99% доверительного интервала может привести к пересечению интервалов.

Были проведены измерения скорости работы метода обработки данных и сравнения цифровых паттернов после уменьшения признакового пространства модели движения. В ходе эксперимента использовались статистические данные для контрольной и тестовой выборок. Разработанный метод не только улучшает различение цифровых паттернов, но и позволяет достичь этого гораздо быстрее.

Продукты прошли государственную регистрацию в Роспатенте [14–16]. Для разработки перчатки контроля движений потребовалось спроектировать собственную плату для чипа VM160. Существующие платы не позволяли разместить их на фалангах пальцев.

## Заключение

Аппаратная составляющая системы обеспечивает представление движений в виде цифровых паттернов движений, которые объединяются в матрицу с данными о движении. Программная часть системы включает усовершенствованное вейвлет-преобразование, отбор столбцов матрицы с данными о движении, формирование эталонных движений и сравнение цифровых паттернов движения с использованием алгоритма динамического искажения временной шкалы.

В ходе исследования были решены следующие задачи:

- Разработан программно-аппаратный комплекс на основе инерционных датчиков для сравнения исследуемых и тестовых движений.
- Создана методика сокращения объема первичной информации до 24-48 раз при сохранении точности анализа.
- Проведено сравнение быстродействия разработанного программно-аппаратного комплекса с аналогичными решениями без сжатия матрицы с данными о движении, показавшее его увеличение в 14-17 раз.

В аппаратной части системы была использована определенная физиологическая топология узлов, спроектированная с помощью инструментов проектирования печатных плат и логического анализатора. 16 датчиков были расположены на кисти руки с учетом синхронизации получения данных по протоколу SPI. Применение правильной конфигурации линии CS позволило получить необходимый набор данных.

В программной части проекта ключевую роль играли корреляционный анализ, вейвлет-преобразование и алгоритм динамического искажения временной шкалы. Была разработана методика правильного сочетания столбцов матрицы движения и определения подходящего масштаба вейвлет-пре-

образования. Были созданы обобщенные цифровые паттерны для каждого типа движения на основе контрольной и тестовой выборок. Были обнаружены сильные линейные связи между столбцами матрицы движения, определен уровень избыточности в моделях движения, разработан алгоритм сокращения признакового пространства, проведено сравнение степени схожести цифровых паттернов и оценено различие между эталонными цифровыми паттернами.

В целом, для системы важно было работать с данными, сравнивая цифровые паттерны без использования 3D-реконструкции движения. Это позволило избежать ошибок дрейфа нулевого уровня датчиков, связанных с накопительными интегральными ошибками, вызванными дрейфом нуля гироскопа и дребезгом значений акселерометра. Существующие алгоритмы, направленные на устранение этих эффектов, либо требуют значительных вычислительных ресурсов, либо не решают проблему для макродвижений. В рамках данного проекта необходимо учитывать как микро-, так и макродвижения на устройствах с ограниченной вычислительной мощностью.

## **Основные результаты**

Все поставленные в начале исследования задачи были решены. В частности, был разработан программно-аппаратный комплекс для контроля движений кисти человека, что позволило получить следующие новые результаты:

1. Проведен анализ алгоритмов преобразования и сравнения цифровых паттернов движений, взятых с МЭМС, в сфере захвата движений.
2. Разработан программно-аппаратный комплекс, основанный на использовании МЭМС, для сравнения исследуемых и тестовых движений. Проанализирована электрическая схема аппаратной части системы, что позволило разработать авторские печатные платы для сенсоров ВМ160.
3. Разработана методика сокращения объема первичной информации до



24-48 раз при сохранении точности результатов ее анализа. Представлен способ сортировки столбцов матрицы с данными о движении для увеличения компактности матрицы. Сформированы 7 эталонных цифровых паттернов действий.

4. Реализованная система с применением всех методик позволяет увеличить быстродействие системы от 14 до 17 раз по сравнению с аналогичными решениями без сжатия матрицы с данными о движении.

### **Рекомендации, перспективы дальнейшей разработки темы**

1. Разработать логику для микроконтроллера, включающую алгоритмы обратной связи для идентификации и отсека неиспользуемых столбцов в матрице данных о движении. Это позволит минимизировать избыточные данные путём их обработки на уровне микроконтроллера.
2. Внедрить на микроконтроллере вейвлет-преобразование и алгоритм динамического искажения времени, обеспечивая тем самым частичное выполнение вычислений на одноплатном компьютере.
3. Оптимизировать алгоритм динамического временного выравнивания для более эффективного сравнения цифровых паттернов движения. Улучшения могут включать в себя внедрение C-расширений для повышения производительности, параллельную обработку данных для ускорения вычислений и применение ассоциативных массивов для хранения и доступа к данным.

# Литература

- [1] Gajniyarov Igor, Obabkov Ilya, Khlebnikov Nikolai. Motion capture in educational robotics // AIP Conference Proceedings / AIP Publishing. Т. 1886. 2017. С. 020085.
- [2] The Motion Capture as Behavior Analyzing Method of Spontaneous Motor Activity / Igor Gajniyarov, Igor Mikhailov, Ilya Starodubtsev [и др.] // 2019 SIBIRCON / IEEE. 2019. С. 681–684.
- [3] Animatronic Hand Model on the Basis of ESP8266 / Ilya Starodubtsev, Igor Gajniyarov, Rustam Samedov [и др.] // 2019 SIBIRCON / IEEE. 2019. С. 500–503.
- [4] The Tasks of Designing and Developing Virtual Test Stands / Vladimir L Averbukh, Natalya V Averbukh, Pavel Vasev [и др.] // 2020 Global Smart Industry Conference (GloSIC) / IEEE. 2020. С. 49–54.
- [5] Averbukh Vladimir L, Averbukh Natalya V, Gajniyarov Igor. Problems Arising in the Design of Workstations Based on Augmented Reality // 2020 Global Smart Industry Conference (GloSIC) / IEEE. 2020. С. 55–59.
- [6] М. Гайниyarов И. Уменьшение признакового пространства моделей движения кисти человека. Т. 71. 2021.
- [7] Гайниyarов И. М. Обабков И.Н. Метод снижения размерности данных о движении, основанный на вейвлет–преобразовании. Т. 72. 2021.
- [8] Гайниyarов И.М., Морозов А.А. Сравнение алгоритмов фильтрации сырых данных для маркерной киберфизической системы захвата движений // Молодой ученый. 2017. № 16. С. 192–195.
- [9] Гайниyarов И.М., Морозов А.А. Метод захвата движений как средство естественного интерфейса // Графикон. 2017. № 16. С. 193–196.
- [10] Гайниyarов И.М., Морозов А.А. Технология захвата движений и образовательные роботы // ФТИ-2017 / Издательский центр УрФУ. 2017. С. 127–128.
- [11] Исследование мелкой моторики в VR / И.М Гайниyarов, Э.Р. Валиева, А.А. Данин [и др.]. 2019.
- [12] Аниматронная модель руки на базе ESP8266 / ИС Стародубцев, РФ Самедов, ИМ Гайниyarов [и др.]. 2019.

- [13] Гайниязов Игорь Мадыхатович, Стародубцев Илья Сергеевич. Построение модели движения кисти руки на базе микроэлектромеханических сенсоров и алгоритма динамической трансформации временной шкалы // Интеллектуальные системы в науке и технике. Искусственный интеллект в решении актуальных социальных и экономических проблем XXI века. 2020. С. 165–170.
- [14] Свидетельство о государственной регистрации программы для ЭВМ №2017617823. Обработка данных с трехосевого гироскопа/акселерометра фильтром Калмана, настроенного под Arduino MEGA 2560 /Правообладатель УрФУ, Авторы Гайниязов И. М., Серков К.В, Морозов А.А. – заявка №2017614693; дата государственной регистрации в Реестре программ для ЭВМ 12.07.2017.
- [15] Свидетельство о государственной регистрации программы для ЭВМ №2019611071. Прошивка ESP8285 для сбора данных с 18 ВМ160 и дальнейшей передачи POST методом /Правообладатель УрФУ, авторы Гайниязов И. М., Серков К.В, Обабков И.Н., Зозуля А.А. – заявка №2018665481; дата государственной регистрации в Реестре программ для ЭВМ 28.12.2018.
- [16] Свидетельство о государственной регистрации программы для ЭВМ №2020662364. Визуализатор потоковых CSV данных GY-ВМ160 из UDP пакетов /Правообладатель ИММ УрО РАН, авторы Гайниязов И. М. – заявка №2020662364; дата государственной регистрации в Реестре программ для ЭВМ 12.10.2020.
- [17] Свидетельство о государственной регистрации программы для ЭВМ №2021664101. Анализ шумов в МЭМС сенсорах с помощью вариации Аллана /Правообладатель ИММ УрО РАН, авторы Гайниязов И. М. – заявка №2021663218; дата государственной регистрации в Реестре программ для ЭВМ 31.08.2021.
- [18] Свидетельство о государственной регистрации программы для ЭВМ №2022667983. Визуализатор двигательных паттернов после вейвлет-преобразования /Правообладатель Федеральное государственное бюджетное учреждение науки Институт математики имеханики им. Н.Н. Красовского Уральскогоотделения Российской академии наук, Авторы Гайниязов И. М. – заявка №2022667084; дата государственной регистрации в Реестре программ для ЭВМ 20.09.2022.
- [19] Wavelet-based algorithm for auto-detection of daily living activities of older adults captured by multiple inertial measurement units (IMUs) / Fouaz S Ayachi, Hung P Nguyen, Catherine Lavigne-Pelletier [и др.] // Physiological measurement. 2016. Т. 37, № 3. С. 442.
- [20] Биомеханический анализ движений пальцев травмированной кисти как метод функциональной диагностики / Елена Владимировна Би-

- рюкова, АА Фролов, ИВ Гринягин [и др.] // Вестник травматологии и ортопедии им. НН Приорова. 2010. № 2. С. 70–77.
- [21] Fathia R.N., Batubara S, Safitri D.M. Usulan pengurangan waktu setup menggunakan metode smed serta pengurangan waktu proses produksi dan perakitan menggunakan metode most di pt. panasonic manufacturing Indonesia // Jurnal teknik industri. 2016. Т. 6, № 2.
- [22] The automatic detection of chronic pain-related expression: requirements, challenges and a multimodal dataset / Min Aung, Sebastian Kaltwang, Nick Tyler [и др.] // IEEE transactions computing. 2016. № 1. С. 1–1.
- [23] Mobile human airbag system for fall protection using MEMS sensors and embedded SVM classifier / Guangyi Shi, Cheung Shing Chan, Wen Jung Li [и др.] // IEEE Sensors Journal. 2009. Т. 9, № 5. С. 495–503.
- [24] Human motion video analysis in clinical practice / VV Borzиков, OV Vorobyova, AN Kuznetsov [и др.] // Современные технологии в медицине. 2015. Т. 7, № 4 (eng).
- [25] An unconstrained monitoring system for home rehabilitation / Yasushi Masuda, Mitsuyoshi Sekimoto, Masayuki Nambu [и др.] // IEEE engineering in medicine and biology magazine. 2005. Т. 24, № 4. С. 43–47.
- [26] Ubiquitous rehabilitation center: An implementation of a wireless sensor network based rehabilitation management system / B.P. Jarochowski, S. Shin, D. Ryu [и др.] // Convergence Information Technology / IEEE. 2007. С. 2349–2358.
- [27] Whitchurch A.K., Abraham J.K., Varadan V.K. Design and development of a wireless remote point-of-care patient monitoring system // Technical Conference, 2007 IEEE / IEEE. 2007. С. 163–166.
- [28] A review of wearable sensors and systems with application in rehabilitation / Shyamal Patel, Hyung Park, Paolo Bonato [и др.] // Journal of neuroengineering and rehabilitation. 2012. Т. 9, № 1. С. 21.
- [29] Soft robotic glove for combined assistance and at-home rehabilitation / Panagiotis Polygerinos, Zheng Wang, Kevin C Galloway [и др.] // Robotics and Autonomous Systems. 2015. Т. 73. С. 135–143.
- [30] Karime Ali, Eid Mohamad, Alja'Aml. A fuzzy-based adaptive rehabilitation framework for home-based wrist training // IEEE Transactions on Instrumentation and Measurement. 2013. Т. 63, № 1. С. 135–144.
- [31] Niżnikowski T., Sadowski J., Starosta W. Coordination Abilities in Physical Education, Sports and Rehabilitation. Józef Piłsudski University of Physical Education, Warsaw, 2016.

- [32] Towards miniaturization of a MEMS-based wearable motion capture system / M.N. Brigante, N. Abbate, A. Basile [и др.] // IEEE Transactions on Industrial Electronics. 2011. Т. 58, № 8. С. 3234–3241.
- [33] Development of a biofeedback therapeutic-exercise-supporting manipulator / S. Komada, Y. Hashimoto, N. Okuyama [и др.] // IEEE. 2009. Т. 56, № 10. С. 3914–3920.
- [34] Информационно-измерительная система контроля параметров тренировочного процесса боксеров / Г.И. Мокеев, М.П. Иванов, В.Н. Харрасов [и др.] // Ученые записки университета им. ПФ Лесгафта. 2010. № 4 (62).
- [35] Mon D., Zakynthinaki M.S., Calero S. Connection between performance and body sway/morphology in juvenile Olympic shooters. 2019.
- [36] Ли В.Г., Сапрунов В.Н., Улядуров А.А. Виртуальная среда имитационного моделирования тренировок космонавтов-операторов робототехнических средств МКС // Известия ЮФУ. 2005. Т. 45, № 1.
- [37] Михайлюк М.В., Брагин В.И. Технологии виртуальной реальности в имитационно-тренажерных комплексах подготовки космонавтов // Пилотируемые полеты в космос. 2013. № 2. С. 82.
- [38] Ling Yin. Advanced information processing of MEMS motion sensors for gesture interaction // Sensor Systems. 2016. Т. 5, № 2. С. 419–431.
- [39] Cheng Hong, Yang Lu, Liu Zicheng. Survey on 3D Hand Gesture Recognition. // IEEE Trans. Circuits Syst. Video Techn. 2016. Т. 26, № 9. С. 1659–1673.
- [40] Which Body Would You Like to Have? The Impact of Embodied Perspective on Body Perception and Body Evaluation in Immersive Virtual Reality / Solène Neyret, Anna I Bellido Rivas, Xavi Navarro [и др.] // Frontiers in Robotics and AI. 2020. Т. 7. С. 31.
- [41] Механика человеческих ходунков: анатомически-физиологическое исследование. Dieterich, 1836.
- [42] Гильбрет Ф, Гильбрет Л. Изучение движений, как способ повысить производительность при всякой работы/Пер. 28зд28 28зд28. АВ Панкина, препод // Михайловск. Артил. 28зд2828. и горн. Инж. ЛА Левенстерна. 1913.
- [43] Исследования по биодинамике ходьбы, бега, прыжка / НА Бернштейн, ЛС Осипов, ПИ Павленко [и др.] // М.: Физкультура и спорт. 1940.
- [44] Cavanagh Peter R. Biomechanics of Distance Running. ERIC, 1990.
- [45] Сеченов ИМ. Очерк рабочих движений человека // ИМ Сече-2. 1906.

- [46] 3dID: a low-power, low-cost hand motion capture device / Michele Sama, Vincenzo Pacella, Elisabetta Farella [и др.] // Proceedings of the conference on Design, automation and test in Europe: Designers' forum / European Design and Automation Association. 2006. С. 136–141.
- [47] Flexible CNT-array double helices strain sensor with high stretchability for motion capture / Cheng Li, Ya-Long Cui, Gui-Li Tian [и др.] // Scientific reports. 2015. Т. 5. С. 15554.
- [48] Characterization of wireless accelerometer sensor and its industrial applications / Satish Mohanty, Karunesh Kumar Gupta, Kota Solomon Raju [и др.] // 2014 Twentieth National Conference on Communications (NCC) / IEEE. 2014. С. 1–5.
- [49] Preliminary experiments in motion programming of humanoid robot by human demonstrations / Atsushi Konno, Takahide Yoshiike, Koichi Nagashima [и др.] // JSME International Journal Series C Mechanical Systems, Machine Elements and Manufacturing. 2000. Т. 43, № 2. С. 401–407.
- [50] Методы биомеханики тренажера руки человека / Александр Константинович Платонов, А А Фролов, Елена Владимировна Бирюкова [и др.] // Препринты Института прикладной математики им. МВ Келдыша РАН. 2012. № 0. С. 82–40.
- [51] Синяченко Олег Владимирович. Диагностика и лечение болезней суставов. 2012.
- [52] Князь ВА. Оптическая система захвата движения для анализа и визуализации трехмерных процессов // ГРАФИКОН. 2015. С. 232–236.
- [53] Feasibility of a wearable, sensor-based motion tracking system / Farrokh F Mohammadzadeh, Shijing Liu, Kyle A Bond [и др.] // Procedia Manufacturing. 2015. Т. 3. С. 192–199.
- [54] Callaway Andrew J, Cobb Jon E, Jones Ian. A comparison of video and accelerometer based approaches applied to performance monitoring in swimming // International Journal of Sports Science & Coaching. 2009. Т. 4, № 1. С. 139–153.
- [55] Slyper Ronit, Hodgins Jessica K. Action capture with accelerometers // Proceedings of the 2008 ACM SIGGRAPH/Eurographics Symposium on Computer Animation / Eurographics Association. 2008. С. 193–199.
- [56] Aminian Kamiar, Najafi Bijan. Capturing human motion using body-fixed sensors: outdoor measurement and clinical applications // Computer animation and virtual worlds. 2004. Т. 15, № 2. С. 79–94.
- [57] Moeslund Thomas B, Hilton Adrian, Krüger Volker. A survey of advances in vision-based human motion capture and analysis // Computer vision and image understanding. 2006. Т. 104, № 2-3. С. 90–126.

- [58] Wang Jiang, Liu Zicheng, Wu. Learning actionlet ensemble for 3D human action recognition // IEEE transactions on pattern analysis and machine intelligence. 2014. T. 36, № 5. C. 914–927.
- [59] Kitagawa Naoki, Ogihara Naomichi. Estimation of foot trajectory during human walking by a wearable inertial measurement unit mounted to the foot // Gait & posture. 2016. T. 45. C. 110–114.
- [60] Estimating 3D L5/S1 moments and ground reaction forces during trunk bending / GS Faber, CC Chang, I Kingma [и др.] // Journal of biomechanics. 2016. T. 49, № 6. C. 904–912.
- [61] Roetenberg Daniel, Luinge Henk, Slycke Per. Xsens MVN: full 6DOF human motion tracking using miniature inertial sensors // Xsens Motion Technologies BV, Tech. Rep. 2009. T. 1.
- [62] Toward a human-like biped robot gait: Biomechanical analysis of human locomotion / Bulat Gabbasov, Igor Danilov, Илья Afanasyev [и др.] // 2015 10th International Symposium on Mechatronics and its Applications (ISMA) / IEEE. 2015. C. 1–6.
- [63] Space-time representation of people based on 3D skeletal data: A review / Fei Han, Brian Reily, William Hoff [и др.] // Computer Vision and Image Understanding. 2017. T. 158. C. 85–105.
- [64] Kinematics of the shoulder joint in tennis players / Alexandre Lädemann, Sylvain Chague, FC Kolo [и др.] // Journal of science and medicine in sport. 2016. T. 19, № 1. C. 56–63.
- [65] Shankar Ravi. Flexible smart glove. 2016. US Patent 9,529,433.
- [66] Iervolino Onorio, Meo Michele. Wearable Spiral Passive Electromagnetic Sensor (SPES) glove for sign language recognition of alphabet letters and numbers: a preliminary study // Industrial and Commercial Applications of Smart Structures Technologies 2017 / International Society for Optics and Photonics. T. 10166. 2017. C. 1016607.
- [67] Recognition of sign language with an inertial sensor-based data glove / Kyung-Won Kim, Mi-So Lee, Bo-Ram Soon [и др.] // Technology and Health Care. 2016. T. 24, № s1. C. S223–S230.
- [68] A novel data glove for fingers motion capture using inertial and magnetic measurement units / Bin Fang, Fuchun Sun, Huaping Liu [и др.] // Robotics and Biomimetics (ROBIO), 2016 IEEE International Conference on / IEEE. 2016. C. 2099–2104.
- [69] Low-cost static gesture recognition system using MEMS accelerometers / Ajay K, Ateendra R, Lakshminarasimhan S [и др.] // Global Internet of Things Summit (GIoTS), 2017 / IEEE. 2017. C. 1–6.
- [70] Lan Hai, Chen Ga-Lane, Wang Yu-Bin. Game controller glove. 2011. US Patent App. 12/888,424.

- [71] Acceleration sensing glove (ASG) / John Kangchun Perng, Brian Fisher, Seth Hollar [и др.] // *Wearable Computers*, 1999. Digest of Papers. The Third International Symposium on / IEEE. 1999. С. 178–180.
- [72] Bui The Duy, Nguyen Long Thang. Recognizing postures in Vietnamese sign language with MEMS accelerometers // *IEEE sensors journal*. 2007. Т. 7, № 5. С. 707–712.
- [73] Lee Minkyung, Woo Woontack. ARKB: 3D vision-based Augmented Reality Keyboard. // *ICAT*. 2003.
- [74] Прокопенко Семен Владимирович, Можейко Елена Юрьевна. Возможности восстановления тонкой моторики кисти с использованием сенсорной перчатки у больных, перенесших инсульт // *Сибирское медицинское обозрение*. 2014. № 2 (86).
- [75] Реабилитация больных, перенесших инсульт, с помощью биоинженерного комплекса "интерфейс мозг-компьютер+ экзоскелет- / СВ Котов, ЛГ Турбина, ПД Бобров [и др.] // *Журнал неврологии и психиатрии им. СС Корсакова. Спецвыпуски*. 2014. Т. 114, № 12. С. 66–72.
- [76] On the Effect of Stabilization Methods for Quaternion Invariants on the Uncertainty in Optimization-based Estimation / Fabian Girrbaach, Jeroen D Hol, Raymond Zandbergen [и др.] // *IFAC-PapersOnLine*. 2018. Т. 51, № 25. С. 116–121.
- [77] Яцына Юрий, Гриднев Юрий, Шведко Александр. Сравнительный анализ дискретных фильтров КАЛМАНА и МАДЖВИКА // *Наука и инновации*. 2017. Т. 2, № 168.
- [78] Hou Haiying. Modeling inertial sensors errors using Allan variance. University of Calgary, Department of Geomatics Engineering, 2004.
- [79] Application of the digital signal procession in the MEMS gyroscope de-drift / XunSheng Ji, Shourong Wang, Yishen Xu [и др.] // *Nano/Micro Engineered and Molecular Systems*, 2006. NEMS'06. 1st IEEE International Conference on / IEEE. 2006. С. 218–221.
- [80] Mohr Daniel, Zachmann Gabriel. A survey of vision-based markerless hand tracking approaches // *Preprint submitted to Computer Vision and Image Understanding*. 2013.
- [81] Madgwick Sebastian OH, Harrison Andrew JL, Vaidyanathan Ravi. Estimation of IMU and MARG orientation using a gradient descent algorithm // *Rehabilitation Robotics (ICORR)*, 2011 IEEE International Conference on / IEEE. 2011. С. 1–7.
- [82] Fast complementary filter for attitude estimation using low-cost MARG sensors / Jin Wu, Zebo Zhou, Jingjun Chen [и др.] // *IEEE Sensors Journal*. 2016. Т. 16, № 18. С. 6997–7007.



- [83] Multiple model adaptive complementary filter for attitude estimation / Rahul Kottath, Parag Narkhede, Vipin Kumar [и др.] // *Aerospace Science and Technology*. 2017. Т. 69. С. 574–581.
- [84] Evaluation of AHRS algorithms for inertial personal localization in industrial environments / Estefania Munoz Diaz, Fabian de Ponte Müller, Antonio R Jiménez [и др.] // *Industrial Technology (ICIT), 2015 IEEE International Conference on / IEEE*. 2015. С. 3412–3417.
- [85] The globally optimal reparameterization algorithm: an alternative to fast dynamic time warping for action recognition in video sequences / Thomas W Mitchel, Sipu Ruan, Yixin Gao [и др.] // *2018 15th International Conference on Control, Automation, Robotics and Vision (ICARCV) / IEEE*. 2018. С. 1290–1297.
- [86] Методы биомехатроники тренажёра руки человека / Александр Константинович Платонов, А А Фролов, Елена Владимировна Бирюкова [и др.] // *Препринты Института прикладной математики им. МВ Келдыша РАН*. 2012. № 0. С. 82–40.
- [87] Winter David A. *Biomechanics and motor control of human movement*. John Wiley & Sons, 2009.
- [88] Мазуров Вадим [и др.]. *Болезни суставов*. Litres, 2017.
- [89] Современные технологии в исследовании сложнокоординационных двигательных действий тхэквондо / М.А. Рогожников, С.Е. Бакулев, А.В. Павленко [и др.] // *Ученые записки университета им. ПФ Лесгафта*. 2015. № 2 (120).
- [90] Inertial sensors for performance analysis in combat sports: a systematic review / Matthew TO Worsey, Hugo G Espinosa, Jonathan B Shepherd [и др.] // *Sports*. 2019. Т. 7, № 1. С. 28.
- [91] Сапин МР. *Анатомия человека. Том 2* // М.: Оникс. Т. 21.
- [92] Бернштейн Николай Александрович, Зациорский ВМ, Фейгенберг ИМ. *О ловкости и ее развитии*. Физкультура и спорт, 1991.
- [93] Деятельностный подход при проектировании человеко-компьютерного взаимодействия: На примере медицинских интерфейсов / ВЛ Авербух, НВ Авербух, АВ Наймушина [и др.] // М.: ЛЕНАНД. 2014.
- [94] Wang Hongchuan, Yao Jeffrey, Lim Damon [и др.]. *Real-time measurements for establishing database of sporting apparatus motion and impact parameters*. 2005. US Patent App. 11/051,087.
- [95] Georgakaki. *Spectral analysis and Allan variance calculation in the case of phase noise* // arXiv:1202.6627. 2012.
- [96] Волков НВ, Титов АС. *Анализ погрешностей датчика угловых скоростей* // *Приволжский научный вестник*. 2016. № 12-2 (64). С. 23–27.

- [97] Типы ошибок в инерциальных навигационных системах и методы их аппроксимации / МА Литвин, АА Малюгина, АБ Миллер [и др.] // Информационные процессы. 2014. Т. 14, № 4. С. 326–339.
- [98] Оценка основных характеристик бесплатформенного инерциального блока с использованием вариации Аллана / Д.А. Кутовой, П.В. Ситников, А.А. Федотов [и др.] // Вестник Самарского государственного аэрокосмического университета. 2014. № 1 (43).
- [99] Матвеев ВВ. Инерциальные навигационные системы: Учебное пособие // Тула: изд-во ТулГУ. 2012.
- [100] Allan variance analysis on error characters of MEMS inertial sensors for an FPGA-based GPS/INS system / Xin Zhang, Yong Li, Peter Mumford [и др.] // Proceedings of the International Symposium on GPS/GNSS. 2008. С. 127–133.
- [101] A Digital Mode-Matching Control System Based on Feedback Calibration for a MEMS Gyroscope / Bo Yang, Lei Wu, Chengfu Lu [и др.] // Journal of Sensors. 2019. Т. 2019.
- [102] Auto detection and segmentation of physical activities during a Timed-Up-and-Go (TUG) task in healthy older adults using multiple inertial sensors / Hung P Nguyen, Fouaz Ayachi, Margaux Blamoutier [и др.] // Journal of neuroengineering and rehabilitation. 2015. Т. 12, № 1. С. 36.
- [103] Pellois Robin, Bruls Olivier. Human arm motion tracking using IMU measurements in a robotic environment. 2018.
- [104] Kan Yao-Chiang, Kuo Yu-Chieh, Lin Hsueh-Chun. Personalized Rehabilitation Recognition for Ubiquitous Healthcare Measurements // Sensors. 2019. Т. 19, № 7. С. 1679.
- [105] OxIOD: The Dataset for Deep Inertial Odometry / Changhao Chen, Peijun Zhao, Chris Xiaoxuan Lu [и др.] // arXiv preprint arXiv:1809.07491. 2018.
- [106] INBED: A Highly Specialized System for Bed-Exit-Detection and Fall Prevention on a Geriatric Ward / Nico Jähne-Raden, Ulf Kulau, Michael Marschollek [и др.] // Sensors. 2019. Т. 19, № 5. С. 1017.
- [107] Бриллинджер Д. Временные ряды. Обработка данных и теория. 1980.
- [108] Time series analysis: forecasting and control / George EP Box, Gwilym M Jenkins, Gregory C Reinsel [и др.]. John Wiley & Sons, 2015.
- [109] Колмогоров Андрей, Фомин Сергей. Элементы теории функций и функционального анализа. Litres, 2018. С. 48–49.
- [110] Rabiner Lawrence R. A tutorial on hidden Markov models and selected applications in speech recognition // Proceedings of the IEEE. 1989. Т. 77, № 2. С. 257–286.

- [111] И.М. Газизов. Метод оценки развития газодинамических процессов с помощью скрытой марковской модели. 2016. С. 10–15.
- [112] Zhao Jiaping, Itti Laurent. shapedtw: Shape dynamic time warping // Pattern Recognition. 2018. Т. 74. С. 171–184.
- [113] Шилько СВ, Черноус ДА. Модифицированная биомеханическая модель опорно-двигательной системы человека // Материалы конф. проблемы физического воспитания. 2017. С. 736–742.
- [114] Загrevский Валерий Иннокентьевич, Загrevский Олег Иннокентьевич [и др.]. Моделирование вращения звеньев биомеханической системы вокруг оси с заданными параметрами угловой ориентации. 2017.
- [115] Ma Hao, Liu Hao. Research on Human Motion Recognition System Based on MEMS Sensor Network // 2019 IEEE 4th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC) / IEEE. Т. 1. 2019. С. 2530–2534.
- [116] Пономарев ДИ. Использование алгоритмов обнаружения паттернов для идентификации жестов оператора в записях управляющего сигнала манипулятора // ТРУДЫ МФТИ. 2012. Т. 4, № 3. С. 187–197.
- [117] Exact discovery of time series motifs / Abdullah Mueen, Eamonn Keogh, Qiang Zhu [и др.] // Proceedings of the 2009 SIAM international conference on data mining / SIAM. 2009. С. 473–484.
- [118] Wearable inertial sensor systems for lower limb exercise detection and evaluation: a systematic review / Martin O’Reilly, Brian C, Tomas Ward [и др.] // Sports medicine. 2018. Т. 48, № 5. С. 1221–1246.
- [119] Kos Anton, Umek Anton. Biofeedback Systems in Sport and Rehabilitation // Biomechanical Biofeedback Systems and Applications. Springer, 2018. С. 61–79.
- [120] Classification of the physical training level by heart rate variability and stabilography data / VS Kublanov, DR Yamaliev, AY Dolganov [и др.] // 2017 Siberian Symposium on Data Science and Engineering (SSDSE) / IEEE. 2017. С. 49–54.
- [121] UWB-aided inertial motion capture for lower body 3-D dynamic activity and trajectory tracking / Shaghayegh Zihajehzadeh, Paul K Yoon, Bong-Soo Kang [и др.] // IEEE Transactions on Instrumentation and Measurement. 2015. Т. 64, № 12. С. 3577–3587.
- [122] Stamm Andy, Hartanto Ronny. Feature Extraction from MEMS Accelerometer and Motion Tracking Measurements in Comparison with Smart Bands during Running // Multidisciplinary Digital Publishing Institute Proceedings. Т. 2. 2018. С. 197.
- [123] An adaptive transmission algorithm for an inertial motion capture system in the aspect of energy saving / Michał P, P J, M Aneta Janik [и др.] //

- 2018 Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA) / IEEE. 2018. C. 70–75.
- [124] Lambrecht Joris M, Kirsch Robert F. Miniature low-power inertial sensors: promising technology for implantable motion capture systems // IEEE Transactions on Neural Systems and Rehabilitation Engineering. 2014. T. 22, № 6. C. 1138–1147.
- [125] Micro-IMU-based motion tracking system for virtual training / Yang Zhang, Yunfeng Fei, Lin Xu [и др.] // 2015 34th Chinese Control Conference (CCC) / IEEE. 2015. C. 7753–7758.
- [126] Paulich Monique, Schepers. Xsens MTw Awinda: Miniature wireless inertial-magnetic motion tracker for highly accurate 3D kinematic applications // Xsens: Enschede. 2018.
- [127] Widagdo Chung-Hsien. Limb motion tracking with inertial measurement units // 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC) / IEEE. 2017. C. 582–587.
- [128] Maier Alexander, Sharp Andrew, Vagapov Yuriy. Comparative analysis and practical implementation of the ESP32 microcontroller module for the internet of things // 2017 Internet Technologies and Applications (ITA) / IEEE. 2017. C. 143–148.

Приложение А. Свидетельство о государственной регистрации программы для ЭВМ

РОССИЙСКАЯ ФЕДЕРАЦИЯ



**СВИДЕТЕЛЬСТВО**  
о государственной регистрации программы для ЭВМ

**№ 2017617823**

**Обработка данных с трехосевого гироскопа / акселерометра  
фильтром Калмана, настроенного под Arduino MEGA 2560.**

Правообладатель: *Федеральное государственное автономное  
образовательное учреждение высшего образования «Уральский  
федеральный университет имени первого Президента России  
Б.Н.Ельцина» (RU)*

Авторы: *Гайниязов Игорь Мадыхатович (RU), Морозов Артём  
Андреевич (RU), Серков Константин Владимирович (RU)*

Заявка № **2017614693**  
Дата поступления **23 мая 2017 г.**  
Дата государственной регистрации  
в Реестре программ для ЭВМ **12 июля 2017 г.**



Руководитель Федеральной службы  
по интеллектуальной собственности

*Г.П. Ислюев* Г.П. Ислюев

Приложение Б. Свидетельство о государственной регистрации программы для ЭВМ

РОССИЙСКАЯ ФЕДЕРАЦИЯ



**СВИДЕТЕЛЬСТВО**  
о государственной регистрации программы для ЭВМ  
**№ 2019611071**

**Прошивка ESP8285 для сбора данных с 18 ВМ1160 и дальнейшей передачи POST методом.**

Правообладатель: *Федеральное государственное автономное образовательное учреждение высшего образования «Уральский федеральный университет имени первого Президента России Б.Н.Ельцина» (RU)*

Авторы: *Гайниyarов Игорь Мадыхатович (RU), Серков Константин Владимирович (RU), Обабков Илья Николаевич (RU), Зозуля Артем Александрович (RU)*



Заявка № **2018665481**  
Дата поступления **28 декабря 2018 г.**  
Дата государственной регистрации  
в Реестре программ для ЭВМ **21 января 2019 г.**

Руководитель Федеральной службы  
по интеллектуальной собственности  
 Г.П. Ильев

Приложение В. Свидетельство о государственной регистрации программы для ЭВМ

РОССИЙСКАЯ ФЕДЕРАЦИЯ



**СВИДЕТЕЛЬСТВО**  
о государственной регистрации программы для ЭВМ

**№ 2020662364**

**ВИЗУАЛИЗАТОР ПОТОКОВЫХ CSV ДАННЫХ  
GY-VM160 ИЗ UDP ПАКЕТОВ**

Правообладатель: *Федеральное государственное бюджетное учреждение науки Институт математики и механики им. Н.Н. Красовского Уральского отделения Российской академии наук (RU)*

Автор: *Гайниязов Игорь Мадыхатович (RU)*

Заявка № **2020617601**  
Дата поступления **23 июля 2020 г.**  
Дата государственной регистрации  
в Реестре программ для ЭВМ **12 октября 2020 г.**



Руководитель Федеральной службы  
по интеллектуальной собственности

 Г.П. Ивлиев

Приложение Г. Свидетельство о государственной регистрации программы для ЭВМ

РОССИЙСКАЯ ФЕДЕРАЦИЯ



**СВИДЕТЕЛЬСТВО**  
о государственной регистрации программы для ЭВМ  
№ 2021664101

**Анализ шумов в МЭМС сенсорах с помощью вариации Аллана**

Правообладатель: **ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ УЧРЕЖДЕНИЕ НАУКИ ИНСТИТУТ МАТЕМАТИКИ И МЕХАНИКИ ИМ. Н.Н. КРАСОВСКОГО УРАЛЬСКОГО ОТДЕЛЕНИЯ РОССИЙСКОЙ АКАДЕМИИ НАУК (RU)**

Автор(ы): **Гайниязов Игорь Мадыхатович (RU)**

Заявка № **2021663218**  
Дата поступления **23 августа 2021 г.**  
Дата государственной регистрации в Реестре программ для ЭВМ **31 августа 2021 г.**



Руководитель Федеральной службы по интеллектуальной собственности



Г.П. Излиев



## Приложение Д. Свидетельство о государственной регистрации программы для ЭВМ

РОССИЙСКАЯ ФЕДЕРАЦИЯ



**RU**      **2022667983**

ФЕДЕРАЛЬНАЯ СЛУЖБА  
ПО ИНТЕЛЛЕКТУАЛЬНОЙ СОБСТВЕННОСТИ  
(12) ГОСУДАРСТВЕННАЯ РЕГИСТРАЦИЯ ПРОГРАММЫ ДЛЯ ЭВМ

Номер регистрации (свидетельства): <a href="#">2022667983</a>	Автор: <b>Гайниаров Игорь</b> Мадыхатович (RU)
Дата регистрации: 29.09.2022	Правообладатель: Федеральное государственное бюджетное учреждение науки Институт математики и механики им. Н.Н. Красовского Уральского отделения Российской академии наук (RU)
Номер и дата поступления заявки: 2022667084 20.09.2022	
Дата публикации: <a href="#">29.09.2022</a>	
Контактные реквизиты: i.m.gainiarov@imm.uran.ru	

Название программы для ЭВМ:  
**ВИЗУАЛИЗАТОР ДВИГАТЕЛЬНЫХ ПАТТЕРНОВ ПОСЛЕ ВЕЙВЛЕТ-ПРЕОБРАЗОВАНИЯ**

### Реферат:

Программа предназначена для визуализации двигательных паттернов после вейвлет-преобразования сырых данных с датчиков МЭМС. Цель использования - отладка работы алгоритмов обработки преобразования данных о движении, полученных с датчиков МЭМС. Программой обеспечивается автоматическая загрузка данных в формате CSV, применяется второй уровень декомпозиции вейвлета Добеши первого порядка и выводится единая визуализация результата. ОС: Windows 11.

Язык программирования: Python

Объем программы для ЭВМ: 3 КБ

## Приложение Е. Листинг программы микроконтроллера

```
1  #include <BMI160Gen.h>
2
3  const char* ssid = "ssid";
4  const char* password = "password";
5  const int udpPort = 8080;
6  const char * udpAddress = "192.168.1.43";
7
8  const int hertz = 215.0;
9  const int sens_num = 16;
10 int bmi_pins[sens_num] = {0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15};
11 float timeout = 1000000.0/hertz;
12 int old_time = 0;
13
14 void setup() {
15     start_OTA();
16     start_BMI160();
17 }
18
19 void loop() {
20     udp.beginPacket(udpAddress, udpPort);
21     int gxRaw, gyRaw, gzRaw, axRaw, ayRaw, azRaw;
22
23     for (int i = 0; i < sens_num; i++){
24         BMI160.spi_ss = bmi_pins[i];
25         BMI160.readMotionSensor(axRaw, ayRaw, azRaw, gxRaw, gyRaw, gzRaw);
26         udp.printf("%d,%d,%d,%d,%d,%d",
27                 gxRaw, gyRaw, gzRaw, axRaw, ayRaw, azRaw);
28     }
29
30     udp.endPacket();
31 }
32
33 void start_BMI160(){
```

```

34   for (int i = 0; i < sens_num; i++){
35       BMI160.begin(BMI160GenClass::SPI_MODE, bmi_pins[i]);
36       BMI160.setGyroRate(1600);
37       BMI160.setGyroRange(500);
38       BMI160.setAccelerometerRate(1600);
39       BMI160.setAccelerometerRange(8);
40       BMI160.autoCalibrateGyroOffset();
41   }
42 }
43
44 void start_OTA() {
45     WiFi.disconnect(true);
46     WiFi.onEvent(WiFiEvent);
47     WiFi.mode(WIFI_STA);
48     WiFi.begin(ssid, password);
49
50     while (WiFi.waitForConnectResult() != WL_CONNECTED) {
51         delay(5000);
52         ESP.restart();
53     }
54 }
55
56 void WiFiEvent(WiFiEvent_t event){
57     switch(event) {
58         case SYSTEM_EVENT_STA_GOT_IP:
59             udp.begin(WiFi.localIP(), udpPort);
60             break;
61
62         case SYSTEM_EVENT_STA_DISCONNECTED:
63             break;
64     }
65 }

```

## Приложение Ж. Листинг подпрограммы приема данных с микроконтроллера

```
1  import socket
2  import time
3  import os
4  import winsound
5  import math
6  import numpy
7  import pandas as pd
8
9  UDP_PORT, UDP_IP = 8080, ''
10 SOCK = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
11 SOCK.bind((UDP_IP, UDP_PORT))
12
13 TIME_FN = str(round(time.time()))
14 HZ_FILENAME = '' # 'hz_data_' + TIME_FN + '.csv'
15 DATA_FILENAME = '' # 'all_data_' + TIME_FN + '.csv'
16 GENERAL_PATH = '' # 'D:\\YandexDisk\\1_Practice\\3_Python\\2
    _Working_dir\\'
17
18
19 def herz_count(ch_vibor=math.inf):
20     '''It is count the incoming data in Hz'''
21     try:
22         data_count = 0
23         srednee = []
24         sbor = 0
25         while sbor < ch_vibor:
26             old_t = time.localtime()[5]
27             data = SOCK.recvfrom(4000)[0]
28             if (time.localtime()[5] - old_t) == 1:
29                 sbor += 1
30                 print(round((100.0*sbor)/ch_vibor, 2), '%', data_count)
31                 srednee.append(data_count)
```

```

32     data_count = 0
33     data_count += 1
34     print('\nMean Hz: ', round(numpy.mean(srednee [1:]), 2))
35     print('\n', data)
36 except KeyboardInterrupt:
37     print('\nConection is closed by user now')
38     print('\nMean Hz: ', round(numpy.mean(srednee [1:]), 2))
39     print('\n', data)
40
41
42 def save_d(ch_vibor=math.inf):
43     ''' It is saving both Hz data and every udp packets '''
44     ds_path = GENERAL_PATH # + 'Data_saving'
45     hzs_path = GENERAL_PATH # + 'Hz_testing'
46     input_variable = input('Enter file name: ')
47     hz_filename_u = input_variable + '_' + HZ_FILENAME + '.csv'
48     data_filename_u = input_variable + '_' + DATA_FILENAME + '.csv'
49     ds_csv_file = open(os.path.join(ds_path, data_filename_u), 'a')
50
51     try:
52         data_count = 0
53         srednee = []
54         sbor = 0
55         while sbor < ch_vibor:
56             old_t = time.localtime()[5]
57             data = SOCK.recvfrom(4000)[0]
58             if (time.localtime()[5] - old_t) == 1:
59                 sbor += 1
60                 print(round((100.0*sbor)/ch_vibor, 2), '%', data_count)
61                 srednee.append(data_count)
62                 data_count = 0
63                 data_count += 1
64                 ds_csv_file.write(data.decode('utf-8') + '\n')
65             ds_csv_file.close()
66
67             print('\nMean Hz: ', round(numpy.mean(srednee [1:]), 2))
68             print('\n', data)
69             print('\nThis saving was done')
70             winsound.Beep(1000, 100)
71 except KeyboardInterrupt:
72     print('\nConection is closed now')

```

### Приложение 3. Листинг подпрограммы обработки и визуализации данных

```
1  import os
2  import sys
3  import pandas as pd
4  import seaborn as sns
5  import pywt
6  import pywt.data
7  import time
8  import pickle
9  import matplotlib.pyplot as plt
10 import numpy as np
11 import matplotlib
12 from matplotlib import cm
13 import math as m
14 from matplotlib.ticker import LinearLocator, FormatStrFormatter
15 import matplotlib.collections as collections
16 from mpl_toolkits.mplot3d import Axes3D
17 import rpy2.robjjects.numpy2ri
18 from rpy2.robjjects.packages import importr
19 rpy2.robjjects.numpy2ri.activate()
20 R = rpy2.robjjects.r
21 DIW = importr('dtw')
22
23
24 W_MODE = ['', 'aaaa']
25
26
27 class Wavelet():
28     def __init__(self, filename):
29         self.filename = filename
30         self.sorted_columns = []
31         self.dataframe = []
32
```

```

33 def column_generator(self):
34     for flow_type in enumerate(['gxRaw_', 'gyRaw_', 'gzRaw_',
35                               'axRaw_', 'ayRaw_', 'azRaw_']):
36         temp = []
37         for i in range(1, 7):
38             temp.append(flow_type[1] + str(i))
39         self.sorted_columns.append(temp)
40
41 def resocolum(self):
42     sep_df = [None]*6
43     temp_df = pd.DataFrame()
44     columns = self.sorted_columns
45     for i in range(6):
46         sep_df[i] = pd.read_csv(self.filename, usecols=columns[i],
47                               compression='bz2')
48     temp_df = pd.concat([sep_df[0], sep_df[1], sep_df[2],
49                        sep_df[3], sep_df[4], sep_df[5]], axis=1)
50     self.dataframe = temp_df # [:32]
51     return self.dataframe
52
53 def read_raw_csv(self):
54     self.dataframe = pd.read_csv(self.filename, compression='bz2').
55         drop('time', axis='columns')
56     return self.dataframe
57
58 def read_raw_pkl(self):
59     self.dataframe = pd.read_pickle(self.filename)
60     return self.dataframe
61
62 def wavelet_init(self):
63     wp = pywt.WaveletPacket2D(data=self.dataframe, wavelet='db1',
64                               mode='symmetric')
65     self.dataframe = wp
66
67 def vis3d(self):
68     w_mode = W_MODE # [: self.dataframe.maxlevel]
69     for wi in range(len(w_mode)):
70         df = self.dataframe[w_mode[wi]].data
71         shape_0 = df.shape[0]
72         shape_1 = df.shape[1]
73         if wi == 0:

```

```

73     shape_21 = max(df.max())
74     shape_22 = min(df.min())
75     else:
76         shape_21 = df.max()
77         shape_22 = df.min()
78     fig = plt.figure()
79     ax = fig.add_subplot(111, projection='3d')
80     matplotlib.rcParams.update({'font.serif': 'Times New Roman',
81                                'font.size': 10.0,
82                                'axes.labelsize': 'Medium',
83                                'axes.labelweight': 'normal',
84                                'axes.linewidth': 0.8,
85                                'figure.figsize': [9, 6]
86                                })
87     x = np.linspace(0, df.shape[1], df.shape[1])
88     y = np.linspace(0, df.shape[0], df.shape[0])
89     X, Y = np.meshgrid(x, y)
90     Z = df
91     surf = ax.plot_surface(X, Y, Z, rstride=1, cstride=1,
92                            color='yellow', alpha=0.7, cmap=cm.Spectral,
93                            linewidth=0, antialiased=True)
94     ax.set_xlim3d(-shape_1/2, shape_1*1.5)
95     ax.set_ylim3d(-shape_0/8, shape_0*1.5)
96     ax.set_zlim3d(shape_22*1.3, shape_21)
97     ax.w_zaxis.set_major_locator(LinearLocator(5))
98     ax.w_zaxis.set_major_formatter(FormatStrFormatter('%0.2f'))
99     fig.colorbar(surf, shrink=0.7, aspect=7)
100    ax.contour(X, Y, Z, zdir='z',
101              offset=shape_22*1.3, cmap=cm.coolwarm)
102    ax.contour(X, Y, Z, zdir='x',
103              offset=-shape_1/2, cmap=cm.coolwarm)
104    ax.contour(X, Y, Z, zdir='y', offset=shape_0*1.5,
105              cmap=cm.coolwarm)
106    ax.set_xlabel('Data flow')
107    ax.set_ylabel('Index')
108    ax.set_zlabel('Raw int')
109    ax.set_title(w_mode[wi] + '_shape_' + str(df.shape))
110    fig.tight_layout()
111    plt.savefig(self.filename[:-4] + '_' + w_mode[wi] + '
112                _wlet_3D_unsorted.png',
                bbox_inches='tight', dpi=200)

```



```

113     plt.clf()
114     plt.close()
115
116 def graths(self):
117     fig, axs = plt.subplots(nrows=2, ncols=3, figsize=(12, 12),
118                             subplot_kw={'xticks': [], 'yticks': []})
119     w_mode = W_MODE[:self.dataframe.maxlevel]
120     for ax, interp_method in zip(axs.flat, w_mode):
121         df = self.dataframe[interp_method].data
122         ax.set_title(str(interp_method) + '_shape_' + str(df.shape))
123         ax.legend([str(x) for x in range(df.shape[1])])
124         ax.set_xlabel('Data frame')
125         ax.set_ylabel('Wavelet value')
126         ax.plot(df)
127     fig.tight_layout()
128     plt.savefig(self.filename[:-4] + '_wlet_graths.png',
129                 bbox_inches='tight', dpi=400)
130     plt.clf()
131     plt.close()
132
133 def only_aaaa_vvvv_wavelet(self, min_v=13, max_v=20):
134     df = self.dataframe['vvv'].data
135     df = df[:, :-1]
136     fig, ax = plt.subplots()
137
138     ax.plot(df)
139     ax.set_xlabel('Data frame')
140     ax.set_ylabel('Wavelet value')
141     ax.set_title('vvv_' + str(df.shape))
142
143     t = np.arange(min_v, max_v)
144     temp_df = pd.Series(df[:, 0])
145     collection = collections.BrokenBarHCollection.span_where(t,
146                                                             ymin=df.min(), ymax=df.max(), where=temp_df < m.inf,
147                                                             facecolor='green', alpha=0.2)
148     ax.add_collection(collection)
149
150     ax.legend([str(x) + ' dataflow' for x in range(df.shape[1])],
151              bbox_to_anchor=(1.02, 0.7))
152     ax.grid()
153     # plt.tick_params(labelsize=2)

```

```

154 plt.xticks(range(df.shape[0]))
155 plt.savefig(self.filename[:-4] + '_vvv.png',
156             bbox_inches='tight', dpi=400)
157 plt.clf()
158 plt.close()
159
160 df = pd.DataFrame(df)
161 df[min_v:max_v].to_csv('atom_' + self.filename[4:], compression='
162                        bz2',
163                        index=False)
164
165 def speed_mem_test(self):
166     start_time = time.time()
167
168     wp = pywt.WaveletPacket2D(data=self.dataframe, wavelet='db1',
169                               mode='symmetric')
170     self.dataframe = wp
171     df = self.dataframe['vvv'].data
172
173     time_count = (time.time() - start_time)
174     memory_bytes = sys.getsizeof(df)
175
176     print("%s seconds—" % time_count, ' for ',
177           self.filename, ' bytes:', memory_bytes)
178     return (self.filename, time_count, memory_bytes)
179
180 def dataset_to_pic(self):
181     fig, axs = plt.subplots(nrows=2, ncols=7, figsize=(12, 6),
182                            subplot_kw={'xticks': [], 'yticks': []})
183     for ax, interp_method in zip(axs.flat, W_MODE):
184         df = self.dataframe[interp_method].data
185         ax.imshow(df, cmap='viridis')
186         ax.set_title(str(interp_method))
187         plt.xlabel('Data frame')
188         plt.ylabel('Wavelet value')
189         plt.grid()
190     fig.tight_layout()
191     plt.savefig(self.filename[:-4] + '_wlet_pic.png',
192                bbox_inches='tight', dpi=800)
193     plt.clf()
194     plt.close()

```

```

194
195     def recombine_dataset(self):
196         temp_df = pd.DataFrame(self.dataframe)
197         temp_df.to_csv(self.filename[4:], compression='bz2')
198
199
200     def grath_plotter():
201         every_files = [x for x in os.listdir() if x.endswith('.csv')]
202         df_count = len(every_files)
203         for i in range(df_count):
204             unit = Wavelet(every_files[i])
205             unit.column_generator()
206             unit.resocolum()
207
208             #         unit.read_raw_csv()
209             unit.wavelet_init()
210             #         unit.only_aaaa_vvvv_wavelet()
211
212             #         unit.graths()
213             #         unit.dataset_to_pic()
214             unit.vis3d()
215             del unit
216             print(i)
217
218
219     def atom_saver(path, min_v, max_v):
220         unit = Wavelet(path)
221         #         unit.column_generator()
222         #         unit.resocolum()
223         unit.read_raw_csv()
224         unit.wavelet_init()
225         unit.only_aaaa_vvvv_wavelet(min_v, max_v)
226         del unit
227
228
229     def pattern_processing():
230         every_files = [x for x in os.listdir() if x.endswith('_used.csv')]
231         df_count = len(every_files)
232         edf = [None]*df_count
233         for i in range(df_count):
234             unit = Wavelet(every_files[i])

```

```

235     edf[i] = unit.read_raw_csv()
236     col_list = list(edf[0])
237     temp_df = np.zeros((edf[0].shape[0], edf[0].shape[1]))
238     for i in edf:
239         temp_df = temp_df + i[col_list]
240         print(i[col_list].shape)
241     temp_df /= len(col_list)
242     del edf
243     plt.grid()
244     plt.title(every_files[0][: -4] + '_general_pattern')
245     plt.xlabel('Dataframe')
246     plt.ylabel('Wavelet value')
247     plt.plot(temp_df)
248     plt.savefig(every_files[0][: -4] + '_general_pattern.png',
249                 bbox_inches='tight', dpi=800)
250     plt.clf()
251     plt.close()
252     temp_df.to_csv(every_files[0][: -4] + '_general_pattern.csv',
253                  compression='bz2', index=False)
254     return temp_df
255
256
257 def dtw_comparison():
258     every_files = [x for x in os.listdir() if x.endswith('.csv')]
259     df_count = len(every_files)
260     template_nm = [x for x in os.listdir() if x.endswith('ral_pattern.
261                  csv')]
262     results = [None]*df_count*len(template_nm)
263     chetchik = 0
264
265     for k in range(len(template_nm)):
266         template = np.array(pd.read_csv(template_nm[k], compression='bz2'
267         ).dropna())
268         rt, ct = template.shape
269         templateR = R.matrix(template, nrow=rt, ncol=ct)
270
271         for i in range(df_count):
272             query = np.array(pd.read_csv(every_files[i],
273             compression='bz2').dropna())
274             rq, cq = query.shape
275             queryR = R.matrix(query, nrow=rq, ncol=cq)

```

```

274
275     alignment = R.dtw(templateR, queryR, keep=True,
276                       dist_method="Euclidean")
277     dist = alignment.rx('distance')[0][0]
278     print(round(dist, 4), every_files[i][5:-25],
279           '\tVS\t', template_nm[k][5:-25])
280     results[chetchik] = (dist, every_files[i], template_nm[k])
281     chetchik += 1
282
283     print('\n')
284
285     temp_df = pd.DataFrame(results)
286     temp_df.to_csv('comparison_dtw_results.csv',
287                  compression='bz2', index=False)
288     return temp_df
289
290
291 def biomeh_reducing():
292     every_files = [x for x in os.listdir() if x.endswith('.pkl')]
293     df_count = len(every_files)
294     edf = [None]*df_count
295     for i in range(df_count):
296         unit = Wavelet(every_files[i])
297         edf[i] = unit.read_raw_pkl()
298     temp_df = np.zeros((edf[0].shape[0], edf[0].shape[1]))
299     for i in edf:
300         temp_df = temp_df + i
301     temp_df /= len(every_files)
302     temp_df.to_pickle(every_files[0][:6] + '_general_corr_mat.pkl')
303
304
305 def reduce_nms():
306     every_files = [x for x in os.listdir() if x.endswith('.pkl')]
307     df_count = len(every_files)
308     edf = [None]*df_count
309
310     for i in range(df_count):
311         unit = Wavelet(every_files[i])
312         test_df = unit.read_raw_pkl()
313         test_df = test_df[test_df > 0.9]
314         edf[i] = test_df[test_df != 1]

```

```

315     nm_list = list(edf[i])
316     edf[i] = edf[i].to_numpy()
317     index_sv = []
318
319     for indexing in range(96):
320         for columns in range(indexing, 96):
321             if edf[i][indexing, columns] > 0:
322                 index_sv.append(nm_list[columns])
323
324     uniq = sorted(set(index_sv))
325     print(len(uniq), every_files[i])
326
327     output = open(every_files[i][: -4] + '_columns_to_delete.pkl', 'wb
328                  ')
329     pickle.dump(uniq, output, 2)
330     output.close()
331     return uniq
332
333 def srednee_chislo_cv9zey():
334     every_files = [x for x in os.listdir() if x.endswith('.pkl')]
335     df_count = len(every_files)
336     edf = [None]*df_count
337
338     for i in range(df_count):
339         unit = Wavelet(every_files[i])
340         test_df = unit.read_raw_pkl()
341         test_df = test_df[test_df > 0.9]
342         edf[i] = test_df[test_df != 1]
343
344         nm_list = list(edf[i])
345         edf[i] = edf[i].to_numpy()
346         index_sv = []
347         crednee = 0
348
349         for indexing in range(96):
350
351             for columns in range(96):
352                 if edf[i][indexing, columns] > 0:
353                     crednee += 1
354
355     print(round(crednee/96., 4), every_files[i])

```

```

355
356 def first_speed_mem_test():
357     every_files = [x for x in os.listdir() if x.endswith('.csv')]
358     df_count = len(every_files)
359     edf = [None]*df_count
360
361     for i in range(df_count):
362         unit = Wavelet(every_files[i])
363         #         unit.column_generator()
364         #         unit.resocolum()
365         unit.read_raw_csv()
366         edf[i] = unit.speed_mem_test()
367         del unit
368
369     output = open('vvv_speed_test.pkl', 'wb')
370     pickle.dump(edf, output, 2)
371     output.close()
372
373
374 def reduce_dataset():
375     every_files = [x for x in os.listdir() if x.endswith('.csv')]
376     every_pickle = [x for x in os.listdir() if x.endswith('.pkl')]
377     col_for_del = pd.read_pickle(every_pickle[0])
378
379     df_count = len(every_files)
380
381     for i in range(df_count):
382         unit = Wavelet(every_files[i])
383         unit.column_generator()
384         df = unit.resocolum()
385         new_col = [x for x in list(df) if x not in col_for_del]
386         df = df[new_col]
387         df.to_csv('new_' + every_files[i][2:], compression='bz2')
388         print(i)

```

## Приложение И. Листинг подпрограммы снижения избыточности данных

```
1  import os
2  import pickle
3  import pandas as pd
4  import seaborn as sns
5  import matplotlib.pyplot as plt
6
7
8  NM = ['gxRaw_1', 'gxRaw_2', 'gxRaw_3', 'gxRaw_4', 'gxRaw_5', 'gxRaw_6',
9        'gxRaw_7', 'gyRaw_1', 'gyRaw_2', 'gyRaw_3', 'gyRaw_4', 'gyRaw_5',
10       'gyRaw_6', 'gyRaw_7', 'gzRaw_1', 'gzRaw_2', 'gzRaw_3', 'gzRaw_4',
11       'gzRaw_5', 'gzRaw_6', 'gzRaw_7', 'axRaw_1', 'axRaw_2', 'axRaw_3',
12       'axRaw_4', 'axRaw_5', 'axRaw_6', 'axRaw_7', 'ayRaw_1', 'ayRaw_2',
13       'ayRaw_3', 'ayRaw_4', 'ayRaw_5', 'ayRaw_6', 'ayRaw_7', 'azRaw_1',
14       'azRaw_2', 'azRaw_3', 'azRaw_4', 'azRaw_5', 'azRaw_6', 'azRaw_7']
15
16
17  class data_analyzer():
18      def __init__(self, filename, porog=0.9, save=True, sep_mat=True,
19                  proc=True, col=NM):
20          self.porog = porog
21          self.save = save
22          self.sep_mat = sep_mat
23          self.proc = proc
24          self.filename = filename
25          self.data = []
26          self.col = col
27
28      def open_pkl_corr_mat(self):
29          input_pkl = open(self.filename, 'rb')
30          matrix = pickle.load(input_pkl)
31          matrix = matrix[NM]
```



```

32
33     if self.proc is True:
34         up_diap = self.porog
35         down_diap = -self.porog
36         self.data = matrix.applymap(lambda x: x > up_diap or x <
37             down_diap)
38     else:
39         self.data = matrix
40
41     input_pkl.close()
42
43 def open_csv_corr_mat(self):
44     # matrix = pd.read_csv(self.filename, usecols=self.col,
45     #                       compression='bz2')
46     matrix = pd.read_csv(self.filename, usecols=self.col)
47     self.data = matrix.corr() # pearson
48     output = open(self.filename[:-4] + '_corr_coef.pkl', 'wb')
49     pickle.dump(self.data, output, 2)
50     output.close()
51
52 def vis_and_save(self):
53     sns.set(font_scale=1)
54     f, ax = plt.subplots(figsize=(30, 30))
55     ax.set_title(self.filename[:-4], size=30)
56     sns.heatmap(self.data, annot=True, fmt='0.2g',
57                 annot_kws={"size": 3}, center=0)
58     if self.save is True:
59         plt.savefig(self.filename[:-4] + '_pg_' + str(self.porog) + '.
60             png',
61                 bbox_inches='tight', dpi=300)
62     else:
63         pass
64     plt.clf()
65     plt.close()
66
67 def processing_pkl():
68     every_files = [x for x in os.listdir() if x.endswith('.pkl')]
69     df_count = len(every_files)
70     for i in range(df_count):
71         flname = every_files[i]

```

```
71     testing = data_analyzer(filename=filename)
72     testing.open_pkl_corr_mat()
73     testing.vis_and_save()
74     del testing
75     print(i)
76
77
78 def processing_csv():
79     every_files = [x for x in os.listdir() if x.endswith('.csv')]
80     df_count = len(every_files)
81     for i in range(df_count):
82         filename = every_files[i]
83         testing = data_analyzer(filename=filename)
84         testing.open_csv_corr_mat()
85         testing.vis_and_save()
86         del testing
87         print(i)
88
89
90 processing_csv()
```

## Приложение К. Листинг подпрограммы выделения цифрового паттерна движения

```
1 import pandas as pd
2 import matplotlib.pyplot as plt
3
4 min_v = 54
5 max_v = 64
6 filename = 'wavelet_without_gz7_az7_bd_04.5_v2_s1_.csv'
7 df = pd.read_csv(filename)
8
9 _, ax1 = plt.subplots()
10 ax1.plot(df)
11 plt.show()
12
13 fig, ax = plt.subplots()
14 df_temp = df.copy()
15 df_temp = df_temp[min_v:max_v]
16
17 ax.plot(df_temp)
18 ax.set_xlabel('Data frame')
19 ax.set_ylabel('Wavelet value')
20 ax.set_title(filename)
21
22 ax.legend([str(x) + ' Dataflow' for x in range(df.shape[1])],
23          bbox_to_anchor=(1.02, 0.7))
24 ax.grid()
25
26 plt.savefig('atom_' + filename[:-4] + '.png', bbox_inches='tight',
27           dpi=200)
28 plt.show()
29 df_temp = pd.DataFrame(df_temp)
30 df_temp.to_csv('atom_' + filename, index=False)
```

## Приложение Л. Листинг подпрограммы формирования общего цифрового паттерна движения

```
1 import pandas as pd
2 import math
3 import tkinter as tk
4 import tkinter.filedialog as fd
5 import numpy as np
6 import matplotlib.pyplot as plt
7
8 root = tk.Tk()
9 path = r'C:\Users\kocmoc\PycharmProjects\sup\disser\data\11
   _atom_norm_aa'
10 plt.rcParams['font.family'] = 'Times New Roman'
11 root.withdraw()
12
13 every_files = fd.askopenfilenames(initialdir=path, title='Choose a
   file', filetypes=[('csv', '*.csv')])
14 raw_patterns = []
15
16 for file in every_files:
17     raw_patterns.append(pd.read_csv(file))
18
19 max_shape = max([x.shape[0] for x in raw_patterns])
20 adding_nan = [max_shape - x.shape[0] for x in raw_patterns]
21
22 nanhz = [math.floor(0.7 * (max_shape / (max_shape - x.shape[0]))) if
   x.shape[0] != max_shape
23         else 0
24         for x in raw_patterns]
25 nanhz = [x - 1 if x > 2 else x for x in nanhz]
26
27 for i, x in enumerate(zip(raw_patterns, adding_nan, nanhz)):
28     df, nan_count, nan_hz = x
29     print(nan_count)
```

```

30     for ii in range(nan_count):
31         df.loc[nan_hz * ii + 0.1] = np.nan
32         raw_patterns[i] = df.sort_index().reset_index(drop=True)
33
34     interpolated_patterns = [x.interpolate() for x in raw_patterns]
35
36     col_list = interpolated_patterns[0].columns
37     general_pattern_df = np.zeros((interpolated_patterns[0].shape[0],
38         interpolated_patterns[0].shape[1]))
39     for i in interpolated_patterns:
40         general_pattern_df = general_pattern_df + i
41     general_pattern_df /= len(col_list)
42
43     plt.grid()
44     plt.title(every_files[0][: -4] + '_general_pattern')
45     plt.xlabel('Dataframe')
46     plt.ylabel('Wavelet value')
47     plt.plot(general_pattern_df, alpha=0.4)
48
49     save_path = ''.join(every_files[0].split('/')[ : -1]) + '/'
50     general_pattern_ + every_files[0].split('/')[ -1]
51     plt.savefig(save_path[: -4] + '_general_pattern.png', bbox_inches='
52         tight', dpi=600)
53     general_pattern_df.to_csv(save_path, index=False)
54
55     plt.show()

```

## Приложение М. Листинг подпрограммы отбора признаков

```
1 def pickle_processing():
2     pickles_files = []
3
4     every_files = [f'{full_path}\\{x}' for x in os.listdir(full_path)
5                     if x.endswith('.pkl')]
6     for i in every_files:
7
8         stop_list = np.array([])
9
10        corr_matrix = pd.read_pickle(i)
11        corr_matrix = corr_matrix[corr_matrix >= 0.9]
12        corr_matrix = corr_matrix[corr_matrix != 1]
13
14        for ii in list(corr_matrix):
15            if int(ii) not in stop_list:
16                temp_stop_list = np.nonzero(pd.DataFrame(corr_matrix[ii
17                ]).fillna(0).transpose().to_numpy()[0])[0]
18                stop_list = np.append(stop_list, temp_stop_list)
19
20        filename = i.split('\\')[-1].replace('_corr_coef.pkl', '.csv')
21        pickles_files.append((filename, np.unique(stop_list)))
22
23    return pickles_files
```

## Приложение Р. Акт использования программного обеспечения

УТВЕРЖДАЮ

Директор УГИ УрФУ

Э.Э.Сыманюк

“20” 09 2022 г.

### АКТ О ВНЕДРЕНИИ АППАРАТНО-ПРОГРАММНОЙ СИСТЕМЫ для измерения и сопоставления движений человека с применением инерциальных микроэлектромеханических сенсоров

Настоящий Акт свидетельствует, что аппаратно-программная система для измерения и сопоставления движений человека с применением инерциальных микроэлектромеханических сенсоров GY-VM160, разработанная Гайнирявым И.М., внедрена и используется в лаборатории клинико-поведенческих исследований человека.

В ходе эксплуатации системы подтверждено, что она обладает всеми заявленными возможностями и позволяет производить запись движений человека, формировать усредненную модель конкретного типа движения, классифицировать движения без применения алгоритмов, требующих интегрирования, в условиях технически неподготовленного помещения. Это помогает производить выездной сбор данных о движении и необходимую диагностику.

Начальник отдела по организации научной  
и инновационной работы

Ю.В. Запарий

## **Приложение Н. Аппаратная реализация программно-аппаратного комплекса для измерения и сопоставления движений кисти человека**

### **Требования к работе программно-аппаратного комплекса**

Необходимо учитывать требования минимизации потребления электроэнергии и размера аккумуляторной батареи, при этом обеспечить минимальный вес носимого устройства и не ниже заявленное время автономной работы. Для обеспечения максимально быстрой комплектации пакетов данных и синхронной передачи данных с датчиков на МК необходимо выбрать подходящий протокол передачи данных и обеспечить соответствующий протокол. Способ передачи данных должен быть беспроводным и позволять передавать весь объём данных без ограничений по пропускной способности.

Протокол передачи пакетов с данными должен доставлять пакеты с максимальной скоростью передачи данных, с возможными небольшими потерями. Это необходимо для максимально быстрого запуска классификации движений.

Комплекс контроля движений должна работать в условиях переменчивой и низкой освещенности, так как комплекс предусматривает автономную работу без предварительной подготовки помещения. Требование учёта освещённости накладывает дополнительные требования, без которых можно обойтись. В данном случае, выбор элементной базы в виде МЭМС датчиков уже помогает выполнять данное требование по умолчанию, так как данные датчики рассчитаны на такое требование.

Комплекс должна быть устойчивой к перекрытию конечностей, чтобы в любой момент времени иметь достоверную информацию о совершаемом



Таблица 5.1 – Сравнение разных вычислительных устройств для решения задачи

Название	Габариты, мм x мм x мм	Энергопотребление, ма	Тактовая частота, МГц	Количество ядер	Разрядность процессора, бит
arduino nano	<b>18 × 45 × 5</b>	<b>25</b>	16	1	8
Raspberry Pi Zero 2W	От 65 x 30 x 10	1350	<b>2400</b>	<b>4</b>	<b>64</b>
Raspberry Pi 4 Model B	От 85 x 56 x 17	3 000	<b>1500</b>	<b>4</b>	<b>64</b>
esp32s	<b>От 25.5 × 18 × 4</b>	<b>130</b>	<b>80</b>	<b>2</b>	<b>32</b>
esp8266	<b>От 45 × 30 × 4</b>	170	<b>80</b>	1	<b>32</b>
STM32 ARM	<b>От 53 x 22 x 10</b>	<b>50 + Беспроводной модуль</b>	<b>72</b>	1	<b>32</b>

движении.

Технологии реализации комплекса не должны стеснять самих движений, так как тело будучи биомеханической системой будет компенсировать имеющиеся ограничения атипичными движениями.

### Выбор центрального вычислительного устройства

Для выбора вычислительного устройства необходимо учитывать:

1. Энергопотребление.
2. Вычислительные возможности.
3. Габариты.

В данном случае выбираем ESP32S (см. Табл. 5.1) ввиду того, что из представленных вычислительных устройств он имеет достаточные габариты, чтобы разместиться на тыльной стороне ладони, имеет умеренное энергопотребление, тактовая частота имеет средние характеристики для 32 bit-ных МК и имеет два вычислительных ядра. Это полезно, так как МК имеет встроенные модули для беспроводной передачи данных и на работу этих протоколов тоже необходимы вычислительные ресурсы, которые у esp8266 доходят до 80%.

Одноплатные компьютеры Raspberry Pi хоть и имеют гораздо большие вычислительные мощности, но из-за этого энергопотребление тоже на порядок выше, плюс габариты становятся уже большими, что уже начинает мешать располагать блок на руке. А это мешает самой биомеханике движений.

В данном случае для работы центрального узла комплекса будет достаточно элемента питания на 130 мАч. По размеру и весу такой элемент

Таблица 5.2 – Сравнение протоколов передачи данных от датчиков к МК

Название протокола	Скорость передачи, кбит/с	Адресация	Количество сигнальных линий
i2c	Стандартный режим: 100 Быстрый: 400 Высокоскоростной: 3 482	Каждое устройство должно иметь уникальный адрес.	2
<b>SPI</b>	<b>От 10 240 до 102 400</b>	<b>По линии выбора микросхемы</b>	4

питания будет меньше батарейки типа Крона. Вес батареи составит не более 50 гр с габаритами 25 \* 20 \* 5 мм.

### Выбор протокола передачи данных от датчиков к МК

Всего возможно выбрать из двух вариантов: SPI и I2C. Более подробно параметры сравнения можно увидеть в Табл. 5.2.

В зависимости от выбранного датчика количество уникальных адресов для протокола i2c может достигать 4. В данном комплексе включено 16 сенсоров, а значит для создания уникальных адресов необходимо дополнительное устройство адресации, которое будет вносить дополнительные задержки в работу комплекса. Протокол SPI не имеет таких ограничений и все заявленные сенсоры теоретически могут быть включены в работу комплекса без дополнительных модулей. Так же протокол работает на гораздо больших скоростях, что позволяет без дополнительных оптимизаций отправлять весь полученный набор данных. Скорость переключения в данном случае за счёт отсутствия модуля переадресации датчиков и более высоких скоростей работы самого протокола.

### Выбор беспроводного протокола передачи данных

Максимальное число значений, которое теоретически могут передать датчики среди всех производителей - 3600 значений в секунду.

В идеальных условиях при наличии только одного датчика. Фактическое значение меньше из-за технической реализации (см. в Табл. 5.3.). При увеличении числа сенсоров добавляются накладные вычислительные рас-

ходы на формирование единого пакета с данными для дальнейшей транспортировки по сетевому протоколу.

Таблица 5.3 – Сравнение беспроводных протоколов передачи данных

Название протокола	Теоретическая скорость передачи, МБит	Фактическое количество, пакетов с данными, которые можно передать, в сек
Bluetooth 4	1	60
Wi-Fi n	600	212
ZigBee	0.25	6
NFR24L01+	до 2	50

Фактическое максимальное число значений в секунду, полученных с одного сенсора составляет не более 1416 значений.

Все заявленные протоколы работы настроены на низкоскоростную передачу данных, кроме Wi-Fi. Это незначительно влияет на энергопотребление, поэтому был выбран протокол, обеспечивающий максимальную пропускную способность. В ходе фактической проверки пересылки тестовых данных - предположение подтвердилось.

### **Выбор протокола передачи данных по сети**

Ещё одним узким местом в комплексе может быть сетевой протокол передачи данных. В выбранном беспроводном протоколе передачи данных Wi-Fi выбор лежит между TCP (Transmission Control Protocol (протокол управления передачей)) и UDP (User Datagram Protocol — протокол пользовательских датаграмм).

Комплекс является чувствительным ко времени, так как при задержке одного пакета нельзя его дожидаться, что бы получить следующие данные. Такие паузы недопустимы, так как отправляющее устройство не хранит значения, полученные от выходных каналов сенсоров. Таким образом задержка пакета значит потерю данных в этом промежутке времени, до тех пор пока конечное устройство не получит подтверждения. TCP требует подтверждения передачи в процессе транспортировки, что дополнительно

нагружает центральное обрабатывающее устройство.

В этой связи выбирается протокол UDP. По фактическим данным этот протокол имеет пропускную способность на 30% более высокую, чем TCP.

### Обоснование расположения датчиков

Согласно тепловой карте (см. Рис. 1.3) наиболее полная оцифровка всех движений кисти возможна по 16 точкам, что соответствует биологической структуре кисти человека (см. Рис. 1.5).

Комплекс для контроля движений представляет собой перчатку с закрепленными датчиками на неподвижных участках кисти. Подробнее схема расположения изображена на рис. 5.1

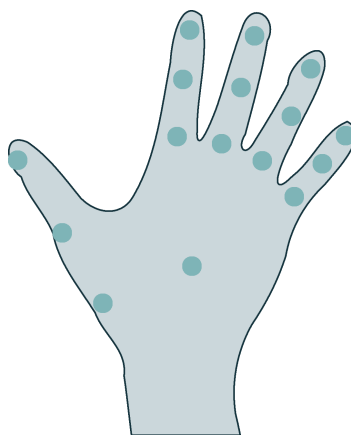


Рисунок 5.1 – Расположение датчиков на перчатке.

С помощью подобного комплекса контроля движений очень удобно оцифровывать непосредственно угловые параметры биомеханической модели кисти и их производные. Однако существует ряд косвенных параметров, которые также можно оцифровать. С помощью акселерометра мы получаем не только угловое ускорение, но и проекцию на вектор гравитации. В связи с этим возможно получить ускорение для прямолинейного движения. Однако стоит отметить два момента. Первый заключается в том, что при одновременном повороте и прямолинейном движении отличить их между собой представляется сложной задачей. Второй состоит в том, что выходной канал  $Z$  у акселерометра показывает отклонение от вертикали. Это

приводит к тому, что одинаковые положения кисти относительно вектора гравитации для акселерометра идентичные (горизонтальные повороты не отражаются на работе).

Также можно выразить расчет силовых показателей при заранее известных значениях массы объекта. В контексте данной работы данное предположение проверяться не будет, так как итоговое применение не предполагает силовых упражнений с дополнительными тренажерами (гантель и др.). Практические исследования тоже касаются в основном упражнений с собственным весом [118, 119].

В контексте техники выполнения упражнений возможен сравнительный анализ движений при выполнении тренировок. Например, приседания: одним из ключевых параметров в этом случае является плотный контакт стопы на протяжении всего выполнения упражнения. Если исключить возможность отслеживать изменения высоты подъема пятки относительно начала выполнения упражнения, то можно обратить внимание, что при неправильном присяде стабильность тела будет ниже, чем при правильном выполнении упражнения. Человеческий глаз не отслеживает данное отличие, но комплекс контроля движений на МЭМС может с этим справиться. Похожие косвенные выводы были сделаны при классификации уровня подготовки спортсменов на основе стабилографических данных [120].

### **Особенности крепления датчиков**

У датчиков данного типа есть ряд удобных преимуществ перед аналогами. Во-первых, они не требуют непосредственного контакта с кожей человека. Во-вторых, они не чувствительны к месту фиксации. Для подробного изучения графики см. рис. 5.2 (для гироскопа) и рис. 5.3 (для акселерометра).

Все это означает, что достаточно попасть на самую целевую фалангу пальца, чтобы получать нужные значения без перекалибровки. В этом случае фаланга выступает в качестве идеальной прямой линии. Для датчика неважно, в какой части фаланги он будет закреплен. Для проверки предположения были взяты два датчика ВМ160 и закреплены в единой

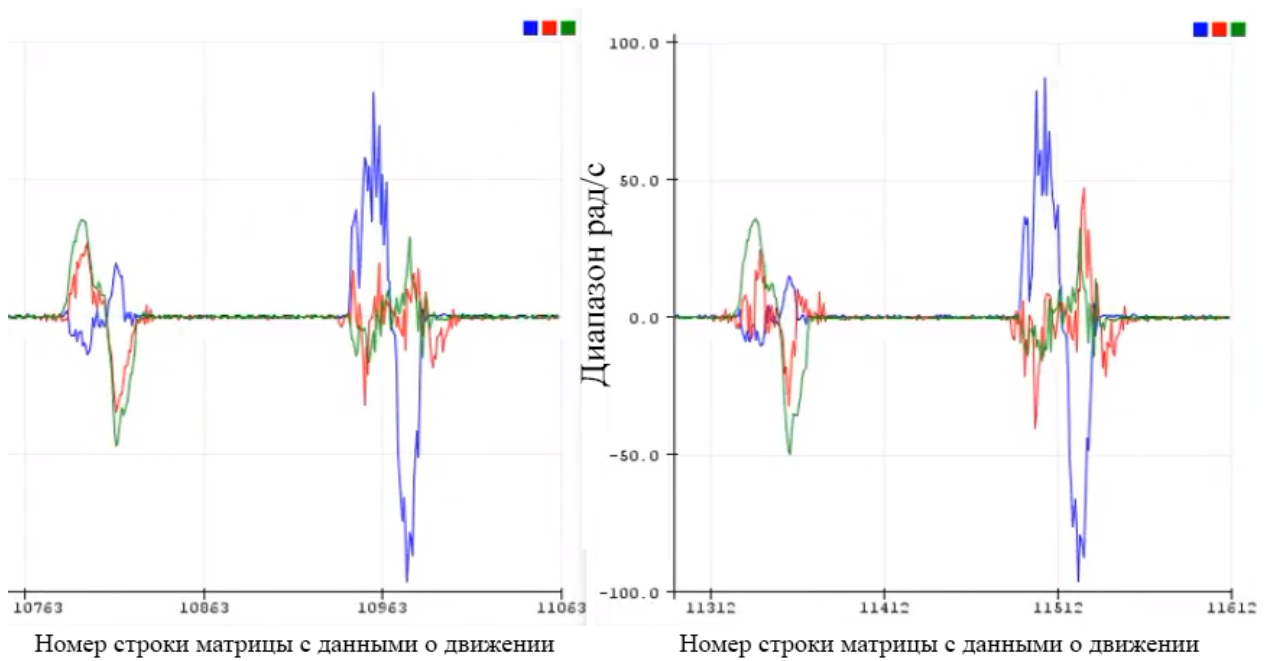


Рисунок 5.2 – Данные с двух гироскопов, закрепленных на жесткой поверхности. Синим отмечена выходной канал X, красным — выходной канал Y, зеленым — выходной канал Z.

ориентации на разных концах макетной платы. Расстояние составило 20 сантиметров. датчики подключили к двум независимым МК; оба были подсоединены к двум USB-портам одного ноутбука. В итоге были получены синхронизированные графики для датчиков.

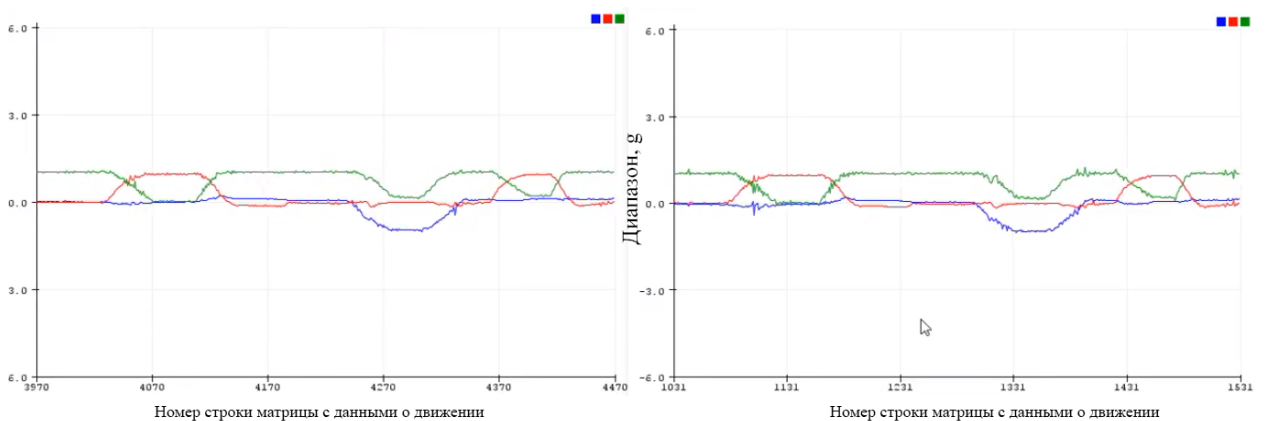


Рисунок 5.3 – Данные с двух акселерометров, закрепленных на жесткой поверхности. Синим отмечена выходной канал X, красным — выходной канал Y, зеленым — выходной канал Z.

В дополнение к благоприятным аспектам следует отметить существова-

ние нюанса, который добавляет дополнительные искажения в получаемые данные. Это ситуация, когда датчик находится близко к суставу. В данном случае добавляются ошибки, связанные с биомеханикой сустава.

## Проверка работы датчиков

Для проверки работы всего комплекса в целом и каждого датчика в отдельности был использован логический анализатор Kingst LA1010, который имеет декодер для шины SPI. Благодаря этому можно наблюдать степень соответствия между данными из технической документации и реальными процессами на линиях SPI. Ниже приведена типичная передача данных. В программе используется функция передачи всех значений сразу, поэтому передача осуществляется 13 байтами для передачи 6 значений для одного датчика. 1 байт — это сопутствующие протоколу данные. Подробнее изображено на рис. 5.4: сиреневым цветом обозначена линия MOSI<sup>14</sup>, синим — MISO<sup>15</sup>, зеленым — Clock<sup>16</sup> и желтым CS<sup>17</sup>.

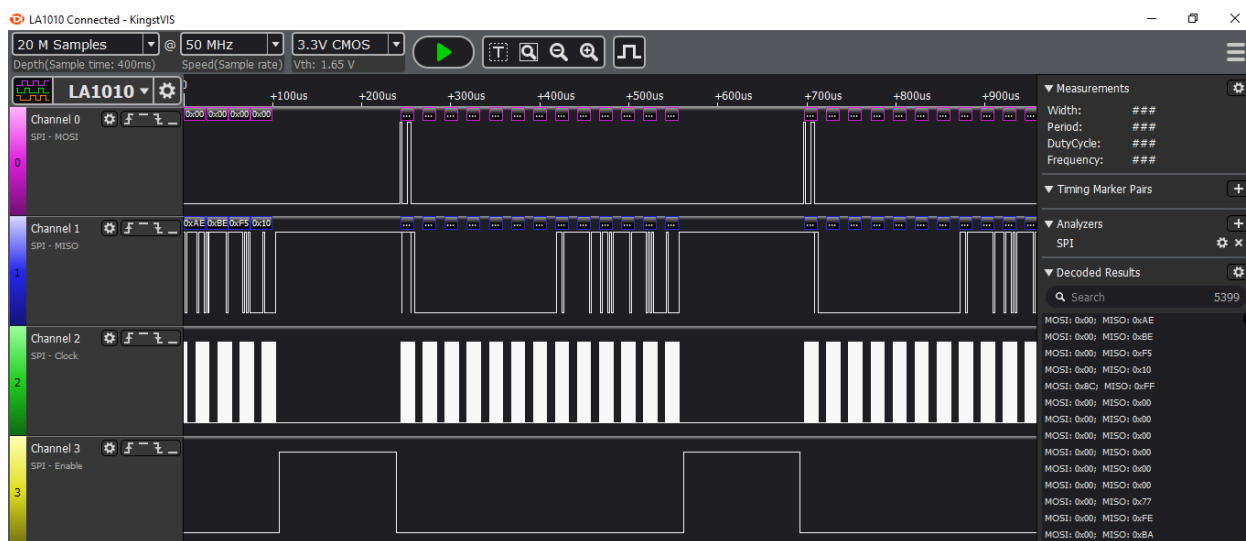


Рисунок 5.4 – Временная диаграмма SPI для BMI160 с Kingst LA1010.

В данном примере была рассмотрена работа одного датчика. Также была зафиксирована скорость работы всего комплекса в зависимости от ко-

<sup>14</sup>(англ.) Master Output Slave Input. Линия передачи от МК к датчику

<sup>15</sup>(англ.) Master Input Slave Output. Линия передачи от датчика к МК

<sup>16</sup>Линия тактирования

<sup>17</sup>(англ.) Chip Select. Выбор микросхемы

личества датчиков.

Были проведены измерительные испытания, зависимости частоты прихода пакетов с данными в зависимости от количества датчиков, подключенных к комплексу. На рис. 5.5 можно заметить, что при работе 4 датчиков частота падает до 46,72% от максимального значения, а при 6 единицах составляет уже менее трети.

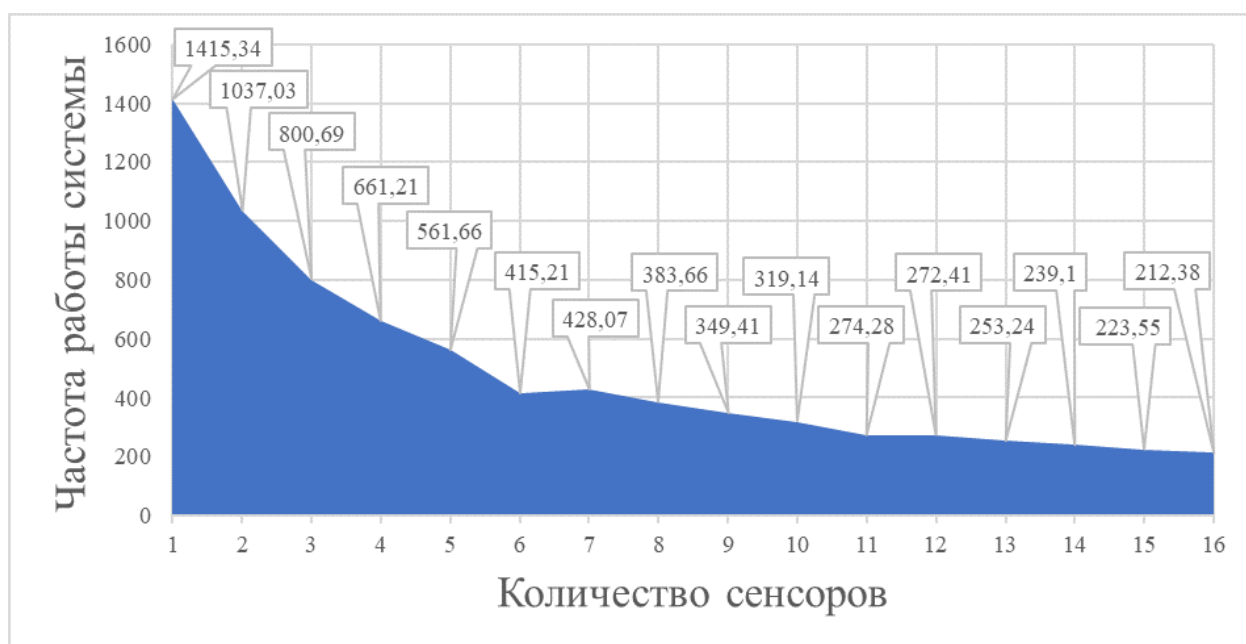


Рисунок 5.5 – Частота прихода данных в зависимости от количества датчиков.

Однако дальнейшее падение частоты замедляется. Это значит, что для существенного увеличения производительности надо сокращать количество используемых датчиков как минимум в 3 раза для преодоления плато с низким увеличением производительности. Полученный результат частоты работы комплекса сопоставим с результатами других исследовательских групп по данному направлению [121–127].

На рис. 5.6 демонстрируются скорости падения частоты приема данных при добавлении одного нового датчика.

Также стоит обратить внимание на точные значения, обозначенные точками. Существуют такие участки, при которых падение частоты либо находится около 0, либо в одном случае имеют отрицательное значение. Это



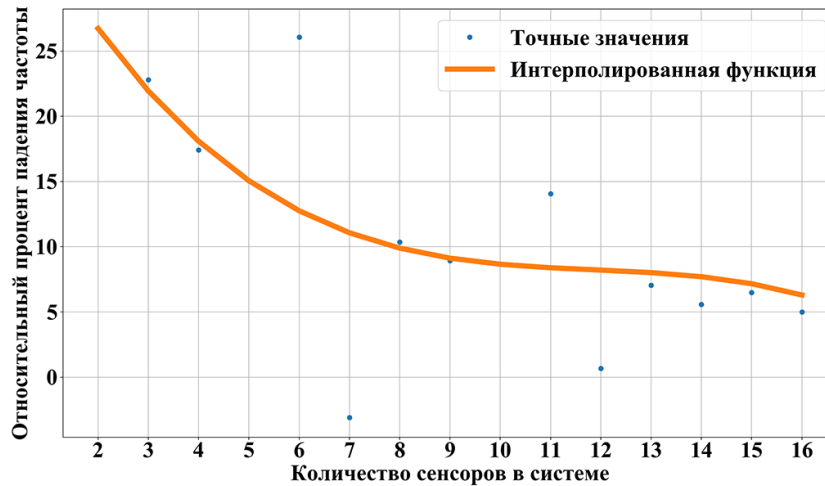


Рисунок 5.6 – Процент падения частоты работы комплекса для текущего числа датчиков по сравнению с конфигурацией, имеющей на 1 датчик меньше.

можно объяснить особенностями работы МК ESP32<sup>18</sup>.

Если говорить про информационную часть комплекса контроля движения кисти, то максимальное число выходных каналов, которое устройство может выдавать без дополнения магнитометров, равняется 12. Каждое значение выходного канала представляется целым числом типа `int` и кодируется двумя байтами.

Выходит, что диапазон необработанных значений составляет от  $-32768$  до  $-32767$ . Для получения правильного результата, следует согласовать его с заданным диапазоном значений.

Таким образом, объем одной строки значений с акселерометра и гироскопа на датчике ВМ160 составляет 24 байта. Это ведет к тому, что объем передаваемых данных в первом приближении можно оценить следующим образом — см. рис. 5.7

Теоретические данные и практическое решение имеет расхождение в части объёма данных, которое объясняется тем, что фактически передаётся не 2 байтовое значение типа `int`, а 8 байтовое типа `char`. Имея хорошую производительность протоколов, можно успешно обрабатывать большие объёмы данных. Таким образом удаётся создать единую строчку для матрицы

<sup>18</sup>[www.espressif.com/sites/default/files/documentation/esp32\\_datasheet\\_en.pdf](http://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf)

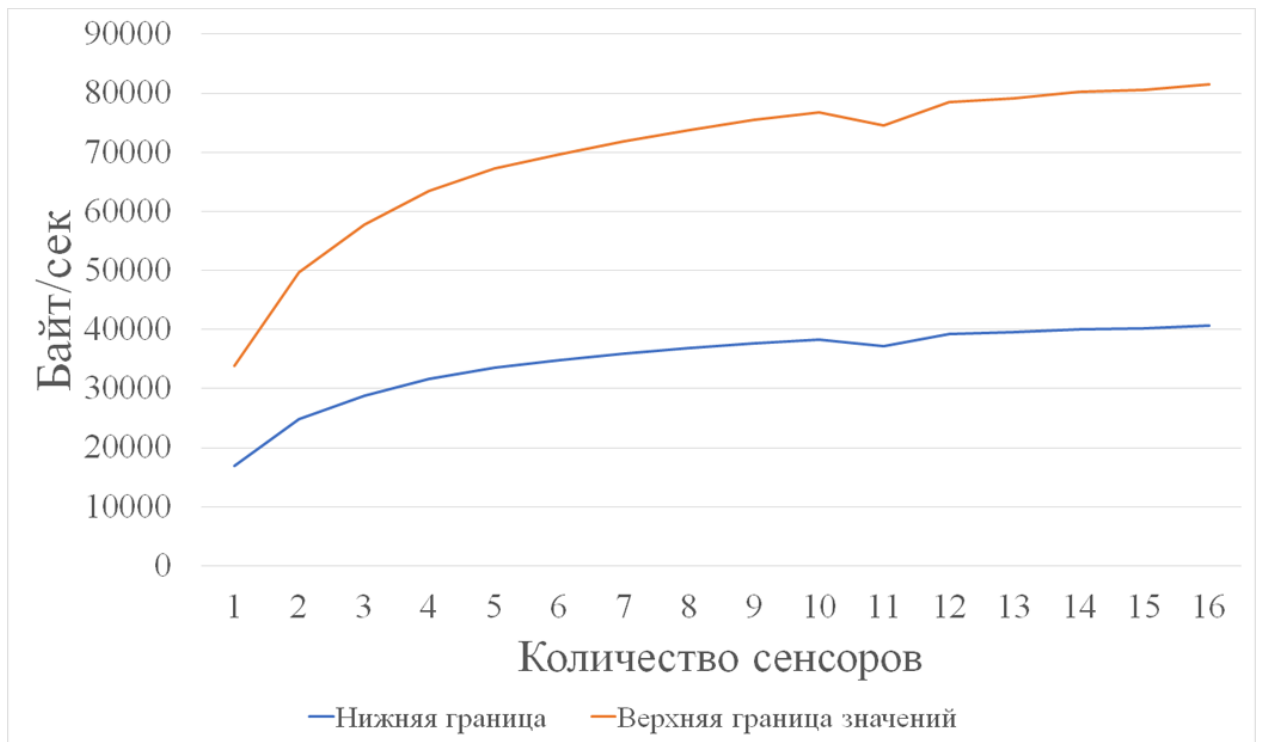


Рисунок 5.7 – Объем данных в зависимости от числа используемых датчиков.

с данными о движении, полученной со всех 16 датчиков.

Была собрана статистика стабильности получения UDP-пакетов на клиентской части. Для этого был записан 6-часовой набор данных, содержащий данные покоя комплекса. Разброс представлен на рис. 5.8.

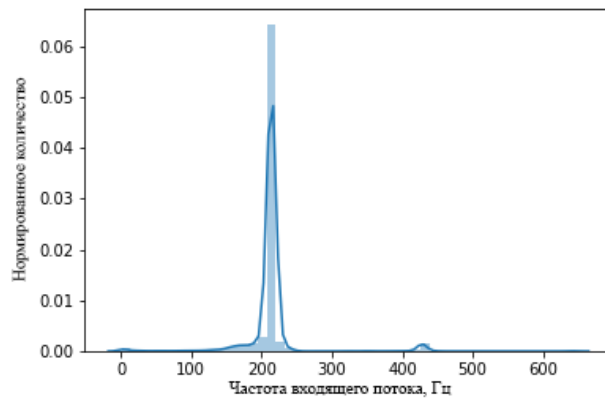


Рисунок 5.8 – Статистика стабильности частоты входящих пакетов данных.

Подавляющее число значений колеблется вокруг среднего значения. Можно заметить некоторые выбросы, равные удвоенному среднему значению. Для поиска причины была написана программа, которая генерирует син-

тетические пакеты, подобные комплексу контроля движений на частоте 6 000 Гц с использованием localhost узла. Эффект с удвоением пакетов повторился, несмотря на смену всех частей, кроме способа передачи через UDP-пакеты. Следовательно, это особенности протокола UDP.

## Характеристики аппаратной части

Все части программ были выполнены на языке Python 3.7 с использованием библиотек pandas, seaborn, ruwt, matplotlib, numpy, gry2.

Параметры аппаратной части комплекса для контроля движений выбраны следующие:

- МК ESP32S (NodeMCU). Имеется два ядра. На одном выполняется программа прошивки, а на втором поддерживается протокол беспроводной передачи данных. Ближайший аналог ESP8266 имеет лимит в передаче на уровне 120 Гц из-за того, что 80% процессорного времени уходит на поддержание протокола Wi-Fi<sup>19</sup>.
- Протокол 802,11n Wi-Fi. Передача данных по Wi-Fi является самой скоростной по сравнению с bluetooth 4,2 и COM портом.
- Шина приема данных с датчиков BMI160 – SPI.
- SPI делитель – 2.
- Частота работы CPU – 240 МГц.
- Размер памяти – 4МБ.

Параметры работы датчиков выбраны таким образом, чтобы максимально охватить широкий диапазон событий.

- Частота работы гироскопа – 400 Гц.
- Диапазон значений  $\pm 500^\circ/\text{сек}$ .
- Частота работы акселерометра – 400 Гц.
- Диапазон значений  $\pm 8g$ .
- Режим автоматической калибровки дрейфа нуля гироскопа – включен.

Параметры Wi-Fi точки доступа и клиента (Lenovo Z580):

- Wi-Fi точка доступа – мобильный Хотспот Windows 10.

---

<sup>19</sup>Wi-Fi stack allow 80% of the processing power [https://www.espressif.com/sites/default/files/documentation/0a-esp8266ex\\_datasheet\\_en.pdf](https://www.espressif.com/sites/default/files/documentation/0a-esp8266ex_datasheet_en.pdf)

- Microsoft Wi-Fi Direct Virtual Adapter.
- Версия драйвера – 10,0,17763.1.
- Сетевой адаптер – Intel Centrino Wireless-N 2200.
  - Диапазон – 2,4 ГГц, 802,11n.
  - Максимальная скорость – 300 Мбит/с.
- Процессор – Intel(R) Core(TM) i7-3520M, 2,9 ГГц.
- ОЗУ – 8 Гб.
- Тип – 64 разрядная.
- Обновление системы – KB4493509 (OS Build 17763,437)

Работа будет выполнена в описанном контексте. Схема расположения датчиков была представлена на рис. 5.1.

## **Разработка оригинальной печатной платы**

Для разработки перчатки контроля движений потребовалось спроектировать собственную плату для чипа ВМ160. Существующие платы не позволяли разместить их на фалангах пальцев.

Проектирование производилось в системе Eagle CAD, далее была изготовлена малая партия плат в размере 50 штук. Схема представлена на рис. 5.9, а готовая плата — на рис. 5.10. Площадь платы составила 7 x 8 мм<sup>2</sup>. В оригинальной схеме указано, что необходимо отдельное питание для работы самого датчика и для вывода его ножек. Однако в данной работе они соединены. Это приводит к тому, что вместо двух фильтрующих конденсаторов на линиях питания расположен только один. Используемый номинал — 100 нФ.

В рабочем прототипе используется односторонняя печатная плата с металлизацией отверстий. Однако в ходе эксплуатации были выявлены существенные недостатки данного решения. Данный вывод пинов удобно использовать для штыревого соединения. При пайке проводов в случае отсутствия защиты от внешних механических воздействий происходит их дальнейшее перетирание. Кроме того, подобное расположение и размер выводов сильно ограничивают минимально возможный размер платы. Есть

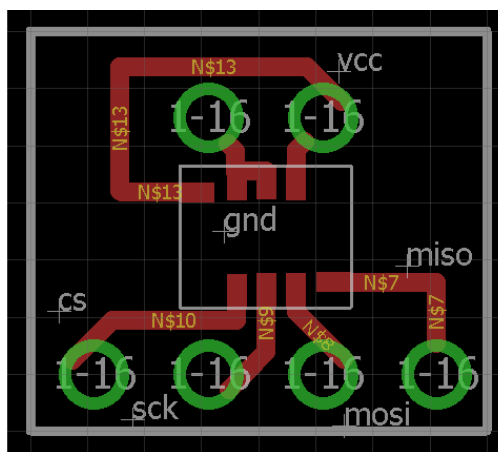


Рисунок 5.9 – Печатная плата, спроектированная в Eagle CAD.

перспективы для миниатюаризации вплоть до размера 3 мм × 3 мм.

Для сравнения размеров можно обратиться к рис. 5.10. Размер данной платы можно уменьшить на 84% вместе с повышением надежности контактов на линиях. Низкое энергопотребление — около 1мВт — подразумевает такие же низкие требования для линий питания.

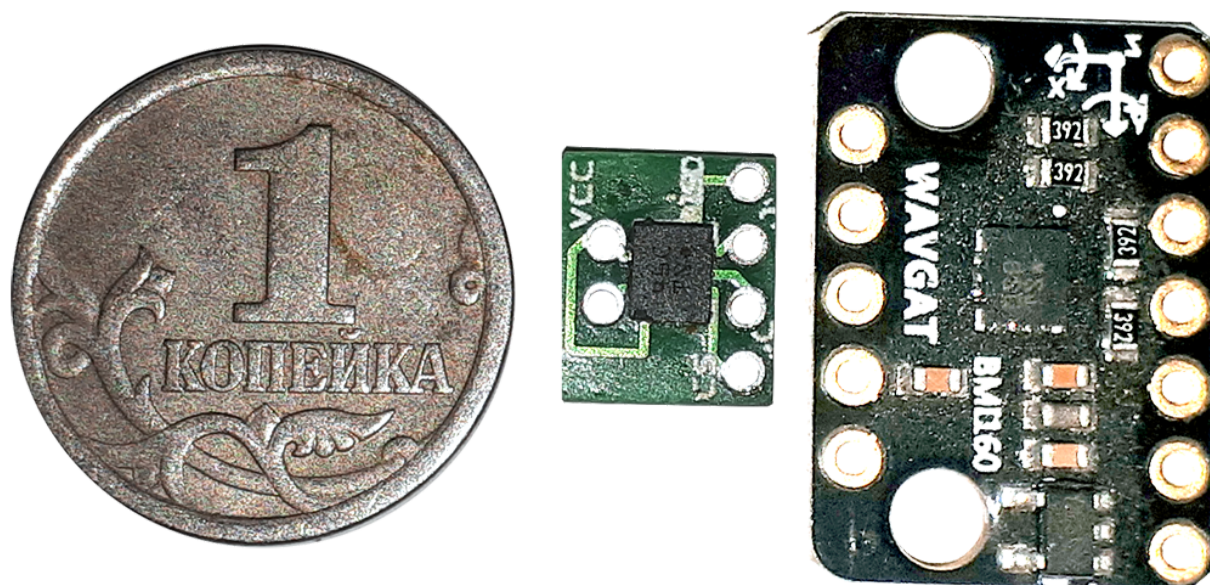


Рисунок 5.10 – Слева — монета номиналом 1 копейка. В центре — изготовленная печатная плата. Справа — потребительская плата, которую можно купить на рынке электроники.

Все датчики были смонтированы на платы путем верхнего нагрева с использованием термофена. Для предотвращения перегрева максимальная

температура была ограничена 200°C.

В качестве элементов питания используются стандартные литий-ионные аккумуляторы. В данной работе применены два таких элемента, включенных параллельно в сеть питания, их суммарная емкость составляет 6 800 мАч. Дополнительных преобразующих напряжение элементов нет, так как используемая плата имеет встроенный контроллер управления питанием с литий-ионных аккумуляторов.

Датчики зафиксированы на перчатке в одинаковой ориентации. В качестве линий связи и питания используется кабель МГТФ сечения 0,12мм<sup>2</sup>. Сами датчики имеют два уровня электро- и влагозащиты. В начале они были покрыты электроизоляционным лаком, а далее уложены в прозрачную термоусадочную трубку с нейтральным клеем. Данное решение позволяет исключить нежелательные замыкания линий и механически защитить плату от трения.

Комплекс в сборе можно посмотреть на рис. 5.11.



Рисунок 5.11 – Реализация программно-аппаратного комплекса в виде перчатки.

Соединения выполнены таким образом, чтобы отдельные пальцы руки можно было отсоединять без применения специальных инструментов. После отсоединения останутся штыревые разъемы, которые подходят для отладочного стенда. Таким образом, можно удостовериться в корректности работы отдельных пальцев перед их включением в общий комплекс. Данное решение помогло найти системную ошибку, которая проявляется при подключении большого числа датчиков на одну шину данных.

Один датчик и мультиплексор закреплены на корпусе МК. Сверху они полностью покрыты термоклеем. Помимо этого, комплекс имеет цветовую и позиционную маркировку: синий обозначает линию MISO, зеленый — MOSI, желтый — SCL, красный — питание, черный — землю. Линии CS имеют только последовательную маркировку: сначала идут нижние, затем средние, потом дальние фаланги пальцев.

В представленной работе используется SPI-протокол для передачи на МК. Для этой платы [128] разрешены не все возможные конфигурации SPI, а только следующие:

- CPOL = 0. CPHA = 0.
- CPOL = 1. CPHA = 1.

В данной работе используется вторая конфигурация.

### **Защита от помех на общей линии выбора микросхемы**

Системная ошибка появляется при одновременном включении в комплекс всех 16 датчиков. На логическом анализаторе она выглядит как на рис. 5.12.

Принцип работы линии SPI подразумевает, что одновременно только один датчик может держать линию CS на низком уровне. Но если посмотреть на каналы 3 и 4, которые соответствуют линия CS двух разных датчиков, то можно заметить ровные прямоугольные импульсы на одном канале и скачки на втором. В итоге возникает ситуация, когда одновременно зажаты на землю два датчика, из-за чего происходят коллизии в передаче данных. Решается данная проблема тривиально: необходимо каждую линию CS подтянуть на линию питания. Тогда подобные наводки сглаживаются,

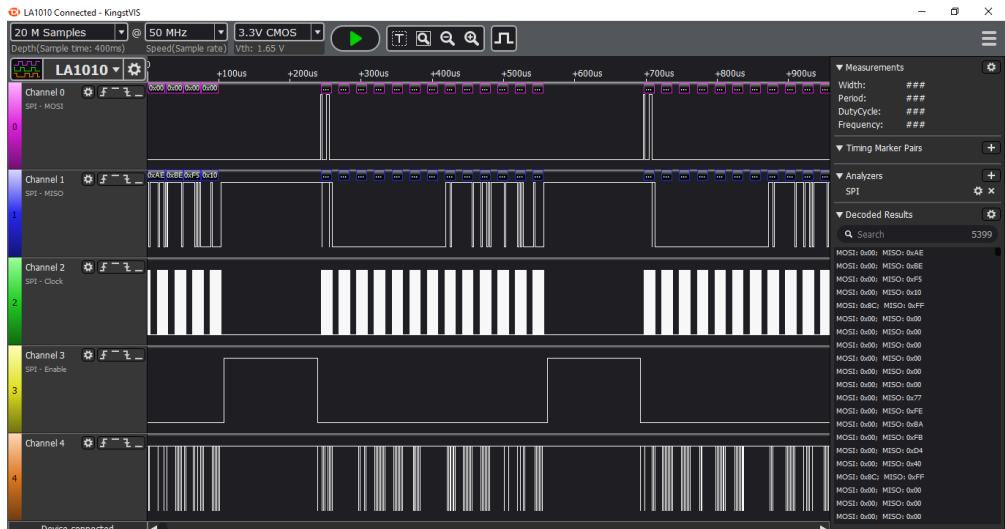


Рисунок 5.12 – Помехи на линии CS (channel 4).

и комплекс начинает работать стабильно. В данной работе используются резисторы номиналом в 5кОм. Работа комплекса с добавлением резисторов отражена на рис. 5.13, где отчетливо видны прямоугольные импульсы на двух линиях CS.

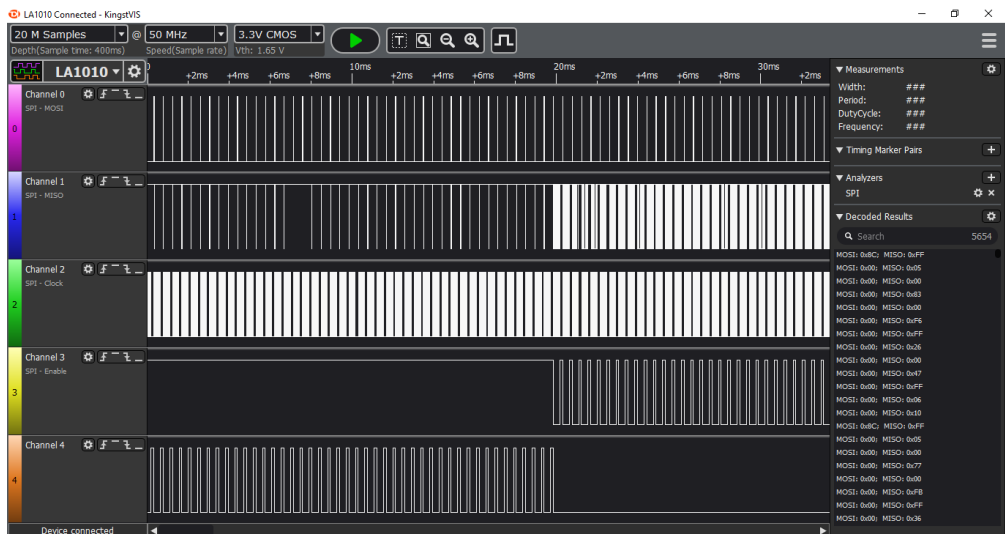


Рисунок 5.13 – Рабочий режим комплекса.



## Публикации автора по теме диссертации

- **Гайниязов И.М.**, Морозов А.А. Сравнение алгоритмов фильтрации сырых данных для маркерной киберфизической системы захвата движений, Молодой ученый, №16, С.192–195, 2017. (0,19 п.л. / 0,15 п.л.).
- **Гайниязов И.М.**, Обабков И.Н., Хлебников Н.А. Методика захвата движений как средство естественного интерфейса. GraphiCon // Международная конференция по машинному зрению. 24-28 сентября 2017. Тезисы. Пермь, ПГНИУ, 2017. С. 193–196. (0,25 п.л. / 0,21 п.л.).
- **Гайниязов И.М.**, Морозов А.А. Технология захвата движений и образовательные роботы. //IV Международная научная конференция «Физика. Технологии. Инновации». 15-19 мая 2017. Тезисы. Екатеринбург, УрФУ, 2017. С. 127–128. (0,13 п.л. / 0,11 п.л.).
- **Гайниязов И.М.**, Стародубцев И.С. и др. Аниматронная модель руки на базе ESP8266. //Международная конференция по машинному зрению. 23-26 сентября 2019. Брянск, Брянский государственный технический университет, 2019. С. 274–278. (0,32 п.л. / 0,23 п.л.).
- **Гайниязов И.М.**, Валиева Э.Р. и др. Исследование мелкой моторики в VR. // Международный форум "COGNITIVE NEUROSCIENCE — 2019 . 06-07 ноября 2019. Екатеринбург, Уральский федеральный университет, 2019. С. 221–222. (0,13 п.л. / 0,11 п.л.).
- **Гайниязов И. М.**, Стародубцев И. С. Построение модели движения кисти руки на базе микроэлектромеханических сенсоров и алгоритма динамической трансформации временной шкалы //Интеллектуальные системы в науке и технике. Искусственный интеллект в решении актуальных социальных и экономических проблем XXI века. – 2020. – С. 165-170. (0,38 п.л. / 0,28 п.л.).

**В том числе в ведущих рецензируемых журналах из списка ВАК и приравненных к ним**

- **И.М. Гайниyarov.**, И.Н. Обабков. Методика снижения размерности данных о движении, основанный на вейвлет–преобразовании // Инженерный вестник Дона. – 2021. – Т. 72. – №. 9 (72). – С. 246-255. (0,63 п.л. / 0,51 п.л.).
- **И.М. Гайниyarov.** Уменьшение признакового пространства моделей движения кисти //Инженерный вестник Дона. – 2021. – Т. 71. – №. 8 (71). – С. 252-259. (0,5 п.л. / 0,5 п.л.).
- Averbukh V. L., Averbukh N. V., **Gajniyarov I.** Problems Arising in the Design of Workstations Based on Augmented Reality //2020 GloSIC. – IEEE, 2020. – С. 55-59. (0,32 п.л. / 0,21 п.л.). (Scopus, WoS).
- Averbukh V. L., Natalya V. Averbukh, Pavel Vasev, **Igor Gajniyarov**, Ilya Starodubtsev. The Tasks of Designing and Developing Virtual Test Stands //2020 GloSIC. – IEEE, 2020. – P. 49-54. (0,44 п.л. / 0,11 п.л.). (Scopus, WoS).
- **Igor Gajniyarova**, Ilya Obabkovb, Ilya Starodubtsev, Olga Lvova, Ekaterina Suleimanova, Irina Antipina: The Motion Capture as Behavior Analyzing Method of Spontaneous Motor Activity in Human Infants, 2019 SIBIRCON, IEEE, 2019. – PP. 681-684. (0,25 п.л. / 0,15 п.л.). (Scopus).
- **Igor Gajniyarov**, Ilya Starodubtsev, Rustam Samedov, Alisa Sibogatova, Irina Antipina, Yana Zolotareva: Animatronic Hand Model on the Basis of ESP8266, 2019 International Multi-Conference on Engineering, Computer and Information Sciences (SIBIRCON), IEEE, 2019. – PP. 500-503. (0,25 п.л. / 0,15 п.л.). (Scopus).
- **Igor Gajniyarov**, Ilya Obabkovb, Nikolai Khlebnikovc: Motion capture in educational robotics, AIP Conference Proceedings, vol. 1886, №1, AIP Publishing, 020085, Eds: American Institute of Physics Inc., 2017. (0,5 п.л. / 0,35 п.л.). (Scopus, WoS).