

Министерство образования и науки Украины  
Одесский национальный политехнический университет

На правах рукописи

НГУЕН ГУИ КЫОНГ

УДК 004.93

МОДЕЛИ И МЕТОДЫ КОМПЬЮТЕРНОЙ ПОДДЕРЖКИ ПРИНЯТИЯ  
РЕШЕНИЙ В АВТОМАТИЗИРОВАННЫХ СИСТЕМАХ ДВИГАТЕЛЬНОЙ  
РЕАБИЛИТАЦИИ

Специальность 05.13.06 – Информационные технологии

Диссертация на соискание ученой степени  
кандидата технических наук

Научный руководитель  
Болтенков Виктор Алексеевич  
кандидат технических наук, доцент

Одесса – 2015

## СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	7
РАЗДЕЛ 1 АНАЛИЗ СОСТОЯНИЯ И ТЕНДЕНЦИЙ РАЗВИТИЯ СИСТЕМ ПРИНЯТИЯ РЕШЕНИЙ ПРИ ДИСТАНЦИОННОЙ ДВИГАТЕЛЬНОЙ РЕАБИЛИТАЦИИ И ПОСТАНОВКА ОСНОВНЫХ ЗАДАЧИ ИССЛЕДОВАНИЯ.....	13
1.1 Процесс поддержки принятия решений при дистанционной двигательной реабилитации.....	13
1.2 Анализ технических средств поддержки принятия решений при дистанционной двигательной реабилитации.....	16
1.3 Анализ моделей внутрикадрового представления тела человека и моделей движения человека в видеопотоке.....	22
1.3.1 Внутрикадровые модели.....	22
1.3.2 Анализ моделей движения человека в видеопотоке.....	29
1.4 Анализ методов поддержки принятия решений.....	33
1.5 Постановка основных задач исследования.....	35
Выводы по первому разделу.....	37
РАЗДЕЛ 2 ДВУМЕРНАЯ ПРОЕКТИВНАЯ ТОЧЕЧНАЯ МОДЕЛЬ ДВИЖЕНИЯ ТЕЛА ЧЕЛОВЕКА.....	39
2.1 Принцип построения двумерной проективной точечной модели движения тела человека.....	39
2.2 Построение точечной модели.....	42
2.3 Базовая информационная технология обнаружения координат характерных точек тела.....	57
Выводы по второму разделу.....	58
РАЗДЕЛ 3 КИНЕМАТИЧЕСКАЯ МОДЕЛЬ ДВИЖЕНИЯ ЧЕЛОВЕКА В ВИДЕОПОТОКЕ, МЕТОД ПОСТРОЕНИЯ СТАТИСТИЧЕСКИХ ОЦЕНОК КИНЕМАТИЧЕСКИХ ПАРАМЕТРОВ И МЕТОД ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ ПО КОРРЕКЦИИ ДВИЖЕНИЙ ПАЦИЕНТА.....	60
3.1 Кинематическая модель движения тела человека в видеопотоке.....	60

3.2 Метод построения статистических оценок кинематических параметров .....	67
3.3 Проверка адекватности построенной модели и метода усреднения кинематических параметров .....	69
3.4 Базовая информационная технология построения кинематической модели и оценивания кинематических параметров .....	75
3.5 Оценка энтропии оценивания кинематических параметров движения по разработанной модели .....	76
3.6 Метод поддержки принятия решений по коррекции движений пациента .....	82
3.7 Базовая информационная технология формирования результата сравнения движений и вынесения корректирующего решения .....	88
Выводы по третьему разделу .....	89
<b>РАЗДЕЛ 4 ИНФОРМАЦИОННАЯ ТЕХНОЛОГИЯ ПОСТРОЕНИЯ СИСТЕМЫ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ ДЛЯ ДИСТАНЦИОННОЙ ДВИГАТЕЛЬНОЙ РЕАБИЛИТАЦИИ И ОЦЕНКА ХАРАКТЕРИСТИК РАЗРАБОТАННОЙ СИСТЕМЫ .....</b>	<b>91</b>
4.1 Информационная технология поддержки принятия решений .....	91
4.2 Система поддержки принятия решений, разработанная на основании ИТ и ее архитектура .....	94
4.3 Программная разработка СППР .....	97
4.3.1 Технология программного средства .....	97
4.3.2 Описание основных программных классов, модулей и библиотек ...	98
4.3.3 База данных системы .....	101
4.4 Тестирование, экспериментальная проверка системы и оценка характеристик системы .....	102
4.4.1 Калибровка системы .....	103
4.4.2 Экспериментальная оценка рабочей зоны системы .....	104
4.4.3 Оценка показателей достоверности принятия решений .....	107
Выводы по четвертому разделу .....	109

ВЫВОДЫ.....	110
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ .....	112
ПРИЛОЖЕНИЕ А Ключевые фрагменты программного кода.....	125
ПРИЛОЖЕНИЕ Б Акты внедрений результатов диссертационной работы.	142

## ПЕРЕЧЕНЬ УСЛОВНЫХ СОКРАЩЕНИЙ

АДПРХИ	– Анализ данных, принятие решения и хранение информации
АР	– Авторегрессия
БИТ	– Базовая информационная технология
БД	– База данных
ВД	– Выполненное движение
ДДР	– Дистанционная двигательная реабилитация
ИТ	– Информационная технология
ЛПР	– Лицо, принимающее решение
НЧМБ	– Нормированная чебышевская мера близости
ОУ	– Обучающее упражнение
ППР	– Поддержка принятия решений
СКО	– Среднее квадратическое отклонение
СММ	– Скрытая марковская модель
СППР	– Система поддержки принятия решений
СМА	– Cumulative moving average
CMS	– Content Management System
DTW	– Dynamic time warping
EAV	– Entity-Attribute-Value

- MA – Moving average
- MAT – Medial Axis Transform
- WMA – Weighted moving average

## ВВЕДЕНИЕ

**Актуальность темы.** Одной из эффективных отраслей современной телемедицины является дистанционная двигательная реабилитация (ДДР) пациентов с заболеваниями опорно-двигательного аппарата, центральной нервной системы, детским церебральным параличом. Сегодня в мире существуют миллионы потенциальных пользователей систем ДДР. ДДР построена на удаленной демонстрации пациенту врачом-инструктором комплекса двигательных упражнений, повторении его пациентом и корректирующих рекомендаций врача пациенту по правильному выполнению движений.

В основе ДДР лежат модели движения человека в видеопотоке и методы сравнения этих движений с учебными движениями. На основании сравнительного анализа выполняемых пациентом движений с учебными врач принимает решение по текущей коррекции процедуры лечения. Эффективность лечения в значительной степени определяется правильностью решения, принятого врачом. С этой точки зрения весьма актуальны системы компьютерной поддержки принятия решения (ППР) врачом. Существующие системы поддержки принятия решения (СППР) при ДДР, как правило, дают бинарное решение: "движение выполнено правильно/неправильно" без оценки количественных параметров, позволяющих оценить правильность выполнения упражнения и выработать корректирующие рекомендации. Известные модели движения человека в видеопотоке сложны, и вместе с тем недостаточно формализованы и точны по построению двигательного образа. Методы количественного сравнения движений сложны и практически неприменимы для практикующего врача.

Это говорит об актуальности исследований в данном научном направлении. В работе для устранения перечисленных недостатков известных систем предлагается разработать информационную технологию (ИТ) поддержки принятия решений на основании моделей оценки

количественных параметров движения человека по видеопоследовательности и методов, позволяющих выработать обоснованное решение по коррекции движения.

**Связь работы с научными программами, планами, темами.** Диссертационная работа выполнялась в соответствии с перечнем приоритетных направлений научных исследований на период до 2015 года в области «Информационные и телекоммуникационные технологии», утвержденным постановлением № 943 Кабинета министров Украины от 07.09.2011, и приоритетными направлениями научно-исследовательских работ Одесского национального политехнического университета (ОНПУ), в частности в рамках госбюджетных научно-исследовательской работы кафедры информационных системных «Исследование и разработка методов и алгоритмов распознавания визуальной информации и анализа сцен» (№ гос. регистрации 0111U010458 – при непосредственном участии автора.

**Цель и задачи исследования.** Целью диссертационной работы является разработка и исследование моделей, методов и информационной технологии компьютерной поддержки принятия решений в автоматизированных системах двигательной реабилитации для повышения достоверности принятия решений.

Для достижения поставленной цели решены следующие задачи:

- Анализ существующих систем ППР при ДДР и лежащих в их основе моделей и методов оценки параметров движения человека в видеопотоке и выбор направления исследований.
- Разработка проективной модели для покадрового описания элемента движения человека в видеопотоке.
- Разработка кинематической модели для описания движения человека в видеопотоке.
- Разработка метода поддержки принятия решений на основании количественных мер близости движений.



– Разработка информационной технологии построения системы ППР для ДДР.

– Практическая реализация системы ППР врачом для двигательной реабилитации на базе разработанных методов и моделей и информационной технологии.

– Апробация системы ППР для ДДР в реальных условиях.

**Объект исследования** – Процесс поддержки принятия решения при двигательной реабилитации пациентов.

**Предмет исследования** – Модели и методы анализа движения человека в видеопотоке для компьютерной поддержки принятия решений при двигательной реабилитации пациентов.

**Методы исследования** – методы цифровой обработки изображений, теоретическая кинематика, теория мер близости, методы скелетизации изображений, методы теории вероятности, математической статистики при разработке моделей и методов и имитационного моделирования при разработке и проверке адекватности предложенных моделей и методов.

#### **Научная новизна работы.**

Научная новизна диссертационной работы сформулирована следующими положениями:

1. Впервые разработана двумерная проективная модель для покадрового представления движения человека в видеопотоке, которая в отличие от существующих на основании построения и обработки скелета позволяет формально описать плоский кадр с помощью 16-ти характерных точек с целью последующей статистической оценки кинематических параметров.

2. Получил дальнейшее развитие метод обнаружения характерных точек, который учитывает результаты логического анализа скелета с помощью скользящей маски, что позволяет повысить достоверность обнаружения характерных точек;

3. Впервые разработана кинематическая модель тела человека в видеопотоке, которая в отличие от существующих построена на аппроксимации траекторий характерных точек в 3-х кадровых фрагментах видеопотока дугами окружностей, что позволяет получить точечные оценки мгновенных тангенциальных скоростей и ускорений характерных точек тела;

4. Получил дальнейшее развитие метод построения статистических оценок кинематических параметров, который основан на расчете их скользящего среднего по последовательным фрагментам видеопотока, что позволяет получить несмещенные, состоятельные и эффективные оценки координат, тангенциальных скоростей и тангенциальных ускорений ХТ тела;

5. Впервые разработан метод и информационная технология поддержки принятия решения по оценке правильности и коррекции движения пациента, которые отличаются возможностью количественного сравнения параметров обучающего и текущего движений, что позволяет повысить достоверность принятия правильного решения;

**Практическая ценность** полученных результатов состоит в следующем:

На основании разработанных моделей и методов разработана информационная технология, реализованная в виде автоматизированной системы поддержки принятия решения при дистанционной двигательной реабилитации. Автоматизированная система испытана и внедрена на предприятии "Vietlogic". Разработанные модели и методы и информационная технология позволяют существенно сократить объемы информации, передаваемой по каналу связи "Пациент-Лечебное учреждение" и информации, сохраняемой в архивах медицинского учреждения. Основные положения диссертационного исследования использованы в учебном процессе Одесского национального политехнического университета кафедрой информационных систем в лабораторных практикумах, при курсовом и дипломном проектировании.

**Личный вклад соискателя.** Все научные положения и практические результаты, представленные в диссертационной работе, были получены автором лично. Работы [5, 6] выполнены без соавторов. В работах, опубликованных в соавторстве, личный вклад соискателя такой: [1] – разработана двумерная проективная модель для покадрового представления движения человека в видеопотоке; [2,7,9] – разработана и исследована кинематическая модель движения человека и метод построения статистических оценок кинематических параметров; [3,8] – разработан метод обнаружения характерных точек на основе анализа скелетов изображения человека [4, 10, 11] – разработан метод поддержки принятия решения по оценке правильности и коррекции движения пациента и разработана информационная технология и структура системы поддержки принятия решений при двигательной реабилитации.

**Апробация результатов диссертации.** Основные положения диссертационной работы докладывались и обсуждались на научных семинарах кафедры информационных систем и следующих научных конференциях:

– Международная научно-практическая конференция «Современные информационные и электронные технологии – 2013 (СИЭТ-2013)» (г. Одесса, 23-27 мая 2013 г.);

– Международная научно-практическая конференция «Информационные управляющие системы и технологии – 2013 (ИУСТ-2013)» (г. Одесса, 8-10 октября 2013 г.);

– Международная конференция студентов и молодых ученых «Современные Информационные Технологии – (МИТ-2014)» (г. Одесса, 22-26 апреля 2014 г.);

– Международная научно-практическая конференция «Современные информационные и электронные технологии – 2014 (СИЭТ-2014)» (г. Одесса, 26-30 мая 2014 г.);

– 8-th International Forum on Innovative Technologies for Medicine

(ITMED 2014) – (Poland, Suprasl, December 4-6, 2014);

– Международная научно-практическая конференция молодых ученых и студентов «Информационное, программное и техническое обеспечение систем управления организационно-техническими комплексами» (г. Луцк, 5-6 апреля 2015 г.).

По теме диссертации опубликовано 11 работ, в том числе 5 статей в научных журналах из перечня изданий, утвержденных ВАК Украины, 6 – в трудах или тезисах докладов международных и всеукраинских конференций.

### **Структура и объем работы.**

Диссертация состоит из введения, четырех глав, заключения, списка использованных источников из 118 наименований и 2 приложений. Полный объем – 145 страниц, объем основного текста – 111 страниц. Диссертация содержит 37 рисунков, 10 таблиц.

## РАЗДЕЛ 1

### АНАЛИЗ СОСТОЯНИЯ И ТЕНДЕНЦИЙ РАЗВИТИЯ СИСТЕМ ПРИНЯТИЯ РЕШЕНИЙ ПРИ ДИСТАНЦИОННОЙ ДВИГАТЕЛЬНОЙ РЕАБИЛИТАЦИИ И ПОСТАНОВКА ОСНОВНЫХ ЗАДАЧИ ИССЛЕДОВАНИЯ

В течение последнего десятилетия в мире наблюдается значительное повышение интереса к дистанционной двигательной реабилитации (ДДР) как эффективной отрасли телемедицины, направленной на преодоление последствий тяжелых последствий неврологических заболеваний, сосудистых катастроф, тяжелых травм. Главной особенностью систем ДДР является разделение в пространстве пациента, выполняющего двигательную терапию и врача-реабилитолога, являющегося лицом принимающим решение (ЛПР). Врач-реабилитолог, вынося решение о правильности выполнения движений и их коррекции, сегодня полагается не только на свой личный опыт, но и на средства поддержки принятия решения, которыми являются информационные технологии и технические средства, позволяющие количественно оценить параметры движений пациента.

В классических системах ДДР врач и пациент разделены в пространстве, но связаны во времени. Целью исследований в области построения систем является их перевод в автоматический режим, когда рекомендации пациенту выдает автомат, а врач вмешивается в процесс ДДР при необходимости в удобное для него рабочее время. Это требует разработки новых информационных технологий ППР и повышения достоверности принятого решения.

#### 1.1 Процесс поддержки принятия решений при дистанционной двигательной реабилитации

В условиях кризисного финансирования социальных и, в частности, медицинских программ особо важную роль приобретает телемедицина –

дистанционное оказание консультативных, лечебных и иных услуг пациентам с использованием телекоммуникационных и информационных технологий. Телемедицина позволяет в условиях интегрированного информационного пространства эффективно использовать интеллектуальный потенциал и клинический опыт высококлассных специалистов с минимальными затратами [12,13].

Одной из эффективных отраслей современной телемедицины является дистанционная двигательная реабилитация (ДДР) пациентов с заболеваниями опорно-двигательного аппарата, центральной нервной системы, детским церебральным параличом. В мире существуют миллионы потенциальных пользователей систем ДДР. Действие систем ДДР основано на многократном повторении пациентом движений, продемонстрированных удаленным врачом-инструктором и коррекции этих повторяемых движений [14-18]. Классическая и исторически первая схема ДДР представляет собой взаимодействие врача и пациента в режиме онлайн. В данном случае решения по коррекции движений пациента формулируются лицом, принимающим решение (ЛПР) – врачом-реабилитологом исключительно на основании формируемой им визуальной оценки параметров движения пациента и на основе личного профессионального опыта.

С появлением первых систем ДДР начали внедряться системы поддержки принятия решений (ППР) врачом-реабилитологом, позволяющие ЛПР принять более достоверное решение за счет применения технических средств, математических моделей и методов и информационных технологий, позволяющих объективно оценить параметры обучающего и повторяемого пациентом движений и получить количественные меры сравнения близости двух движений. Такие системы выполняют следующие функции:

- выделение на кадрах видеопотока характерных точек тела человека;
- оценка параметров движения характерных точек;
- количественная оценка близости или отличия параметров обучающего движения и движения, выполняемого пациентом.

В рамках общих тенденций развития технологий ППР при ДДР появились автоматические системы, в которых решение по коррекции движений пациента принимает автомат [19-20].

Формализуем задачу ППР при ДДР на основании общих положений теории принятия решений как задачу выбора на множестве альтернатив [21] в следующем виде. Множество альтернатив  $alt_i \in A, i = (1, n)$  представляет собой множество  $n$  наблюдаемых в видеопотоке частей тела или соответствующих им характерных точек (ХТ) тела человека, описывающих его движение, и состоит из двух подмножеств  $A = \{A^{naц} \cup A^{обуч}\}$ , где  $A^{naц}$  - подмножество ХТ, описывающих движение, выполняемое пациентом,  $A^{обуч}$  - подмножество ХТ, описывающих обучающее движение. Альтернативы оцениваются по множеству критериев  $K = \{k_j\}, j = (1, m)$ , которыми являются параметры оценки движения ХТ, в частности, координаты ХТ в кадре видеопотока, скорость и ускорение ХТ. Для каждой альтернативы (ХТ)  $alt_i^{naц} \in A^{naц}, alt_i^{набч} \in A^{набч}, i = (1, n)$  предполагается известным кортеж  $\langle R_i^{naц} = \{r_{ij}^{naц}\}, R_i^{обуч} = \{r_{ij}^{обуч}\} \rangle$  оценок каждой из альтернатив по каждому из критериев. При принятии решения лицом, принимающим решение (ЛПР) в качестве которого может выступать врач или автомат, задача принятия решения заключается в выборе альтернатив  $alt_i^{naц}$ , которые в наибольшей степени по некоторой мере отличаются от соответствующих альтернатив  $alt_i^{набч}$  по всем перечисленным критериям  $k_j$ . Выбранные альтернативы (ХТ) указываются пациенту как точки тела, для которых требуется коррекция выполняемого движения и выдаются соответствующие корректирующие рекомендации.

Общая схема информационной технологии ППР при ДДР показана на рис. 1.1.

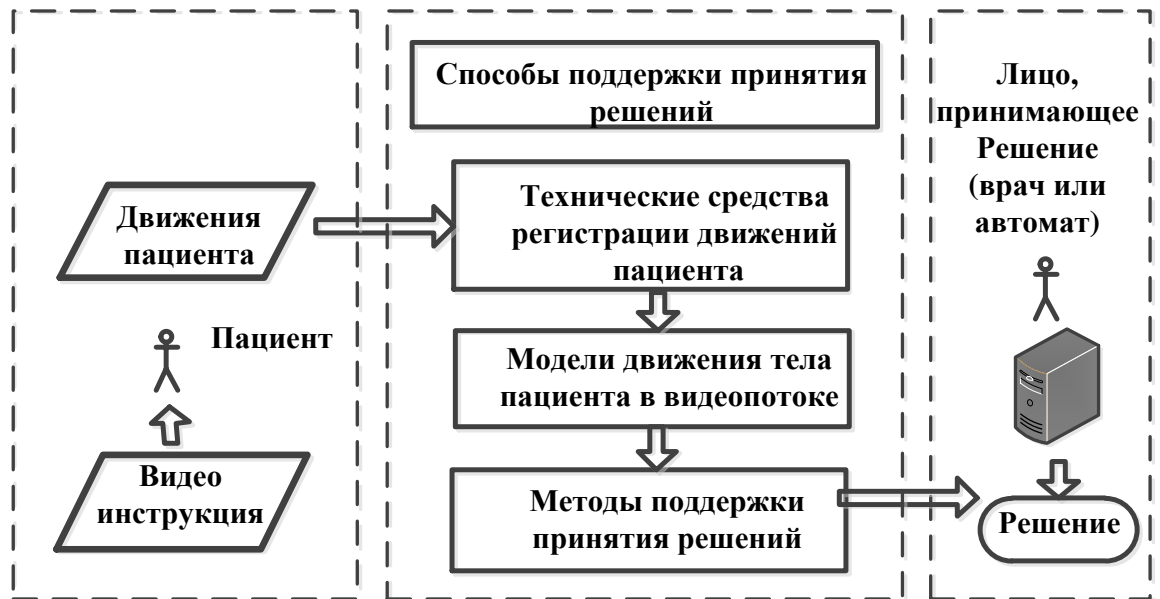


Рисунок 1.1 – Общая схема информационной технологии поддержки принятия решений

## 1.2 Анализ технических средств поддержки принятия решений при дистанционной двигательной реабилитации

Для оценки параметров движения тела пациента в процессе ДДР ЛПР доступны различные технические средства [22]. Их классификация приведена на рис. 1.2.

Сенсорные технические средства предполагают размещение на теле пациента сенсоров различной физической природы. Сенсоры соединяются с компьютерными средствами регистрации и обработки данных проводными или беспроводными каналами передачи данных. Число размещенных на теле пациента сенсоров может достигать 20-ти и более.

Наиболее точными считаются системы с гироскопическими или акселерометрическими сенсорами, позволяющие получить оценки ускорения с ошибкой не более  $0,01 \text{ м/с}^2$  [23-25]. Пример такой системы приведен на рис. 1.3.



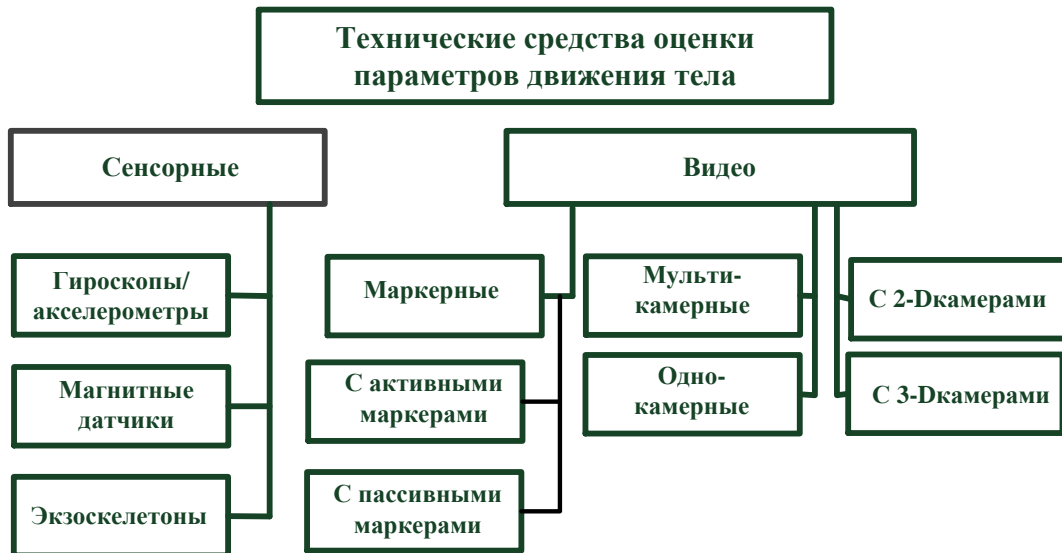


Рисунок 1.2 – Классификация технических средств оценки параметров движения тела



Рисунок 1.3 – Пример системы с акселерометрическими и гироскопическими датчиками, закрепляемыми на теле пациента [25]

Недостатки таких систем:

- высокая стоимость гироскопов и акселерометров в миниатюрном (MEMs) исполнении;
- дополнительный контроллер, прикрепленный к телу человека или даже связка проводов, тянущаяся от соединяющая сенсоры с компьютерными средствами в случае проводных каналов связи;

– для определения положения тела человека в пространстве и закрепленных на теле сенсоров требуется дополнительная мини-система начальной калибровки (например, видеосистема).

В магнитных системах [26] маркерами являются магнитные элементы. Система оценивает их позиции по искажениям магнитного потока.

Недостатки магнитных систем следующие:

- магнитные системы подвержены магнитным и электрическим помехам от металлических предметов;
- они имеют существенно меньшую по сравнению с оптическими системами рабочую зону;
- как и в инерционных системах требуется дополнительный контроллер, прикрепленный к телу и подключенный к магнитным маркерам, или связка проводов;
- высокая стоимость магнитных маркеров.

Экзоскелетоны [27] представляют собой специальный механический мосар-скелет (англ. MOtion CAPture), который надевается на тело пациента и повторяет за ним все движения. В компьютерную систему при этом передаются данные об углах сгибов всех суставов. Мосар-скелет содержит до 30 сенсоров различных принципов действия, регистрирующих смещения, скорости и ускорения различных отделов тела человека. На рис.1.4 показан пример экзоскелетонной конструкции.

Недостатки экзоскелетонных систем:

- мосар-скелет ограничивает движения пациента;
- риск поломки прецизионной механики экзоскелетона. При неосторожном использовании;
- высокая стоимость костюма – около 25,000\$ [28].

Учитывая основные требования к системам оценки параметров движения при ДДР: эргономичность – система не должна причинять неудобства пациенту (не сковывать движений, иметь незначительную массу и т.д.) и экономичность – система должна иметь невысокую стоимость и

быть доступной широкому кругу пользователей пациентов, можно утверждать, что сенсорные системы существенно проигрывают видеосистемам.



Рисунок 1.4 – Пример экзоскелетонной конструкции [28]

Видеосистемы оценки параметров движения пациента могут быть маркерными и безмаркерными. В маркерных системах на теле или одежде пациента в характерных точках для облегчения их захвата оптической системой закрепляются маркеры, которые могут быть активными (светодиоды с интегрированными процессорами и радиосинхронизацией, каждому светодиоду назначается идентификатор, что позволяет видеосистеме правильно определять необходимые маркеры, а также идентифицировать их после того, как они были перекрыты и снова появились в поле зрения камеры) и пассивными – в этом случае в качестве маркеров используются светоотражающие пластины [29-30] (рис.1.5а,б). Иногда в качестве пассивных маркеров выступает разноцветная одежда, надеваемая пациентом на различные части тела (рис.1.5в) [31]. Во всех случаях маркерные системы причинять дополнительные неудобства пациенту

связанные с длительностью размещения маркеров на теле или одежде, и удорожают стоимость оборудования для ДДР.

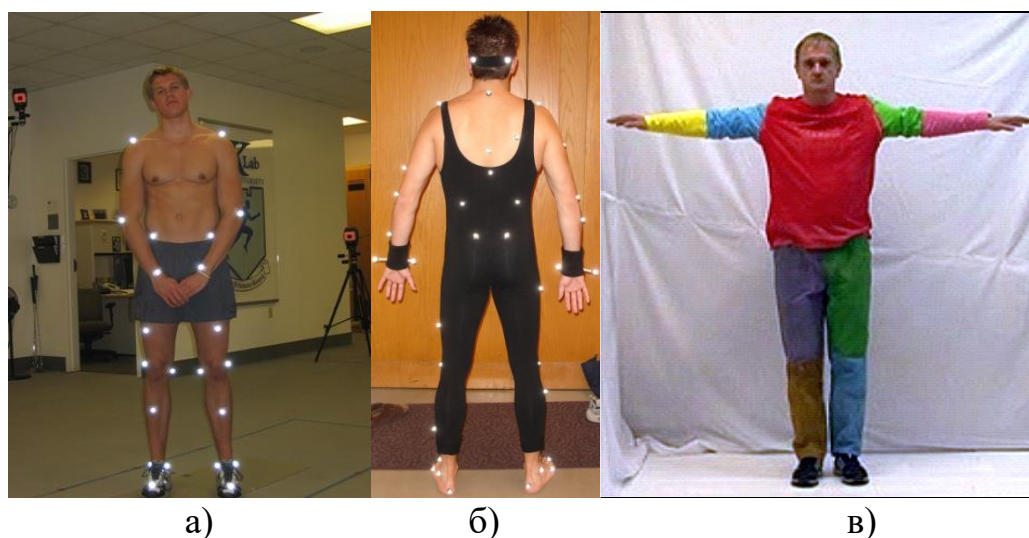


Рисунок 1.5 – Примеры систем со светодиодными маркерами (а,б) [30] и с разноцветными частями одежды, выполняющими роль маркеров (в) [31]

Очень высокие результаты по точности ДДР дают системы типа "сенсоры+маркеры+видеоанализ". В частности, такие системы разработаны научной школой А.В. Воронова [32-34]. Однако такие системы представляют собой испытательно-тренировочные стенды, они не могут быть рекомендованы для домашней ДДР.

Наиболее экономичными и технологически простыми являются системы видеонаблюдения за движениями пациентов. Видеорегистрация движения тела человека имеет давнюю историю [35]. Однако реально первая система видеоанализа движений пациента появилась в 1985 г. [36-37]. Видеосистемы для оценки параметров движения при ДДР могут быть мультикамерными (до 24 камер) и однокамерными. Мультикамерные системы требуют большого помещения и специального освещения, что практически исключает применение технологии rehab@home, т.е. двигательной реабилитации пациента в домашних условиях [38].

В ДДР применяются обычные видеокамеры (2D) и трехмерные (3D) видеокамеры, в основном системы Kinect [39]. Камера Microsoft Kinect, первоначально созданная как аксессуар для игрового проекта представляет собой два сенсора глубины: инфракрасный и CMOS-сенсор, интегрированные с RGB-камерой (рис.1.6). Для каждой характерной точки Kinect камера оценивает три пространственных координаты, что позволяет сформировать трехмерную точечную модель тела.



Рисунок 1.6 – Видеосенсор системы Kinect

Недостатками систем Kinect являются:

- достаточно высокая стоимость, если стоимость собственно камеры с консолит. составляет примерно 600\$, то стоимость программного обеспечения для организации ДДР может в несколько раз превышать стоимость камеры;

- Kinect системы требуют высокопроизводительных вычислительных ресурсов, поскольку выходной информационный поток требует регистрации и обработки примерно 6.5 миллионов пикселей в секунду. Перечисленные недостатки делают Kinect системы практически недоступными для широкого пользователя систем ДДР.

Системы оценивания параметров движения с одиночной бытовой веб-камерой [40-42] на сравнении со всеми проанализированными выше имеют ряд безусловных преимуществ:

- невысокая стоимость и доступность;
- возможность работы в помещениях без специального освещения;
- удобство применения для пациента;
- обеспечение достаточной для ДДР точности оценивания параметров движения пациента при использовании соответствующих математических моделей и информационных технологий.

Несмотря на это в современной профессиональной литературе сегодня ведутся дискуссии, в которых некоторые авторы [26] настаивают на преимуществах сенсорных систем, размещаемых на теле пациента, с беспроводными каналами связи, которые на их взгляд более пригодны для систем ДДР по сравнению с видеорегистраторами.

### 1.3 Анализ моделей внутрикадрового представления тела человека и моделей движения человека в видеопотоке

#### 1.3.1 Внутрикадровые модели

Все модели представления тела человека внутри кадра видеопотока можно разделить на двумерные (2D) и трехмерные (3D) или воксельные ( voxel – volume element – объемный элемент изображения, термин введенный по аналогии с pixel). Как те, так и другие модели являются точечными, т.е. описывают визуальный образ тела множеством его характерных точек [42].

Первый класс точечных моделей представляют силуэтные модели, в которых кадровый образ тела описывается множеством точек его силуэта (рис.1.7 ) [43-45].

Полная силуэтная модель, показанная на рис.1.7, достаточно просто формируется путем последовательного применения операций вычитания фона кадра, пороговой бинаризации изображения, получения его силуэта и замены силуэта множеством силуэтных точек. Очевидный недостаток модели

– необходимость последующего трекинга 30-40 точек, что создает серьезные вычислительные трудности.

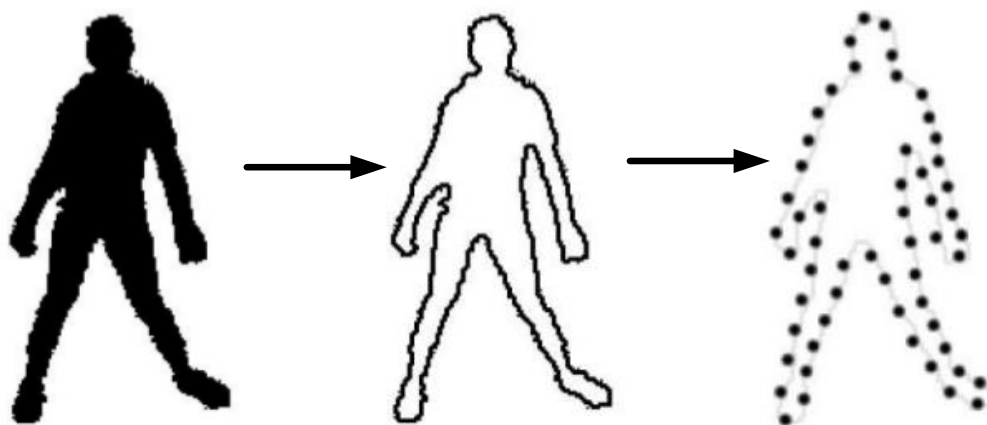


Рисунок 1.7 – Силуэтная модель тела человека [43]

Упрощенным вариантом силуэтной модели является "звездная модель" (англ. Star Model) [46-47], в которой по силуэту определяется точка центра масс тела, а затем при моделировании отслеживаются 6 точек (рис.1.8) – центр масс, конечные точки рук, ног и головы. Модель безусловно проста в вычислительном отношении, но непригодна для задач ДДР, поскольку недостаточно точно описывает зрительный образ тела.



Рисунок 1.8 – "Звездная модель" тела человека [46]

Весьма своеобразно формируется точечная модель 3D тела в Kinect-системах. С помощью Kinect-сенсора для 24 точек тела (рис.1.9) две плоские координаты в кадре  $(x_i, y_i)$  и координата "глубины"  $h_i, i = 1, \dots, 24$ . Далее между каждой тройкой точек строятся пространственные углы, и при движении тела объектом трекинга являются эти углы [48-55]. Модель позволяет построить воксельную картину тела, что достаточно наглядно. Однако для систем ППР при ДДР такая модель не пригодна: ЛПР – врач-реабилитолог – в процессе своей деятельности привыкает мыслить зрительным образом человека в виде частей тела или точек тела. Пространственные углы не являются приемлемым для врача зрительным образом.

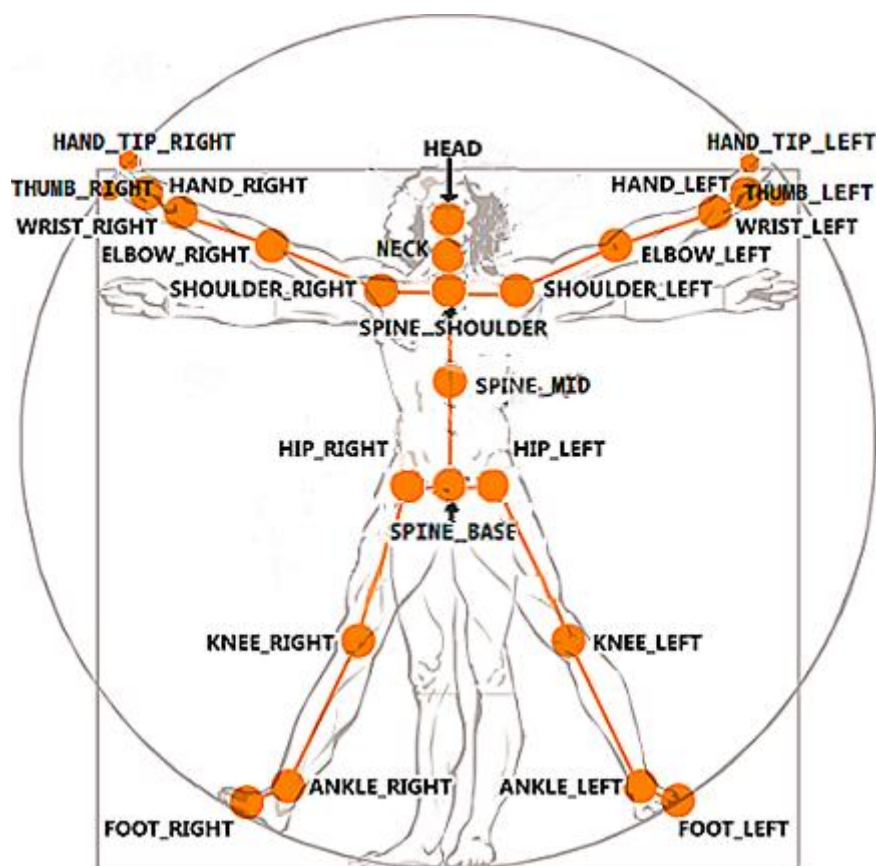


Рисунок 1.9 – 24-точечная модель тела человека в системе Kinect [49]

Скелетные модели. Другим способом представления изображения тела человека внутри кадра видеопотока является скелетная модель, согласно



которой изображение тела преобразуется в скелет и далее в видеопотоке производится трекинг характерных точек скелета [56-62].

Понятие скелета или срединной оси плоской фигуры было впервые введено в конце 60-х годов 20 в. Бламом [62]. Блам показал, что медиальное представление фигур (англ. *medial representation*) на двумерных изображениях, является эффективным способом описания их геометрической структуры. По сравнению с традиционным представлением формы медиальное представление является более информативным, оно отражает как общую структуру объекта, так и детальную структуру его элементов. Скелет сохраняет информацию о топологической структуре объекта и сокращает объем памяти, необходимой для ее хранения. Согласно Бламу скелет (англ. *skeleton*) плоской фигуры – это множество тех ее точек, для которых существует не менее двух равноудаленных ближайших точек границы фигуры. Из этого следует, что каждая точка скелета является центром окружности, лежащей в области и касающейся ее границы в двух или более точках (так называемой максимальной пустой окружности), а скелет – это множество центров максимальных вписанных в эту фигуру окружностей. Правильно построенный скелет имеет следующие свойства:

- сохраняет топологию оригинального объекта;
- расположен в центральных точках геометрической области объекта;
- скелет есть связная фигура, т.е. все пиксели скелета связаны друг с другом;
- имеет ширину равную 1 пикселю (поэтому ряд процедур скелетизации называется утоньшением или тиннингом, от англ. *thinning*);
- для каждого пикселя скелета известно расстояние до границы объекта.

Анализ существующих публикаций по скелетизации позволил построить классификацию существующих алгоритмов скелетизации, которая приведена на рис.1.10.



Рисунок 1.10 – Классификация алгоритмов скелетизации на 3 пикселя),

Неитеративные алгоритмы скелетизации основаны на построении центральной линии изображения непосредственно в один проход без анализа всех отдельных пикселей. Самая большая группа алгоритмов – итеративные – основана на идее итеративного удаления контурных точек объектов до тех пор, пока на изображении останутся только точки скелета. Итеративные алгоритмы обычно используют логическую маску (как правило, размером 3 на 3 пикселя), которая перемещается по всему изображению и в каждый момент времени сопоставляется с соответствующим участком изображения, чтобы определить новое значение центрального пикселя. Таким образом, в результате просмотра всего изображения удаляется один (или несколько) из внешних слоев объекта. Количество просмотров изображения и, как следствие, время работы итеративных алгоритмов, зависит от размера объектов на изображении. Алгоритмы данной группы можно разделить на два класса: параллельные и последовательные. В параллельных алгоритмах логическое окно располагается одновременно во всех пикселях изображения и при его обработке не используются новые (полученные на данной

итерации) значения пикселей. При работе последовательных алгоритмов пиксели обрабатываются последовательно. При реализации параллельных алгоритмов для их корректной работы, обработанные данные (информация об удаленных пикселях) должны заноситься в отдельный массив, который предохраняет от возможности рекурсивного удаления пикселей.

Рассмотрим классификацию алгоритмов построения скелета с методологической точки зрения.

Все методы скелетизации базируются на нескольких основных концепциях:

- морфологические преобразования;
- обнаружения максимумов в дистанционных картах граничных точек;
- расчет диаграммы Вороного для фигуры, образованной граничными точками;
- послойное удаление пикселей начиная с границы фигуры до образования фигуры толщиной в один пиксель.

Морфологические преобразования. Скелет по Бламу наглядно иллюстрируется примером "пожара в прерии". Предполагается, что по границе объекта одновременно вспыхивает огонь, который распространяется внутри нее по всем направлениям с постоянной скоростью. Те точки области, в которых сходятся два или более огненных фронта, являются по определению точками скелета. Практически Блам реализовал скелетизацию как медиальное осевое преобразование (англ. *Medial Axis Transform (MAT)*). Дальнейшее развитие идеи Блама получили в работах [63] Серра, где изложена парадигма анализа изображений, получившая название математической морфологии изображений, основанная на нелинейных операторах. Они традиционно описываются в теоретико–множественном формализме. Согласно морфологической концепции скелетизация выполняется как последовательное выполнение процедур эрозии и дилатации (т.е. послойного монопиксельного утоньшения и наращивания объекта на изображении).

Скелетизация на базе дистанционных карт выполняется в три этапа:

1. Бинарное изображение разделяется на характерные и нехарактерные пиксели, в характерные пиксели принадлежат границе (контур) объекта.
2. Генерируется дистанционная карта, каждый элемент которой указывает для каждого пикселя расстояние до ближайшего характерного элемента.
3. Локальные экстремумы дистанционной карты определяются как скелетные точки.

Поскольку дистанционная карта строится на расстояниях, измеряемых не в евклидовой метрике (обычно в манхэттенской или шахматной метрике) получаемые с использованием разных расстояний скелеты могут существенно различаться. Трудоемкость метода дистанционных карт оценивается как  $O(N)$ , где  $N$  – количество пикселей объекта. Этот метод не гарантирует топологической корректности скелета.

Скелетизация на основе диаграмм Вороного. Напомним, что диаграммой Вороного конечного множества точек  $S$  на плоскости называется такое разбиение плоскости, при котором каждая область этого разбиения образует множество точек, более близких к одному из элементов множества  $S$ , чем к любому другому элементу множества. Если плотность граничных точек объекта стремится к бесконечности, то соответствующая диаграмма Вороного сходится к скелету. Вычислительная сложность метода с построением диаграмма Вороного оценивается как  $O(N^2 \log N)$ . Наиболее детально это направление скелетизации развито в работах научной школы Местецкого [64].

Методы топологического утоньшения строят скелет путем последовательного удаления пикселей границы объекта до тех пор, пока не будет получена необходимая толщина скелета в один пиксель. Все утоньшающие алгоритмы основываются на понятии "простой" точки. «Простая» точка – это пиксель, который может быть удален без изменения

топологии объекта. Важное свойство «простых» точек состоит в том, что они могут быть определены локально, то есть путем анализа локальной окрестности, что делает алгоритмы топологического утоньшения более эффективными.

Процесс утоньшения начинается от границы объекта и продолжается до тех пор, пока не останется «простых» точек. На каждой итерации, каждый граничный пиксель проверяется на принадлежность множеству «простых» точек. Условия обычно реализованы как логические шаблоны (маски) размера  $3 \times 3$  пикселя или более. Центр маски совмещается с рассматриваемым пикселем, анализируется окрестность этого пикселя. В случае выполнения логических условий маски центральный пиксель удаляется. основополагающей работой в направлении алгоритмов утоньшения являются работы Розенфельда [65-67]. Основная масса современных алгоритмов скелетизации – это различные варианты топологического утоньшения.

Скелетные модели представления изображения тела внутри кадра применяются во многих системах оценивания параметров движения тела, и в частности, в системах ДДР, поскольку они сохраняют зрительный образ тела и соответствуют традиционным представлениям врача-реабилитолога как ЛПР по коррекции движения.

### 1.3.2 Анализ моделей движения человека в видеопотоке

Согласно принятой концепции каждая характерная точка тела в кадре характеризуется [68]:

– положением в кадре:

$$\mathbf{p}_k = (x_k, y_k), k = 1, \dots, N \quad (1.1.)$$

где  $N$  – число кадров в видеофрагменте,

– двумя компонентами скорости (англ. velocity):

$$\mathbf{v}_k = ((x_k - x_{k-1}) / \Delta t, (y_k - y_{k-1}) / \Delta t) \quad (1.2)$$

или модулем скорости (англ. speed):

$$v_k = \sqrt{(x_k - x_{k-1})^2 + (y_k - y_{k-1})^2} / \Delta t, \quad (1.3)$$

– ускорением:

$$\mathbf{a}_k = (\mathbf{v}_k - \mathbf{v}_{k-1}) / \Delta t, \quad (\Delta t - \text{период повторения кадров}). \quad (1.4)$$

При описания динамической модели движения точки в кадре сегодня доминируют две конкурирующие модели – фильтр Калмана и скрытая марковская модель (СММ) [69-70].

Фильтр Калмана применяется для траекторного сопровождения точек кадра. Рекурсивный характер фильтра Калмана подразумевает, что расчетное положение точки рассчитывается на основании предыдущего с дискретным шагом по времени. Фильтр Калмана описывает динамику системы следующими двумя основными уравнениями [71-74]:

$$\mathbf{X}_k = \mathbf{F}_k \mathbf{X}_{k-1} + \mathbf{B}_k \mathbf{u}_k + \mathbf{w}_k \quad (1.5)$$

$$\mathbf{z}_k = \mathbf{H}_k \mathbf{X}_k + \mathbf{v}_k, \quad (1.6)$$

где  $\mathbf{X}_k$  – оцененное состояние системы на временном шаге  $k$ ,  $\mathbf{F}_k$  – матрица переходов состояний, связывающая состояние системы на временном шаге  $k$  с ее состоянием на предыдущем шаге  $k - 1$ ,  $\mathbf{B}_k$  – управляющая матрица входа, которая преобразует вектор управления  $\mathbf{u}_k$  в состояние,  $\mathbf{w}_k$  – шум измерительной системы, который моделируется многомерным нормальным

процессом с нулевым средним и ковариационной матрицей  $\mathbf{Q}_k$ .  $\mathbf{z}_k$  – наблюдаемое измерение на шаге  $k$ , которое связано с состоянием системы  $\mathbf{X}_k$  измерительной матрицей  $\mathbf{H}_k$ ,  $\mathbf{n}_k$  – шум переходного процесса, моделируемый многомерным нормальным процессом с нулевым средним и ковариационной матрицей  $\mathbf{R}_k$ , т.е.  $\mathbf{w}_k$  и  $\mathbf{n}_k$  представляются соответственно как:

$$\mathbf{w}_k \sim (0, \mathbf{Q}_k), \quad (1.7)$$

$$\mathbf{n}_k \sim (0, \mathbf{R}_k) \quad (1.8)$$

Все матрицы, имеющие индексы, могут изменяться на каждом временном шаге, но это требование не является обязательным. Кинематические уравнения выглядят так:

$$s_k = s_{k-1} + \Delta t \cdot v_k, \quad (1.9)$$

$$v_k = v_{k-1} + \Delta t \cdot a_k \quad (1.10)$$

где  $s_k$ ,  $v_k$  и  $a_k$  – положение, скорость и ускорение точки на временном шаге  $k$ .

Таким образом, имеем:

$$\mathbf{X}_k = \begin{bmatrix} s_k \\ v_k \\ a_k \end{bmatrix}, \quad (1.11)$$

$$\mathbf{H} = [0 \quad 0 \quad 1]. \quad (1.12)$$

Чтобы связать кинематические параметры на последовательных временных кадрах строится авторегрессия (АР) 1-го порядка. С учетом этого

$$\mathbf{F}_k = \begin{bmatrix} 1 & \Delta t & 0 \\ 0 & 1 & \Delta t \\ 0 & 0 & \alpha \end{bmatrix}, \quad (1.13)$$

здесь  $\alpha$  – авторегрессионный параметр, который находится при построении АР. Поскольку в задаче построения траектории движения нет выраженного управляющего вектора, можно пренебречь величинами  $\mathbf{u}_k$  и  $\mathbf{B}_k$  в уравнении фильтра Калмана (1.5). Уравнения (1.5-1.7) представляют собой задачу калмановской фильтрации для одномерного случая. Для двумерного движения они принимают вид:

$$\mathbf{X}_k = \begin{bmatrix} s_{xk} \\ s_{yk} \\ v_{xk} \\ v_{yk} \\ a_{xk} \\ a_{yk} \end{bmatrix}, \quad (1.14)$$

$$\mathbf{H}_k = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad (1.15)$$

$$\mathbf{F}_k = \begin{bmatrix} 1 & 0 & \Delta t & 0 & 0 & 0 \\ 0 & 1 & 0 & \Delta t & 0 & 0 \\ 0 & 0 & 1 & 0 & \Delta t & 0 \\ 0 & 0 & 0 & 1 & 0 & \Delta t \\ 0 & 0 & 0 & 0 & \alpha_x & 0 \\ 0 & 0 & 0 & 0 & 0 & \alpha_y \end{bmatrix}. \quad (1.16)$$

То есть для двумерного случая все переменные состояния представляются в виде проекций на оси  $X$  и  $Y$ .



Скрытая марковская модель (СММ) [75-79] является статистической моделью для оценки последовательности скрытых состояний по наблюдениям отсчетов временного ряда, она может быть записана в виде кортежа:

$$\lambda = (\mathbf{A}, \boldsymbol{\pi}, \mathbf{w}, \boldsymbol{\mu}, \boldsymbol{\sigma}), \quad (1.17)$$

где  $\mathbf{A}$  представляет собой матрицу вероятностей переходов в пространстве состояний,  $\boldsymbol{\pi}$  – начальное распределение вектора состояния,  $\mathbf{w}$  – вектор коэффициентов смеси,  $\boldsymbol{\mu}$  и  $\boldsymbol{\sigma}$  – среднее значение и среднее квадратичное отклонение (СКО) входной переменной модели. В простой марковской модели состояние может фиксироваться наблюдателем, поэтому вероятности переходов являются единственным параметром. В СММ мы можем следить лишь за переменными, на которые оказывает влияние данное состояние. Каждое состояние имеет вероятностное распределение среди всех возможных выходных значений. Поэтому последовательность значений, сгенерированная СММ, даёт информацию о последовательности состояний. Количество возможных последовательностей скрытых узлов очень велико, метод подсчёта полным перебором значений всех состояний весьма трудоемок, существуют алгоритмы, позволяющие ускорить процесс вычисления вероятности появления данной последовательности:

- алгоритм прямого-обратного хода;
- алгоритм Витерби;
- алгоритм Баума-Велча [76].

Кроме того, варианты СММ с обучением требуют дополнительного этапа обучения.

#### 1.4 Анализ методов поддержки принятия решений

Этап поддержки принятия решения в системах ППР при ДДР подразумевает сравнение оцененных по видеопоследовательностям параметров движения и расчет количественных характеристик различия движения. В большинстве известных аналогов [77-79] в качестве меры различия движения применяется евклидова мера разности параметров выполненного и эталонного движений, в частности для Kinect систем эта мера рассчитывается для множества углов  $\theta_i$ ,  $i=1, \dots, N_a$ , образованных в пространстве тройками характерных точек ( $N_a$  – общее количество углов, принимаемых в рассмотрение):

$$\Delta\theta = \frac{1}{N_a} \sum_{i=1}^{N_a} \sqrt{(\theta_i - \theta_i^0)^2}, \quad (1.18)$$

$\theta_i^0$  –  $i$ -й угловой параметр эталонного упражнения.

Далее величина  $\Delta\theta$  сравнивается с установленным порогом  $Thresh_{\Delta\theta}$ , и в случае превышения порога выдается решение о неправильном выполнении движения. Правила межкадрового накопления угловой ошибки могут быть различными (накопительное суммирование или функция максимума поккадровой ошибки). Решение, выносимое системой, как правило, бинарное: "Движение выполнено правильно/неправильно".

В системе ППР при ДДР [19] выносимые решения при ДДР более детальны. Для сравнения движений при ДДР применяется алгоритм динамической трансформации временной шкалы (DTW-алгоритм, англ. dynamic time warping) – алгоритм, позволяющий найти оптимальное соответствие и меру различия между рядами последовательных координат точек тела при сравнении эталонного движения с текущим, если они выполняются с разной скоростью. Решение выносится на основании нечеткой логики. В частности, на основании DWT получаются результаты:

– по траекториям: "Подобны" или "Не подобны";

– по скорости движения: "Слишком быстро", "Нормально", "Слишком медленно".

На выходе нечеткой логической системы выносятся тернарное решение: "Отлично", "Хорошо", "Плохо".

Тем не менее, количественные результаты в решении отсутствуют, что авторы объясняют использованием СППР в автоматическом режиме, без участия врача, как ЛПР.

### 1.5 Постановка основных задач исследования

В настоящее время наблюдается повышенный интерес к задачам разработки систем поддержки принятия решения при ДДР. Появление новых информационных технологий ППР при ДДР позволит существенно увеличить достоверность принятия решения и тем самым повысить эффективность ДДР как медицинской технологии в целом.

Противоречие между большими объемами информации, регистрируемой в процессе ДДР, и ограниченными во времени возможностями врача-реабилитолога для вынесения решения на ее основе приводит к необходимости разработки информационных технологий, позволяющих на основании анализа данных о движении пациента сформировать компактные информационные образы. Эти образы должны помочь врачу оперативно принять решение по коррекции действий пациента с высокой степенью достоверности.

Здесь можно указать нерешенные на сегодняшний день в должном объеме задачи, которые могут существенно повысить достоверность систем ППР при ДДР:

- разработка простых и в тоже время адекватных моделей движения человека и количественных показателей для описания движения пациента;
- разработка методов и количественных мер близости обучающих движений врача и повторяемых движений пациента;

– разработка и создание информационной технологии реализующей ППР в реальном масштабе времени с высокой степенью достоверности, основанной на доступных широкому контингенту потенциальных пациентов ДДР программно-аппаратных платформах.

Поэтому для повышения достоверности принимаемых решений при ДДР необходимо:

– разработать внутрикадровые модели представления тела человека и модели его движения в видеопотоке, позволяющие количественно оценить параметры движений пациента, что позволит обеспечить врача-реабилитолога необходимой информацией для принятия решений;

– разработать методы, количественно оценивающие различия в обучающем движении и движении, повторенном пациентом, что даст возможность функционирования систем ДДР в автоматическом режиме;

– разработать информационную технологию и программно-алгоритмическое обеспечение систем ППР для ДДР, в котором будут применены разработанные модели и методы, провести экспериментальное апробацию разработанной информационной технологии.

Результаты диссертационных исследований структурированы следующим образом.

Второй раздел посвящен разработке двумерной проективной точечной модели движения тела человека в виде кадрового фрагмента. Модель проста в математическом отношении и в то же время достаточно точно описывает положение тела пациента в любом ракурсе. Разработанная модель позволяет построить далее кинематическую модель движения пациента на основании анализа видеопотока. Далее получил дальнейшее развитие метод обнаружения характерных точек, который учитывает результаты логического анализа скелета с помощью скользящей маски, что позволяет повысить достоверность обнаружения характерных точек.

В третьем разделе разработана кинематическая модель движения человека в видеопотоке, которая в отличие от существующих построена на

аппроксимации траекторий характерных точек в 3-х кадровых фрагментах видеопотока дугами окружностей, что позволяет получить точечные оценки мгновенных тангенциальных скоростей и ускорений характерных точек тела. Получил дальнейшее развитие метод построения статистических оценок кинематических параметров, который основан на расчете их скользящего среднего по последовательным фрагментам видеопотока, что позволяет получить несмещенные, состоятельные и эффективные оценки координат, тангенциальных скоростей и тангенциальных ускорений ХТ тела. Разработаны метод и информационная технология поддержки принятия решения по оценке правильности движения пациента и его коррекции, которые отличаются возможностью количественного сравнения параметров обучающего и текущего движений, что позволяет повысить достоверность принятия правильного решения.

В четвертом разделе разработана информационная технология построения системы ППР при ДДР. На ее основе разработано программно-алгоритмическое обеспечение для создания СППР базе предложенных решений. Проведена экспериментальная проверка разработанной СППР и оценка ее характеристик.

#### Выводы по первому разделу

1. На основе анализа существующих систем дистанционной двигательной реабилитации (ДДР) сформулирована формальная модель принятия решений врачом-реабилитологом, как лицом принимающим решение (ЛПР), на базе общих принципов теории принятия решений. Задача принятия решений при ДДР сформулирована как задача выбора на множестве альтернатив.

2. Показано, что среди имеющихся в распоряжении ЛПР технических средств для оценки параметров движения пациента видеосредства являются наиболее предпочтительными ввиду своей низкой по сравнению с другими

средствами стоимости, удобством применения пациентом и формирования привычного для врача-реабилитолога визуального образа тела пациента..

3. Установлено, что среди видеосредств наиболее доступными являются обыкновенные бытовые видеокамеры. При применении соответствующих моделей движения тела и методов их использования видеокамеры могут обеспечить точность, достаточную для процедур ППР при ДДР.

4. Среди моделей внутрикадрового представления тела наиболее предпочтительными являются скелетные модели. Существует достаточно большое число методов построения скелетных моделей, поэтому вопрос о том, какой из них применять при ППР для ДДР, требует дополнительного исследования.

5. Применяемые в анализе движения фильтры Калмана и скрытые марковские модели мало пригодны для систем ППР при ДДР ввиду своей вычислительной сложности. Значит необходима разработка простой в вычислительном отношении и в то же время достаточно точной модели движения человека в видеопотоке.

## РАЗДЕЛ 2

### ДВУМЕРНАЯ ПРОЕКТИВНАЯ ТОЧЕЧНАЯ МОДЕЛЬ ДВИЖЕНИЯ ТЕЛА ЧЕЛОВЕКА

Задача построения внутрикадровой модели движения тела человека является в определенном смысле ключевой в разработке ИТ ППР для ДДР, поскольку она определяет все дальнейшие технологии обработки информации о движении, поступающей из видеопотока. Кроме того, внутрикадровая модель определяет необходимое множество характерных точек тела, которое с одной стороны должно достаточно детально описывать движение, а с другой стороны определяет возможность построения ИТ в реальном масштабе времени.

По результатам анализа, проведенного в первом разделе, наиболее перспективной в этом отношении является скелетная модель. Ориентировка на скелетную модель порождает задачи для исследования, связанные с выбором качественного и быстрого алгоритма скелетизации и разработкой методов построения множества характерных точек тела.

Построенную модель внутрикадрового представления тела человека необходимо проверить по качественным показателям, в частности по показателям качества обнаружения характерных точек тела

#### 2.1 Принцип построения двумерной проективной точечной модели движения тела человека

В двумерной проекции, которой является кадр видеопотока, любое положение тела человека может быть полностью описано набором характерных точек (ХТ) (рис.2.1), перечень которых приведен в таблице 2.1 [80].

Точечные модели обладает рядом преимуществ [82]:

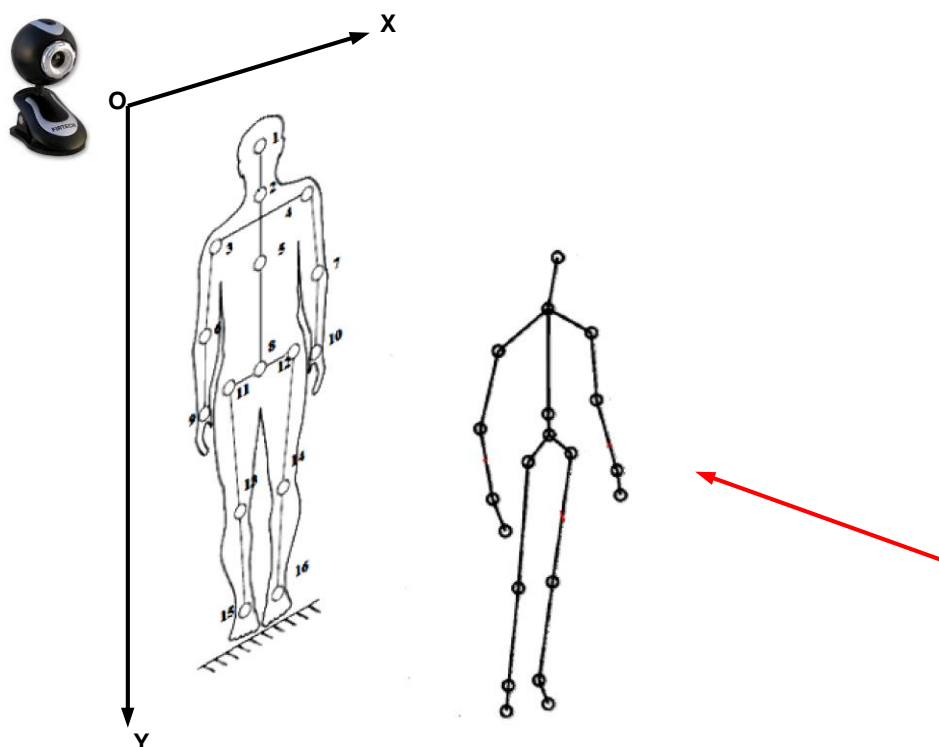


Рисунок 2.1 – Схема формирования двумерной точечной проективной модели движения тела человека

- модель сохраняет пропорции расстояний между точками при переходе от одного изображения к другому и от одного ракурса к другому, что очень важно при сравнении последовательных кадров и построения в дальнейшем модели движения;

- построение точечной проективной модели в отличие от известных является алгоритмически простым, не требует сложных математических вычислений, что важно при построении СППР для ДДР, действующих в реальном масштабе времени на компьютерном ресурсе средней производительности;

- точечная модель сохраняет привычный для ЛПР (врача-реабилитолога) зрительный образ человека-пациента, совершающего движения;

- модель позволяет обоснованно построить кинематическую модель движения пациента на основании анализа последовательных кадров видеопотока.



Таблица 2.1 – Перечень характерных точек тела человека

Номер характерной точки	Описание характерной точки	Номер характерной точки	Описание характерной точки
1	Проекция центра масс головы	9	Правый кистевой сустав
2	С7- проекция последнего шейного позвонка (центр вращения головы)	10	Левый кистевой сустав
3	Правый плечевой сустав	11	Правый бедренный сустав
4	Левый плечевой сустав	12	Левый бедренный сустав
5	5-ый грудной позвонок (примерная проекция центра масс человека)	13	Правый коленный сустав
6	Правый локтевой сустав	14	Левый коленный сустав
7	Левый локтевой сустав	15	Правый голеностопный сустав
8	L1 - проекция 1-го поясничного позвонка (центр вращения тела вокруг ножной опоры)	16	Левый голеностопный сустав

Далее будем обозначать ХТ их номером  $n$  ( $n = 1, \dots, 16$ ).

Для ракурсной проекции, отличной от нормальной (с углом зрения камеры, отличным от нуля) на изображении тела могут быть видны не все точки, но наблюдаемой совокупности практически всегда достаточно для решения задачи сравнения наблюдаемого движения с эталонным движением и оценки их отличия. С приемлемой для нашей задачи ППР при ДДР можно совместить точки 9 и 10 с центрами соответствующих ладоней, а точки 1 – с головным окончанием скелета, а 15 и 16 - с окончаниями ног человека.

Для построения внутрикадровой точечной модели тела человека требуется последовательно выполнить несколько этапов обработки кадра (рис.2.2):

- на первом этапе производится вычитание фона;
- далее производится пороговая бинаризация изображения, в результате которой получается силуэтное бинарное изображение человека.

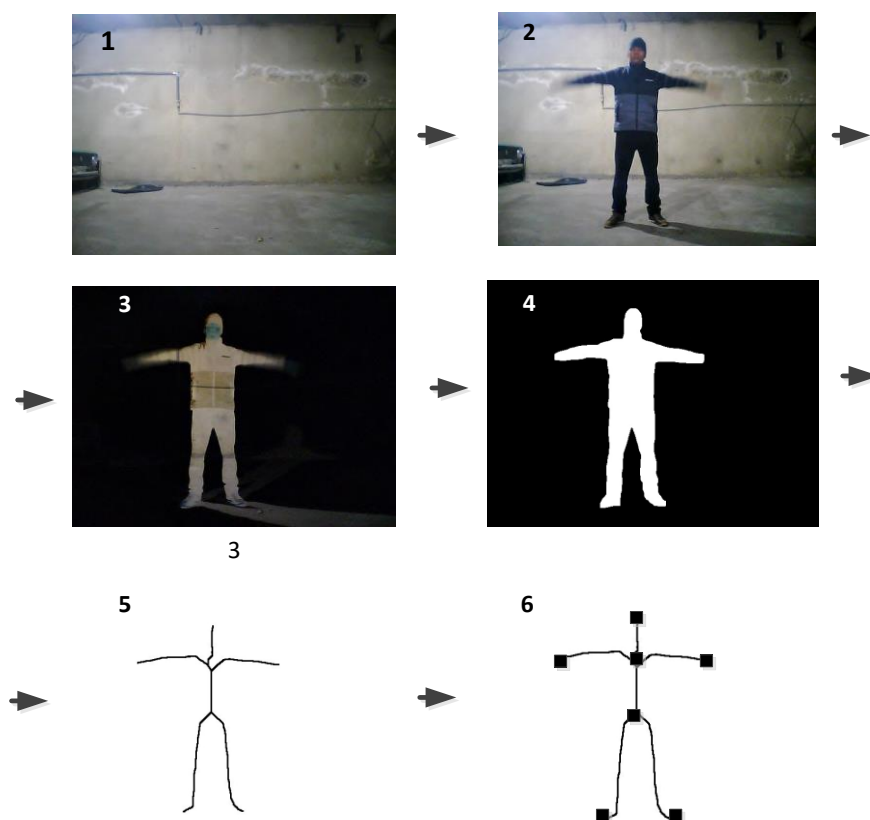


Рисунок 2.2 – Последовательные этапы построения двумерной проективной модели тела человека

## 2.2 Построение точечной модели

В ДДР пациент и врач выполняют упражнения, как правило, в помещении при постоянном освещении, на постоянном расстоянии от камеры, поэтому фоновый образ каждого кадра видеопоследовательности

остаётся одним и тем же. Его можно зарегистрировать в отсутствие пациента (врача).

На основе регистрации кадров видеопотока в отсутствии пациента строится модель фона  $F = \{F(x, y), 0 \leq x < width, 0 \leq y < height\}$  где  $width$  и  $height$  – ширина и высота кадра соответственно.

Далее, на этапе вычитания фона для формирования образа тела человека внутри каждого кадра модель фона попиксельно вычитаются интенсивности текущего кадра видеопотока и модель фона:

$$D_k(x, y) = abs(I_k(x, y) - F(x, y)) \quad (2.1)$$

Для получения бинарного контурного образа человека в кадре осуществляется пороговая бинаризация. Пикселю присваивается белый цвет, если разность интенсивности фона и текущего пикселя превышает пороговое значение:

$$Bin_k(x, y) = \begin{cases} 255, & \text{если } D_k(x, y) \geq Thr \\ 0, & \text{если } D_k(x, y) < Thr \end{cases}, \quad (2.2)$$

где  $Thr$  – экспериментально определенный порог.

Следующим этапом формирования точечной модели тела является этап скелетизации бинарного изображения тела. Покадровое построение скелета изображения тела в видеопотоке является наиболее сложным и вычислительно емким этапом.

Особыми требованиями к алгоритмам скелетизации в системах ППР для ДДР являются:

- быстродействие, поскольку система ППР при ДДР должна работать в реальном масштабе времени;
- высокое качество получаемого скелета, достаточное для безошибочного обнаружения характерных точек скелета.

Для построения модели для ДДР алгоритм скелетизации должен обеспечивать качество получаемого скелета по двум количественным критериям (отсутствие ложных ветвей скелета и отсутствие нарушений непрерывности).

Для оценки качества алгоритма скелетизации введем два показателя качества скелетизации:

1. Показатель ложных ветвей

$$K_1 = \frac{\sum_{i=1}^F length_i^{false}}{Length_0}, \quad (2.3)$$

где  $F$  – число ложных ветвей скелета,  $length_i^{false}$  – длина  $i$ -й ложной ветви скелета,  $Length_0$  – общая длина скелета.

2. Показатель разрывов скелета

$$K_2 = \frac{\sum_{i=1}^W length_i^{white}}{Length_0}, \quad (2.4)$$

где  $W$  – количество разрывов скелета,  $length_i^{white}$  – длина  $i$ -го разрыва скелета. Наиболее эффективным по критерию качества будем считать алгоритм с минимальными показателями  $K_1$  и  $K_2$ .

Были запрограммированы и испытаны на реальных изображениях тела пять алгоритмов скелетизации, являющихся наиболее популярными по данным литературных источников [83-87]:

1) Морфологическая скелетизация с последовательным применением процедур эрозии и дилатации;

2) Алгоритм Жанга-Суена – это алгоритм, работающий с логической маской размера  $3 \times 3$  и состоящий из 2 итераций. Алгоритм сохраняет связность полученного скелета;

3) Алгоритм Гуо-Холла – тоже итеративный и последовательный, также состоящий из 2 итераций, но на каждой итерации применяется своя логическая маска;

4) Алгоритм Стентифорда – он относится к итеративным параллельным алгоритмам. Число итераций – 4, алгоритм использует различных логических маски размером  $3 \times 3$ ;

5) Алгоритм Щепина-Непомнящего. Этот алгоритм не относится к группе популярных параллельных алгоритмов скелетизации, является последовательным. Алгоритм полностью сохраняет 8-икратную связность исходного изображения, причем для анализа каждой точки (с целью возможного удаления точки с сохранением связности и для выбора направления перехода) в каждый момент требуется знание только окрестности пикселя  $3 \times 3$ .

Отдельно промоделирована процедура регуляризации скелета. В работе [88] показано, что скелетизация является некорректной по Адамару задачей. Небольшие нерегулярности в границе фигуры, связанные с зашумленностью исходного изображения, практически незаметные глазом, приводят к появлению ложных или "шумовых" ветвей скелета. Для устранения ложных ветвей разработан ряд процедур его регуляризации (иногда такая процедура называется "стрижкой" скелета, англ. *pruning*). При исследовании эффективности регуляризации промоделирована такая процедура регуляризации [89]:

1) по скелету восстанавливается исходное изображение и высчитывается его площадь  $S_0$ ;

2) выбирается  $i$ -я ветвь скелета, которая одним концом соединена с другой ветвью, а другим нет, т. е. рассматривается внешняя ветвь;

3) рассчитывается площадь фигуры без выбранной ветви  $S_i$ , которая всегда будет находиться внутри восстановленного изображения, и всегда  $S_0 > S_i$  для любого  $i$ ;

4) *если*  $S_0 - S_i < Thresh$ , где  $Thresh$  – пороговая величина, то ветвь  $i$  удаляется из скелета;

5) *если* остались нерассмотренные ветви – переход к п. 2),

*иначе* конец алгоритма.

Результаты моделирования "стрижки" скелета показали, что временная трудоемкость операции регуляризации скелета примерно в 3-5 раз выше, чем трудоемкость самой процедуры скелетизации. Поэтому регуляризацию скелета следует считать неприемлемой для систем ППР при ДДР, которая должна функционировать в реальном масштабе времени, и далее в разработке ИТ она не применялась.

Усредненные показатели качества алгоритмов скелетизации, оцененные по выборке объемом 150 изображений приведен в таблице 2.2.

Таблица 2.2 – Показатели качества алгоритмов скелетизации

Примененный алгоритм	Относительное время счета	$K_1$	$K_2$
1. Морфология	1	0,07	0,34
2. Жанг-Суен	14,6	0,15	0,92
3. Гуо-Холл	18,4	0,24	0,93
4. Стентифорд	6,2	0,06	0,25
5. Щепин	62,5	0,06	0,98

На рис.2.3 приведены примеры полученных скелетов.

Анализ рис. 2.3 и таблицы 2.2 позволяет установить следующее. Быстродействующие алгоритмы (1 и 4) создают скелет неудовлетворительного качества по коэффициенту  $K_2$ , т.е. скелет имеет

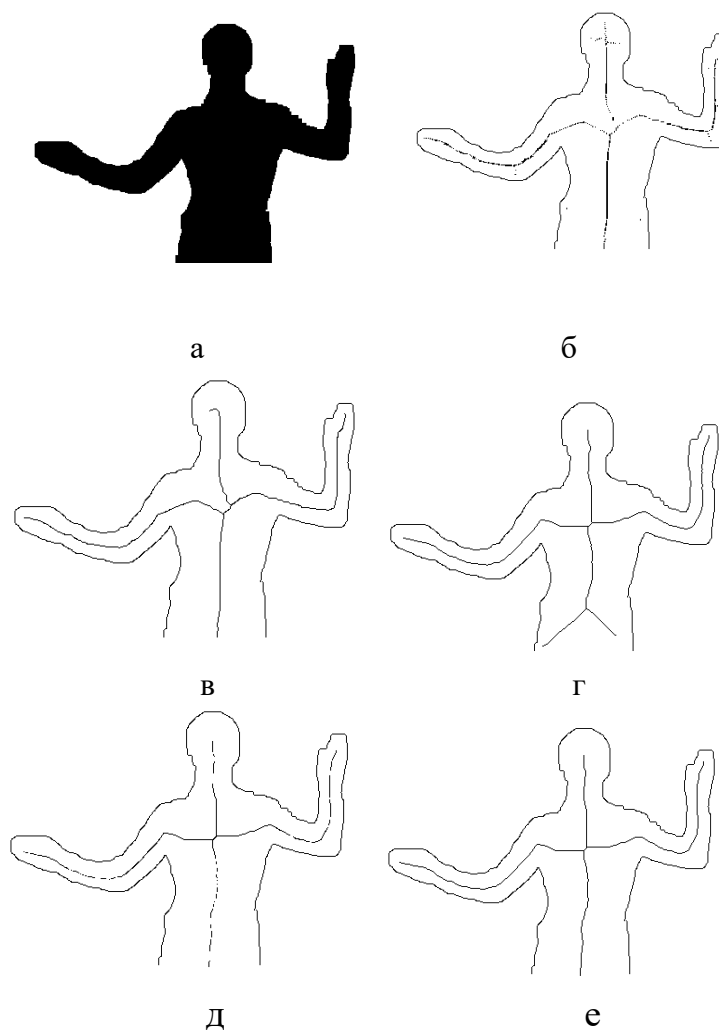


Рисунок 2.3 – Пример работы алгоритмов скелетизации: а) исходное бинаризованное изображение; б) морфологический скелет; в) скелет по ЖангСуену; г) скелет по Гуо-Холлу; д) скелет по Розенфельду, е) скелет по Щепину

большое количество разрывов. Алгоритм Гуо-Холла порождает заметные ложные ветви скелета и требует дальнейшей регуляризации. Алгоритм Щепина создает качественный скелет, но не может быть реализован в реальном масштабе времени. Алгоритм Жанг-Суена обладает лучшим сочетанием показателей  $K_1$  и  $K_2$ . Поэтому он является компромиссным вариантом и может быть рекомендован для применения в ИТ дистанционной двигательной реабилитации.

Рассмотрим процесс обнаружения характерных точек скелета.

Все ХТ скелета будем разделять на три вида:

- ХТ 1-го рода – тупиковые точки скелета (голова, кисти рук, окончания ног) каждая такая точка имеет одного «соседа» в виде черной точки;
- ХТ 2-го рода – точки ветвления скелета: каждая такая точка имеет трех или четырех «соседей» в виде черной точки;
- ХТ 3-го рода – точки излома скелета: это точки в которых ход ветви скелета заметно отличается от прямолинейного (точки изгиба линий скелета тела).

На рис. 2.4 показаны все виды ХТ скелета.

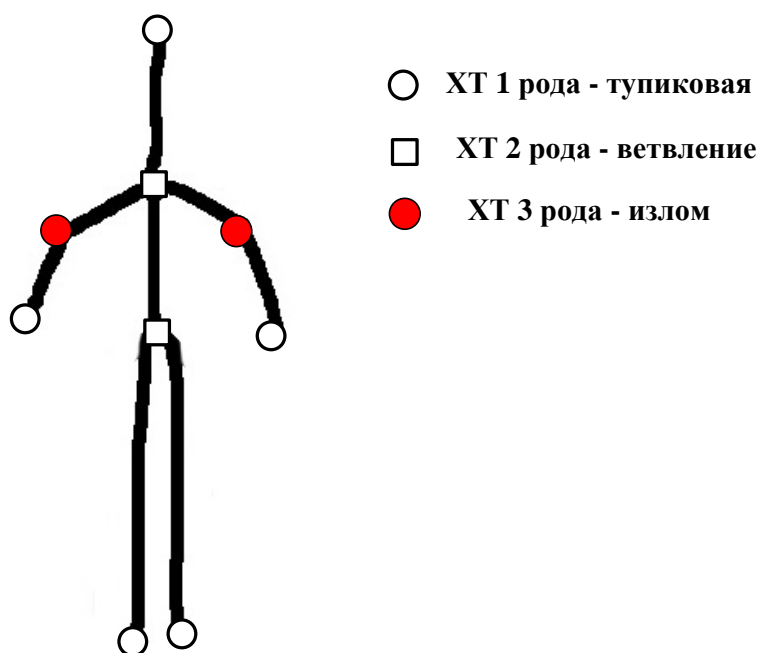


Рисунок 2.4 – Виды характерных точек скелета

Исследованы различные методы обнаружения ХТ скелета.

Одним из простейших и интуитивно понятных методик обнаружения конечных точек и точек ветвления является применение скользящих по изображению окон с размером 3\*3 пикселя и 5\*5 пикселей производится логический анализ восьмисвязной окрестности с подсчетом черных и белых пикселей и простым анализом, метод описан в различных источниках,



например в [90]. Согласно принятой терминологии называть такой метод интуитивным методом.

Если центральная точка скользящих по кадру окон размером  $3 \times 3$  и  $5 \times 5$  пикселей черная и одновременно выполняются условия:

$$\sum_{i=1}^8 N_i \in \{2,3,4\}, \quad (2.5)$$

$$\sum_{j=1}^{15} N_j \in \{2,3,4\}, \quad (2.6)$$

и

$$\sum_{i=1}^8 N_i = \sum_{j=1}^{15} N_j \quad (2.7)$$

где  $N_i$  и  $N_j$  – число черных точек во внутренней "рамке" окна размером  $3 \times 3$  пикселя и во внешней "рамке" окна размером  $5 \times 5$  пикселей соответственно, то принимается решение о том, что центральная точка окна – точка ветвления. Если же при черной центральной точке окна одновременно выполняются условия:

$$\sum_{i=1}^8 N_i = 1, \quad (2.8)$$

$$\sum_{j=1}^{15} N_j = 1, \quad (2.9)$$

и

$$\sum_{i=1}^8 N_i = \sum_{j=1}^{15} N_j \quad (2.10)$$

то принимается решение о том, что центральная точка является тупиковой.

Программная разработка и тестирование этого метода показали, недостатком такого подхода является многократное обнаружение ложных особых точек ("размножение" ложных точек), находящихся вокруг истинной, в таких случаях требуется дополнительная интерполяция соседних точек для

выяснения истинных координат особых точек. Это, в свою очередь, требует дополнительных вычислительных затрат.

Альтернативным является другой, более формализованный подход для обнаружения конечных точек и точек ветвления, основные принципы которого изложены в [91]. Рассмотрим этот подход с точки зрения разработки ИТ и ее программной реализации с некоторыми усовершенствованиями и исправлением допущенных в [91] неточностей.

Двумерная дискретная свертка изображения  $I(x, y)$  с фильтром  $G(x, y)$  размера  $M \times N$  определяется как

$$I'(x, y) = G(x, y) * I(x, y) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} G(m, n) I(x(m), y(n)), \quad (2.11)$$

где  $x(m) = x - (m - \frac{M-1}{2})$ ,

$$y(n) = y - (n - \frac{N-1}{2}).$$

Рассматривая свертку (2.11)  $I'(x, y)$  как набор откликов скользящего по изображению фильтра  $I(x, y)$ , мы можем построить конечные множества всех возможных откликов фильтра для тупиковых точек  $T_1$  и точек ветвления  $T_2$ . Тогда точки  $I'(x, y)$ , точно совпадающие по откликам с элементами  $T_1$  или  $T_2$ , являются тупиковыми или точками ветвления соответственно. Для бинарного изображения скелета можно сконструировать маску размером  $M \times N$ , состоящую только из целых двоичных степеней. Любая комбинация черных и белых точек внутри окна маски будет давать отклик, равный целой степени 2. Пример такой маски размером  $3 \times 3$  показан на рис. 2.5а. На рис. 2.5б показана зеркально отраженная маска, поскольку при выполнении свертки умножение элементов окна производится именно на отраженную маску.

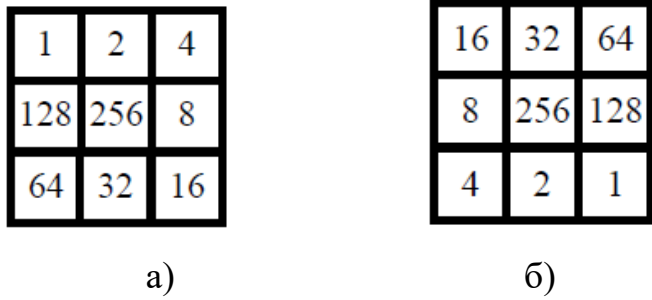


Рисунок 2.5 – прямая (а) и зеркально отраженная (б) двоичные маски размером 3×3

На рис.2.6 показаны все возможные топологические ситуации для расположения тупиковых точек и соответствующие им отклики фильтра, которые равны соответственно  $T_1 = \{257, 258, 260, 264, 272, 288, 320, 384\}$ .

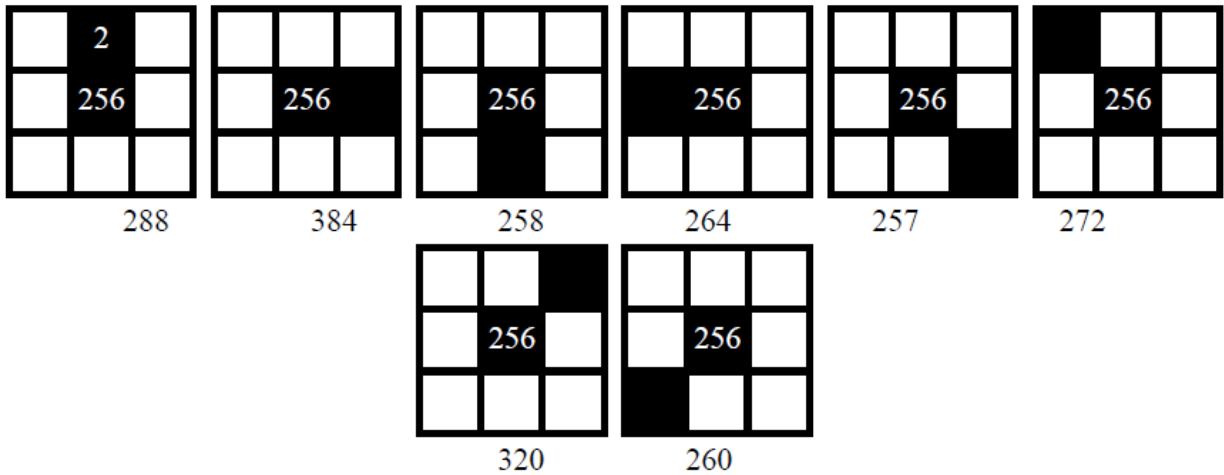


Рисунок 2.6 – Возможные топологические ситуации расположения тупиковых точек и соответствующие им отклики

На рис.2.7 показаны все возможные топологические ситуации расположения точек ветвления и соответствующие им отклики фильтра, которые равны соответственно:

$$T_2 = \{277, 293, 297, 298, 325, 329, 330, 337, 338, 340, 341, 394, 402, 404, 418, 420, 420, 424, 426 \}.$$

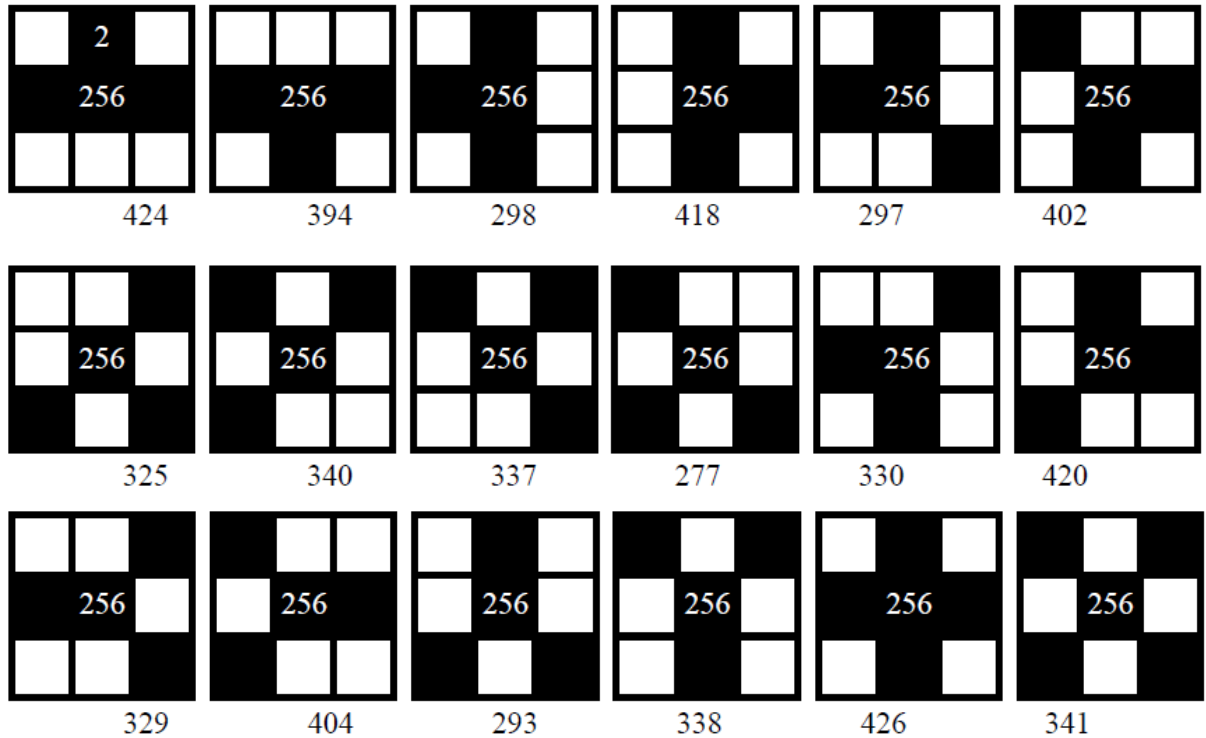


Рисунок 2.7 – Возможные топологические ситуации расположения точек ветвления и соответствующие им отклики

Были разработаны программные фрагменты, основанные на сформулированных методах и проведена их экспериментальная апробация. Для этого была сформирована база экспериментально зарегистрированных скелетов объемом 328 записей (из них 172 – в фронтальном ракурсе, 126 – в профильном ракурсе, 30 – в профильных ракурсах). На рис. 2.8 показаны фрагменты БД скелетов.

Правильность выделения особых точек проверялась визуальным анализом. В таблице 2.3 приведены результаты экспериментальной апробации интуитивного метода и метода свертки с двоичной маской для обнаружения особых точек скелета 1-го и 2-го рода.

Для обнаружения точек 3-го рода – изломов скелета – предложим следующую методику. Для обнаружения используется скользящая маска размером  $N * N$  пикселей. Пусть угол  $\alpha$  образован векторами,

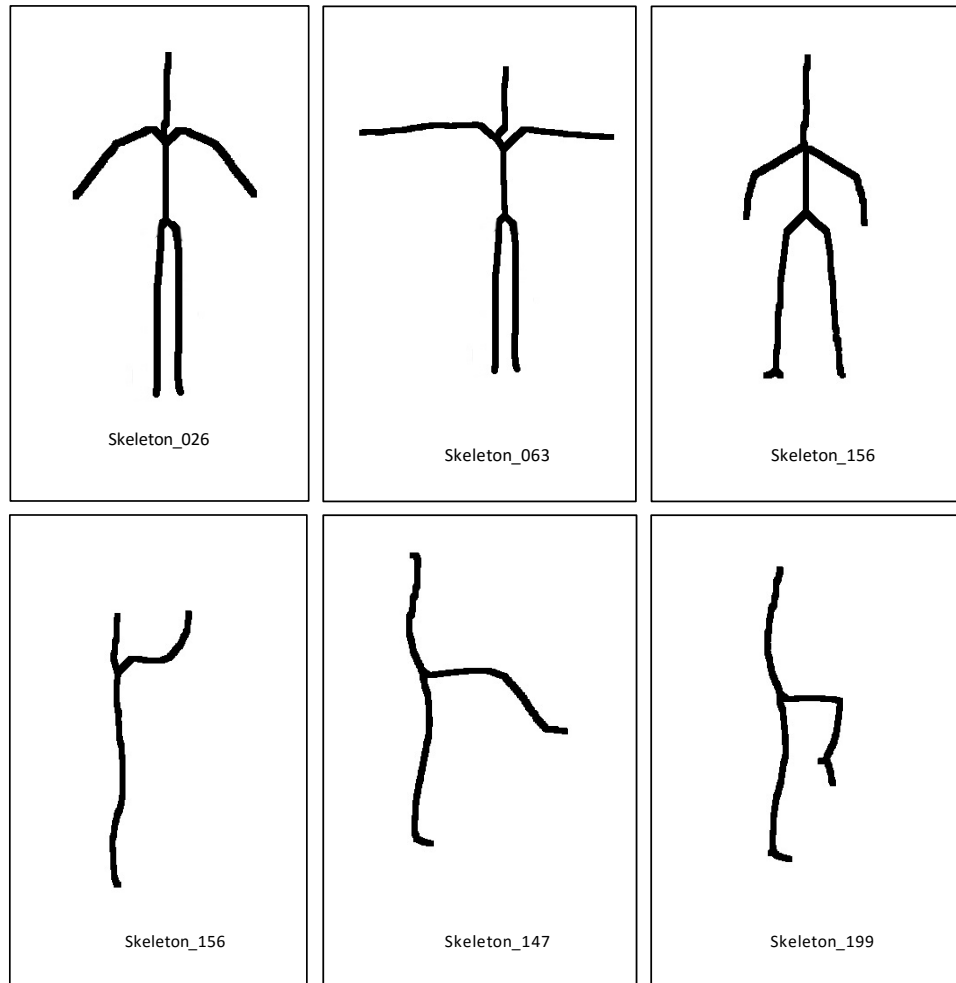


Рисунок 2.8 – Фрагменты БД скелетов

соединяющими центральный пиксель маски и точек входа черных точек скелета в маску. Если  $|\cos(\alpha)| > \cos_{Thresh}$  ( $\cos_{Thresh}$  – некоторое пороговое значение в диапазоне  $[0,8-0,99]$ ), то центральный пиксель маски отмечается как ХТ 3-го рода (рис. 2.9).

В таблице 2.4 приведены результаты экспериментального подбора параметров обнаружения ХТ 3-го рода.

Анализ таблицы 2.4 показывает, что для маски размером  $5*5$  и значения порога  $\cos_{Thresh} = 0,95$  (отмеченная строка) вероятность правильного обнаружения точки излома, полученная по выборке объемом 328 бинарных скелетов, равна 0,97, при вероятности ошибки 2-го рода не более 0,04.

Таблица 2.3 – Показатели качества обнаружения характерных точек 1-го и 2-го рода

Применяемая методика обнаружения	Максимальное время счета, мс	Минимальное время счета, мс	Вероятность ошибок 1-го рода (пропуск ХТ1 или ХТ2)	Вероятность ошибок 2-го рода (ложная ХТ1 или ХТ2)
Интуитивный метод				
Тупиковая точка (ХТ1)	110,4	64,5	0,02	0,02
Точка ветвления на 3 (ХТ2а)	92,5	73,8	0,03	0,03
Точка ветвления на 4 (ХТ2б)	76,8	63,5	0,04	0,06
Метод свертки с двоичной маской				
Тупиковая точка (ХТ1)	4,4	2,1	0,01	0,01
Точка ветвления на 3 (ХТ2а)	8,2	3,3	0,02	0,01
Точка ветвления на 4 (ХТ2б)	9,4	7,4	0,02	0,01

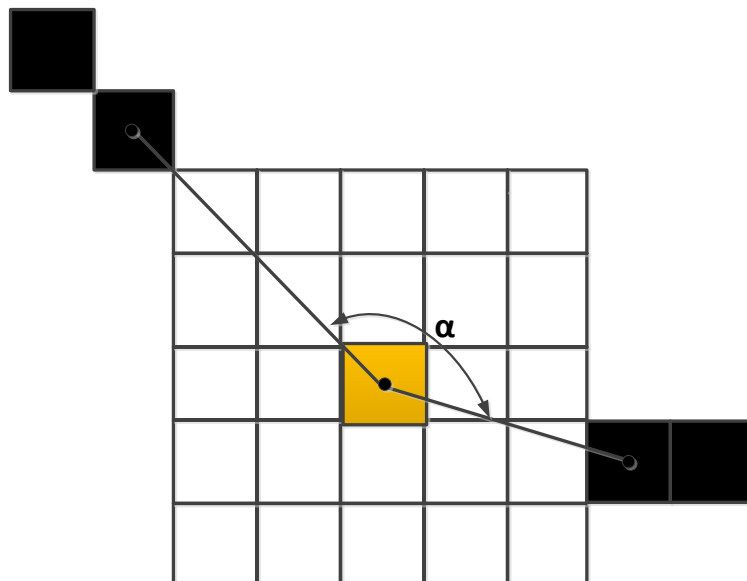


Рисунок 2.9 – Методика поиска особых точек 3-го рода

Таблица 2.4 – Экспериментальный подбор параметров поиска ХТ 3-го рода

Размер маски, пикселей	Максимальное время счета, мс	Минимальное время счета, мс	Вероятность ошибок 1 рода (пропуск ХТ3)	Вероятность ошибок 2 рода (ложная ХТ3)
Порог $\cos_{Thresh}=0,8$				
3*3	3,5	2,2	0,08	0,06
5*5	4,2	3,6	0,06	0,06
7*7	6,8	5,7	0,05	0,05
Порог $\cos_{Thresh}=0,9$				
3*3	3,9	3,1	0,05	0,04
5*5	4,2	4,0	0,04	0,04
6*6	7,5	5,7	0,04	0,04
Порог $\cos_{Thresh}=0,95$				
3*3	3,8	3,1	0,05	0,05
<b>5*5</b>	<b>4,2</b>	<b>4,1</b>	<b>0,03</b>	<b>0,04</b>
7*7	7,6	5,8	0,04	0,04
Порог $\cos_{Thresh}=0,98$				
3*3	3,7	3,1	0,06	0,06
5*5	4,2	4,0	0,04	0,04
7*7	7,9	6,1	0,05	0,05

Таким образом, метод обнаружения особых точек скелета строится на основе скользящей двоичной маски для поиска характерных точек 1-го и 2-го рода и на основе скользящей маски размером 5\*5 пикселей с пороговым значением  $\cos_{Thresh}=0,95$ . При этом метод обеспечивает вероятности правильного обнаружения при обнаружении ХТ не ниже 0,97 при вероятности ложных тревог не более 0,04.

Общие результаты исследования показателей качества метода обнаружения ХТ приведены в таблице 2.5.

Таблица 2.5 – Показатели качества алгоритма выделения ХТ

Вид ХТ	Максимальное время счета, мс	Минимальное время счета, мс	Вероятность ошибок 1 рода (пропуск ХТ)	Вероятность ошибок 2 рода (ложная ХТ)
1. Тупиковая (ХТ1)	4,4	2,1	0,01	0,01
2. Ветвление на 3 (ХТ2а)	8,2	3,3	0,02	0,01
3. Ветвление на 4 (ХТ2б)	9,4	7,4	0,02	0,01
4. Излом (ХТ3)	18,5	14,6	0,03	0,04

Таким образом установлено, что предложенный метод обеспечивает вероятности правильного обнаружения при обнаружении ХТ не ниже 0,97 при вероятности ложных тревог не более 0,04, он позволяет увеличить вероятность правильного выделения ХТ на 0,01 и уменьшить вероятность выделения ложных точек ветвления на 0,02 по сравнению с известными аналогами [90], и в отличие от них обеспечивает обнаружение всех ХТ в реальном масштабе времени.

Этих показателей качества вполне достаточно для построения ИТ СППР при ДДР.

Окончательно сформулируем усовершенствованный метод обнаружения ХТ тела внутри кадра в следующем виде:

#### **Этап 1. Вычитание фона.**

На основании регистрации кадров видеопотока без участия пациента строится модель фона

$$F = \{F(x, y), 0 \leq x < width, 0 \leq y < height\}.$$



На этапе вычитания фона попиксельно вычитаем интенсивности текущего кадра видеопотока и изображения фона

$$D_k(x, y) = \text{abs}(I_k(x, y) - F(x, y)).$$

### **Этап 2. Пороговая бинаризация.**

Пикселю присваивается белый цвет, если разность интенсивности фона и текущего пикселя превышает пороговое значение:

$$\text{Bin}_k(x, y) = \begin{cases} 255, & \text{если } D_k(x, y) \geq \text{Thr} \\ 0, & \text{если } D_k(x, y) < \text{Thr} \end{cases}.$$

### **Этап 3. Скелетизация бинарного изображения тела.**

Осуществляется скелетизация полученного бинарного изображения по алгоритму Жанг-Суена.

### **Этап 4. Обнаружение характерных точек скелета.**

ХТ 1-ого и 2-ого рода находятся путем двумерной дискретной свертки изображения скелета  $I'(x, y)$  со скользящим окном размером 3\*3 пикселя.

ХТ 3-го обнаруживаются ХТ 3-го рода путем анализа в скользящей маске размером 5\*5 пикселей углов  $\alpha$ , образованных векторами, соединяющими центральный пиксель маски и точек входа черных точек скелета в маску. Если  $|\cos(\alpha)| > \cos_{\text{Thresh}}$ , где  $\cos_{\text{Thresh}} = 0,95$ , то центральный пиксель маски метится как ХТ 3-го рода.

2.3. Базовая информационная технология обнаружения координат характерных точек тела

Суммируя результаты раздела 2 можно сформулировать разработанную и апробированную базовую информационную технологию №1 (БИТ№1) –

информационную технологию преобразования изображения (кадра) с изображением тела человека в массив координат характерных точек тела.

**Исходные данные:** кадр видеопотока в формате \*.JPEG.

**Последовательно выполняемые операции:**

1. Оценка фонового образа.
2. Вычитание из кадра фонового образа.
3. Пороговая бинаризация.
4. Скелетизация по алгоритму Жанг-Суена.
5. Обнаружение характерных точек скелета, наблюдаемых в кадре.

**Результат:** Массив координат характерных точек тела, наблюдаемых в кадре.

Выводы по второму разделу

1. Предложена двумерная проективная модель для покадрового представления движения человека в видеопотоке, которая в отличие от существующих на основании построения и обработки скелета позволяет формально описать плоский кадр с помощью 16-ти характерных точек с целью последующей статистической оценки кинематических параметров;

2. Получил дальнейшее развитие метод обнаружения характерных точек, который учитывает результаты логического анализа скелета с помощью скользящей маски, что позволяет повысить достоверность обнаружения характерных точек;

3. Установлено, что предложенный метод обнаружения характерных точек обеспечивает вероятности правильного обнаружения при обнаружении ХТ не ниже 0,97 при вероятности ложных тревог не более 0,04, он позволяет увеличить вероятность правильного выделения ХТ на 0,01 и уменьшить вероятность выделения ложных точек ветвления на 0,02 по сравнению с известными аналогами, и в отличие от них обеспечивает обнаружение всех

ХТ в реальном масштабе времени. Указанных показателей качества вполне достаточно для построения ИТ СППР при ДДР.

4. На основании предложенной точечной модели тела человека внутри кадра и метода обнаружения характерных точек разработана и апробирована базовая информационная технология №1 (БИТ №1) – информационная технология преобразования кадра с изображением тела человека в массив координат характерных точек тела.

## РАЗДЕЛ 3

### КИНЕМАТИЧЕСКАЯ МОДЕЛЬ ДВИЖЕНИЯ ЧЕЛОВЕКА В ВИДЕОПОТОКЕ, МЕТОД ПОСТРОЕНИЯ СТАТИСТИЧЕСКИХ ОЦЕНОК КИНЕМАТИЧЕСКИХ ПАРАМЕТРОВ И МЕТОД ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ ПО КОРРЕКЦИИ ДВИЖЕНИЙ ПАЦИЕНТА

Дальнейшим этапом информационной технологии ППР при ДДР является построение на основании проективной точечной модели тела модели его движения или кинематической модели на основании последовательных кадров видеопотока. Кинематическая модель должна давать последовательные оценки кинематических параметров – координат ХТ, их скоростей и ускорений в последовательные моменты времени.

Поскольку оцененные по модели кинематические параметры являются точечными оценками соответствующих случайных величин, необходимо разработать метод построения статистических оценок по последовательным кадрам потока. Построенные статистические оценки должны быть несмещенными, эффективными и состоятельными.

Следующим элементом ИТ ППР при ДДР является метод количественного сравнения обучающего движения и движения, выполненного пациентом. Именно на этом этапе появляется возможность сформировать количественно обоснованное решение по коррекции движений пациента. Количественно обоснованное решение должно повысить его степень достоверности.

#### 3.1 Кинематическая модель движения тела человека в видеопотоке

Для построения двигательной (кинематической) модели на основании точечной проективной модели рассмотрим фрагмент из  $N$  последовательных кадров видеопотока с частотой повторения кадров  $f$  (межкадровый период

$\tau = 1/f$ ). Любое движение тела с достаточной точностью может быть представлено последовательностью набора ХТ, зарегистрированных на последовательных кадрах видеопоследовательности. Предполагая, что частота кадров достаточно велика, будем считать, что каждая из ХТ, участвующая в сложном движении тела на протяжении трех последовательных кадров совершает вращательное движение. В этом случае траектория каждой ХТ в пределах трех кадров представляет собой дугу окружности. Для каждого фрагмента траектории может быть найден мгновенный центр вращения и радиус дуги (рис. 3.1) [92]. На рис.3.1 для ясности изложения показаны прореженные последовательные кадры движения тела, т.е. рисунок реально соответствует малой частоте повторения кадров).

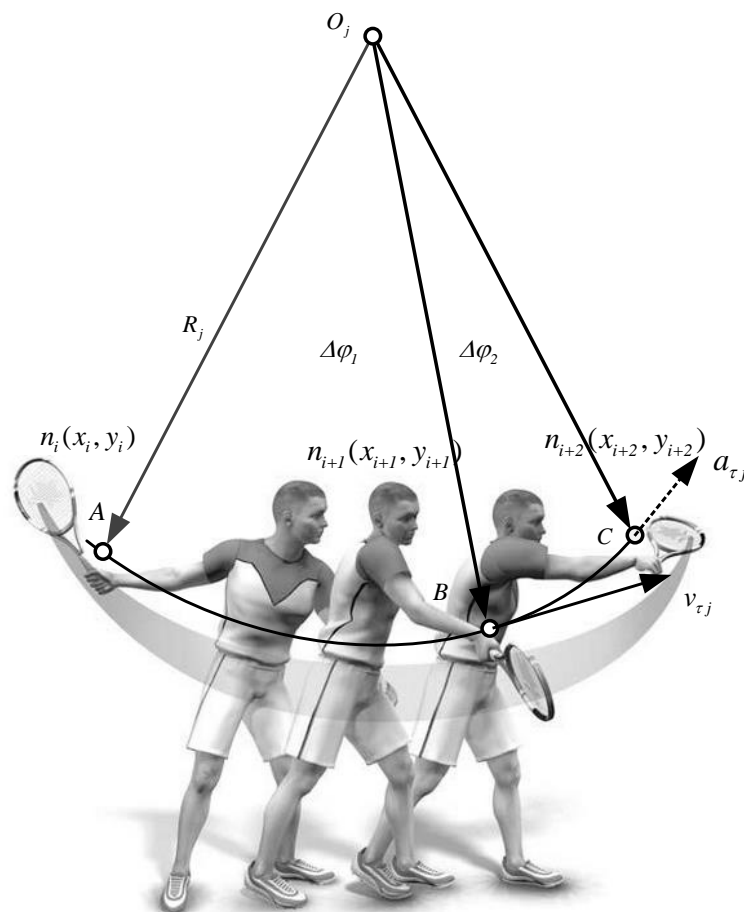


Рисунок 3.1 – Построение фрагмента траектории и оценка кинематических параметров для характерной точки

Для данного трехкадрового фрагмента  $j$  ( $j = 1, \dots, N - 2$ ) может быть рассчитано положение центра мгновенного вращения  $O_j$  по координатам трех последовательных положений ХТ  $n_i$  ( $i = 1, 2, 3$ ) в кадре -  $(x_i, y_i), (x_{i+1}, y_{i+1}), (x_{i+2}, y_{i+2})$  и радиус вращения  $R_j$  (в частности, на рис.3 построения приведены для точки с номером  $n = 9$  (правая кисть)).

Рассмотрим вспомогательную задачу построения центра окружности, проходящей через три точки  $(x_1, y_1), (x_2, y_2), (x_3, y_3)$ , и нахождения координат центра окружности  $(x_0, y_0)$  и ее радиуса  $R$  (рис. 3.2) [93].

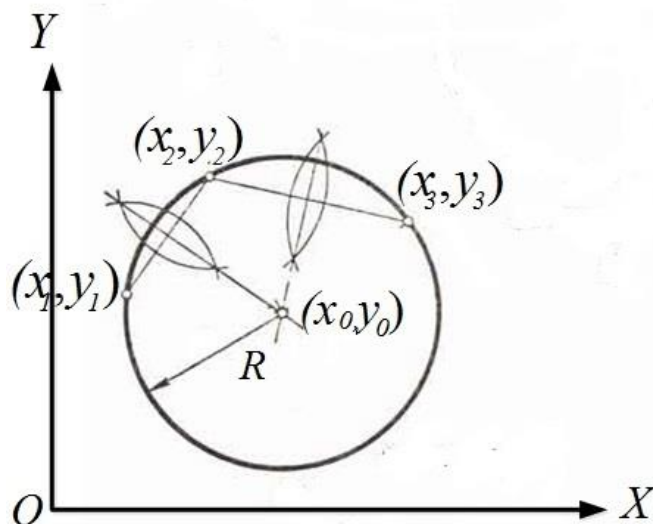


Рисунок 3.2 – Расчет параметров окружности по трем точкам

Для трех точек можно построить систему уравнений:

$$(x_1 - x_0)^2 + (y_1 - y_0)^2 = R^2, \quad (3.1)$$

$$(x_2 - x_0)^2 + (y_2 - y_0)^2 = R^2, \quad (3.2)$$

$$(x_3 - x_0)^2 + (y_3 - y_0)^2 = R^2, \quad (3.3)$$

и решить ее относительно переменных  $x_0, y_0, R$ . Система уравнений существенно нелинейна и требует определенных вычислительных затрат на свое решение.

Рассмотрим другой, более простой алгоритм решения задачи. Проведем через отрезки  $((x_1, y_1), (x_2, y_2))$  и  $((x_2, y_2), (x_3, y_3))$  срединные перпендикуляры (рис.3.2). Точка их пересечения будет центром окружности  $(x_0, y_0)$ . Координаты средин отрезков  $(x_1, y_1), (x_2, y_2)$  и  $(x_2, y_2), (x_3, y_3)$  можно выразить так:

$$x_{12}^c = ((x_1 + x_2)/2), (y_1 + y_2)/2), \quad (3.4)$$

$$x_{23}^c = ((x_2 + x_3)/2), (y_2 + y_3)/2) \quad (3.5)$$

Уравнения перпендикуляров можно записать так:

$$(x_2 - x_1)(x_0 - x_{12}^c) + (y_2 - y_1)(y_0 - y_{12}^c) = 0, \quad (3.6)$$

$$(x_3 - x_2)(x_0 - x_{23}^c) + (y_3 - y_2)(y_0 - y_{23}^c) = 0, \quad (3.7)$$

или в развернутом виде

$$(x_2 - x_1)x_0 + (y_2 - y_1)y_0 = (x_2 - x_1)x_{12}^c + (y_2 - y_1)y_{12}^c, \quad (3.8)$$

$$(x_3 - x_2)x_0 + (y_3 - y_2)y_0 = (x_3 - x_2)x_{23}^c + (y_3 - y_2)y_{23}^c, \quad (3.9)$$

Вводя обозначения

$$A_1 = (x_2 - x_1), B_1 = (y_2 - y_1), C_1 = (x_2 - x_1)x_{12}^c + (y_2 - y_1)y_{12}^c, \quad (3.10)$$

$$A_2 = (x_3 - x_2), B_2 = (y_3 - y_2), C_2 = (x_3 - x_2)x_{23}^c + (y_3 - y_2)y_{23}^c, \quad (3.11)$$

перепишем систему в виде:

$$A_1 x_0 + B_1 y_0 = C_2, \quad (3.12)$$

$$A_2 x_0 + B_2 y_0 = C_2, \quad (3.13)$$

Решение системы есть:

$$x_0 = (C_1 B_2 - C_2 B_1) / (A_1 B_2 - A_2 B_1), \quad y_0 = (A_1 C_2 - A_2 C_1) / (A_1 B_2 - A_2 B_1) \quad (3.14)$$

Программная разработка алгоритмов показала, что вычисления по второму, "геометрическому" алгоритму требует примерно в 80-90 раз меньше расчетного времени, что существенно при реализации кинематической модели в реальном масштабе времени.

Для известного радиуса вращательного движения можно оценить модули тангенциальной скорости  $v_{\tau j}$  и тангенциального ускорения  $a_{\tau j}$  [94]:

$$v_{\tau j} = \omega_j R_j, \quad (3.14)$$

$$a_{\tau j} = \varepsilon_j R_j, \quad (3.15)$$

где  $\omega_j$  и  $\varepsilon_j$  – угловая скорость и угловое ускорение точки  $n$  для кадрового фрагмента  $j$ .

Угловую скорость можно оценить на участке траектории  $AB$  как

$$\omega_{j1} = \frac{\Delta \varphi_l}{\tau}, \quad (3.16)$$

на участке траектории  $BC$  как



$$\omega_{j2} = \frac{\Delta\varphi_2}{\tau}, \quad (3.17)$$

где  $\Delta\varphi_1$  и  $\Delta\varphi_2$  – угловые приращения положения точки  $n$  от кадра к кадру,  $\tau$  – период повторения кадров.  $\Delta\varphi_1$  и  $\Delta\varphi_2$  находятся как углы между векторами  $O_jA$ ,  $O_jB$  и  $O_jB$ ,  $O_jC$  с известными координатами. Угловая скорость для фрагмента  $j$   $\omega_j$  оценивается как среднее арифметическое  $\omega_{j1}$  и  $\omega_{j2}$ . Угловое ускорение для фрагмента  $j$   $\varepsilon_j$  рассчитывается как

$$\varepsilon_j = \frac{\Delta\varepsilon_j}{\Delta t} = \frac{\omega_{j2} - \omega_{j1}}{\tau}, \quad (3.18)$$

а линейное ускорение как

$$a_{\tau j} = \varepsilon_j R_j. \quad (3.19)$$

Таким образом для последовательности  $N$  кадров из видеопотока можно построить для каждой точки  $n$ , наблюдаемой в кадрах  $(N-2)$  столбца, описывающих движение тела в двумерной проекции, представляющих собой набор координат точки, ее тангенциальную скорость и тангенциальное ускорение.

Отметим, что при малой кривизне фрагмента траектории ХТ координаты мгновенного центра вращения  $O_j(x_{oj}, y_{oj})$  и величина радиуса вращения  $R_j$  могут выходить за пределы кадра в кадровых пиксельных координатах в любой квадрант, смежный с кадром. Тем не менее, такое "расширение кадра" не влияет на результаты расчетов мгновенных кинематических параметров.

Таким образом, модель движения тела в рамках фрагмента видеопоследовательности из  $N$  кадров можно представить в виде следующей матрицы.

$$\left( \begin{array}{cccc} 1 & 1,2,3 & x_1^1, y_1^1 v_{\tau 1}^1 a_{\tau 1}^1 & \dots x_1^n, y_1^n v_{\tau 1}^n a_{\tau 1}^n & \dots x_1^{16}, y_1^{16} v_{\tau 1}^{16} a_{\tau 1}^{16} \\ 2 & 2,3,4 & x_2^1, y_2^1 v_{\tau 2}^1 a_{\tau 2}^1 & \dots x_2^n, y_2^n v_{\tau 2}^n a_{\tau 2}^n & \dots x_2^{16}, y_2^{16} v_{\tau 2}^{16} a_{\tau 2}^{16} \\ \dots & \dots & \dots & \dots & \dots \\ j & i, i+1, i+2 & x_j^1, y_j^1 v_{\tau j}^1 a_{\tau j}^1 & \dots x_j^n, y_j^n v_{\tau j}^n a_{\tau j}^n & \dots x_j^{16}, y_j^{16} v_{\tau j}^{16} a_{\tau j}^{16} \\ \dots & \dots & \dots & \dots & \dots \\ K & N-2, N-1, N & x_K^1, y_K^1 v_{\tau K}^1 a_{\tau K}^1 & \dots x_K^n, y_K^n v_{\tau K}^n a_{\tau K}^n & \dots x_K^{16}, y_K^{16} v_{\tau K}^{16} a_{\tau K}^{16} \end{array} \right) \quad (3.20)$$

Опишем структуру построения матрицы. Первый элемент строки - номер трехкадрового фрагмента  $j$  ( $j = 1, 2, \dots, K = N - 2$ ), поскольку для построения первого фрагмента используются три кадра). Второй элемент строки  $i, i+1, i+2$  - номера кадров, обрабатываемых в  $j$ -м фрагменте ( $i = 1, 2, \dots, N$ ). Последующие элементы строки связаны с координатами, скоростями и ускорениями ХТ, поэтому в качестве верхнего индекса они содержат номер  $n$  ХТ ( $i = 1, \dots, 16$ ). В частности,  $x_j^n, y_j^n$  - координаты точки  $n$  в середине фрагмента  $j$ ,  $v_{\tau j}^n, a_{\tau j}^n$  - касательные скорость и ускорение точки  $n$  соответственно, рассчитанные для фрагмента  $j$  видеопотока.

Матрица (3.20) не всегда будет полноразмерной по  $n$ , поскольку все ХТ могут наблюдаться только при движении всего тела в полном фронтальном ракурсе. При любых угловых ракурсах и объемных движениях наблюдается только часть ХТ, т.е. при таких ракурсах в матрице, начиная с третьего столбца, будут участвовать только элементы с верхними индексами, соответствующими реально обнаруженным в кадре точкам.

### 3.2 Метод построения статистических оценок кинематических параметров

Поскольку оцененные по кадру  $i$  координаты ХТ  $x_i^n, y_i^n$  и оцененные по 3-х кадровому фрагменту скорости и ускорения ХТ  $v_i^n$  и  $a_i^n$  ( $n = 1, \dots, 16$ ) являются точечными оценками случайных величин, требуется их скользящее усреднение по последовательным фрагментам.

В существующих системах оценки параметров движения по траекториям движения точки [95-97] применяется следующий метод построения усредненных статистических оценок параметров движения. С помощью скользящих средних сглаживается траектория движения точки интереса. Далее скорость и ускорение рассчитываются по соотношениям:

$$v_i = \frac{x_i - x_{i-1}}{\Delta t}, \quad (3.21)$$

Или

$$v_i = \frac{3x_i - 4x_{i-1} + x_{i-2}}{4\Delta t}, \quad (3.22)$$

$$a_i = \frac{v_i - v_{i-1}}{\Delta t}. \quad (3.23)$$

Соотношения (3.21-3.23) показывают, что фактически скорость и ускорение точек рассчитываются путем численного дифференцирования. Численное дифференцирование экспериментальных данных, содержащих шумы, является некорректной процедурой, требующей регуляризации, например по Тихонову [98,99]. Это вызывает дополнительные серьезные осложнения методов получения статистических оценок. Поскольку в предложенной в разделе 3.1, кинематической модели оцениваются не только

координаты, но и скорость и ускорение, рассмотрим такую модификацию метода построения статистических оценок кинематических параметров.

Метод состоит из трех последовательных этапов.

**Этап 1.** Для видеопоследовательности длиной  $N$  кадров рассчитываются матрицы  $\mathbf{M}_i$

$$\mathbf{M}_i = \left\{ \begin{array}{cccccc} x_{i-1}^1, y_{i-1}^1 & v_{\bar{a}}^1 & a_{\bar{a}}^1 & \dots & x_{i-1}^n, y_{i-1}^n & v_{\bar{a}}^n & a_{\bar{a}}^n & \dots & x_{i-1}^{16}, y_{i-1}^{16} & v_{\bar{a}}^{16} & a_{\bar{a}}^{16} \\ x_i^1, y_i^1 & v_{\bar{a}}^1 & a_{\bar{a}}^1 & \dots & x_i^n, y_i^n & v_{\bar{a}}^n & a_{\bar{a}}^n & \dots & x_i^{16}, y_i^{16} & v_{\bar{a}}^{16} & a_{\bar{a}}^{16} \\ x_{i+1}^1, y_{i+1}^1 & v_{\bar{a}}^1 & a_{\bar{a}}^1 & \dots & x_{i+1}^n, y_{i+1}^n & v_{\bar{a}}^n & a_{\bar{a}}^n & \dots & x_{i+1}^{16}, y_{i+1}^{16} & v_{\bar{a}}^{16} & a_{\bar{a}}^{16} \end{array} \right\}$$

для трехкадровых фрагментов видеопоследовательности с номерами  $i = 1, \dots, (N - 2)$ .

**Этап 2.** Оценки координат, тангенциального ускорения и тангенциальной скорости ХТ рассчитываются как скользящее среднее по  $W$  фрагментам [100-102]:

$$\hat{x}_j^n = 1/W \sum_{k=j-W+1}^j x_k^n, \quad (3.25)$$

$$\hat{y}_j^n = 1/W \sum_{k=j-W+1}^j y_k^n, \quad (3.26)$$

$$\hat{v}_{\bar{a}}^n = 1/W \sum_{k=j-W+1}^j v_{\bar{a}}^n, \quad (3.27)$$

$$\hat{a}_{\bar{a}}^n = 1/W \sum_{k=j-W+1}^j a_{\bar{a}}^n, \quad (3.28)$$

где  $j = \lceil W/2 \rceil, \dots, L = N - \max\{2, \lceil W/2 \rceil\}$ .

Диапазон изменения индекса оценки  $j$  связан с тем, что в векторе оценки скользящего среднего необходимо отбросить  $\lceil W/2 \rceil$  начальных и конечных "переходных" результатов (здесь символ  $\lceil \dots \rceil$  обозначает целую часть с округлением вверх).

**Этап 3.** Из полученных статистических оценок  $\hat{x}_j^n, \hat{y}_j^n, \hat{v}_{\tau_j}^n, \hat{a}_{\tau_j}^n$  строится матрица статистических оценок кинематических параметров  $\hat{\mathbf{M}}$ , описывающая реальное движение, зафиксированное в видеофрагменте

$$\hat{\mathbf{M}} = \left\{ \begin{array}{cccccccccccc} \hat{x}_{\lceil w/2 \rceil}^1, \hat{y}_{\lceil w/2 \rceil}^1, \hat{v}_{\tau_{\lceil w/2 \rceil}}^1, \hat{a}_{\tau_{\lceil w/2 \rceil}}^1, \dots, \hat{x}_{\lceil w/2 \rceil}^n, \hat{y}_{\lceil w/2 \rceil}^n, \hat{v}_{\tau_{\lceil w/2 \rceil}}^n, \hat{a}_{\tau_{\lceil w/2 \rceil}}^n, \dots, \hat{x}_{\lceil w/2 \rceil}^{16}, \hat{y}_{\lceil w/2 \rceil}^{16}, \hat{v}_{\tau_{\lceil w/2 \rceil}}^{16}, \hat{a}_{\tau_{\lceil w/2 \rceil}}^{16} \\ \dots & & \dots & & \dots & & \dots & & \dots & & \dots & & \dots & & \dots \\ \hat{x}_L^1, \hat{y}_L^1, \hat{v}_{\tau_L}^1, \hat{a}_{\tau_L}^1, \dots, \hat{x}_L^n, \hat{y}_L^n, \hat{v}_{\tau_L}^n, \hat{a}_{\tau_L}^n, \dots, \hat{x}_L^{16}, \hat{y}_L^{16}, \hat{v}_{\tau_L}^{16}, \hat{a}_{\tau_L}^{16} \end{array} \right\} \quad (3.29)$$

### 3.3 Проверка адекватности построенной модели и метода усреднения кинематических параметров

Для подтверждения адекватности разработанной кинематической модели и метода усреднения кинематических параметров необходимо показать, что оценки получаемых кинематических параметров т.е. координат ХТ, их скоростей и ускорений являются [103]:

- несмещенными,
- состоятельными,
- эффективными.

Для получения показателей качества предложенной модели применена методика синтеза эталонных видеопоследовательностей. По теоретическим соотношениям синтезируется последовательность кадров, представляющих движение одной точки в плоскости  $(x, y)$  с известными в каждый момент времени координатами, тангенциальной скоростью и тангенциальным ускорением.

Поскольку для синтезированной видеопоследовательности точно известны все кинематические параметры:  $x_0, y_0, v_{\tau_0}, a_{\tau_0}$ , для видеопоследовательностей можно оценить смещение оценки соответствующего кинематического параметра  $k\hat{p}$  ( $k\hat{p} \in \{x, y, v_{\tau}, a_{\tau}\}$ ):

$$\Delta k\hat{p} = M(k\hat{p} - kp_0) = \frac{1}{N} \sum_{i=1}^N (k\hat{p}_i - kp_0), \quad (3.30)$$

где  $M(\dots)$  – символ математического ожидания,  $kp_0$  – истинное значение кинематического параметра в кадре с номером  $N$ ,  $k\hat{p}_i$  – оценка кинематического параметра по текущему  $i$ -му кадру видеопоследовательности, и дисперсию его оценки :

$$D(k\hat{p}) = \sigma^2(k\hat{p} - kp_0) = \frac{1}{N} \sum_{i=1}^N (k\hat{p}_i - kp_0)^2. \quad (3.31)$$

Для подтверждения несмещенности оценки необходимо показать, что смещение оценки  $k\hat{p}$  стремится к нулю при увеличении числа усредняемых кадров, для подтверждения состоятельности – показать, что дисперсия оценки  $k\hat{p}$  стремится к нулю при увеличении числа усредняемых кадров, и для подтверждения эффективности оценки  $k\hat{p}$  – показать, что предложенная в методе оценка скользящего среднего сходится по вероятности к истинному значению быстрее других несмещенных оценок.

Для проверки адекватности кинематической модели применена методика синтеза траекторной видеопоследовательности одиночной движущейся точки. В частности, мы всегда можем покадрово искусственно сформировать видеопоток, описывающий движение точки с известными параметрами, например:

$$t = [0, \Delta t, 1c], \quad \Delta t = 0,04c, \quad R_0 = 1m, \quad \omega = 0,031c^{-1}, \quad \varepsilon = 8,4 \cdot 10^{-4} c^{-2}.$$

Кинематические параметры такой траекторной видеопоследовательности рассчитаны по соотношениям:

$$x = R(t) \cos(\omega t + \varepsilon t^2),$$

$$y = R(t) \sin(\omega t + \varepsilon t^2),$$

$$\varphi = \omega t + \varepsilon t^2,$$

$$v_{\tau} = R(t)(\omega + \varepsilon t),$$

$$a_{\tau} = R(t)\varepsilon.$$

Кадрограмма (последовательность наложенных кадров) видеопоследовательности приведена на рис.3.3.

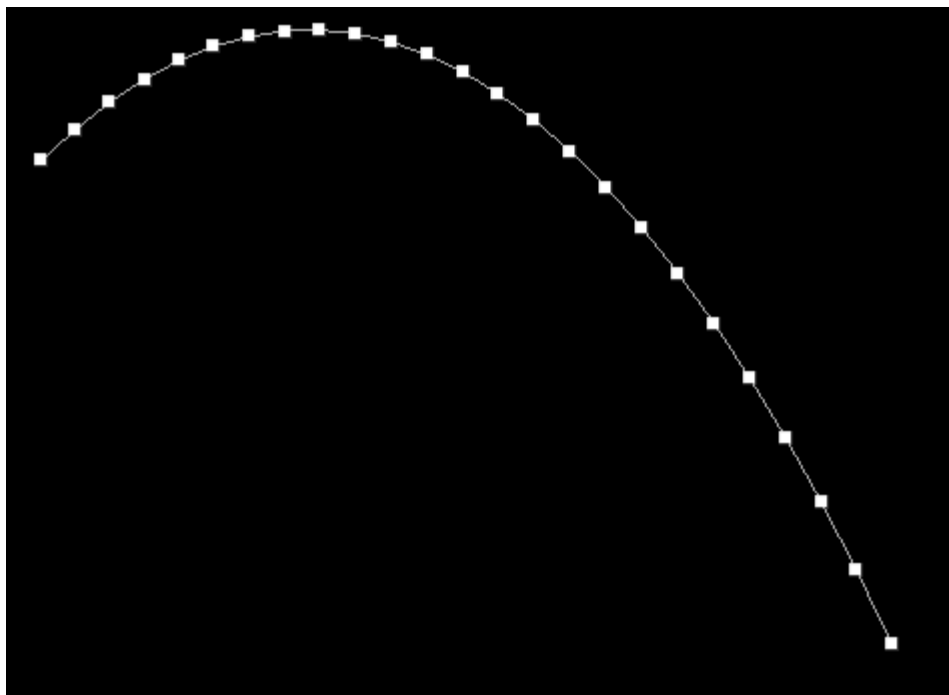


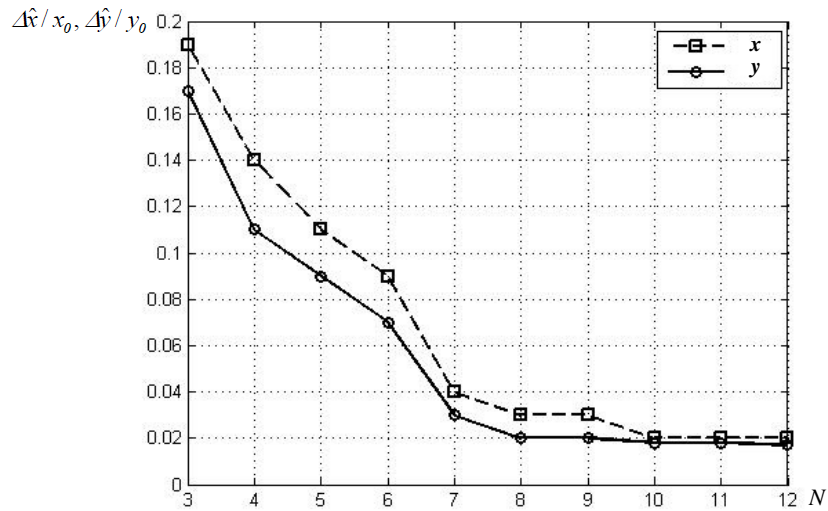
Рисунок 3.3 – Кадрограмма синтезированной траекторной видеопоследовательности с параметрами  $t = [0, \Delta t, 1 \text{ с}]$ ,  $\Delta t = 0,04 \text{ с}$ ,  $R_0 = 1 \text{ м}$ ,

$$\omega = 0,031 \text{ с}^{-1}, \varepsilon = 8,4 \cdot 10^{-4} \text{ с}^{-2}$$

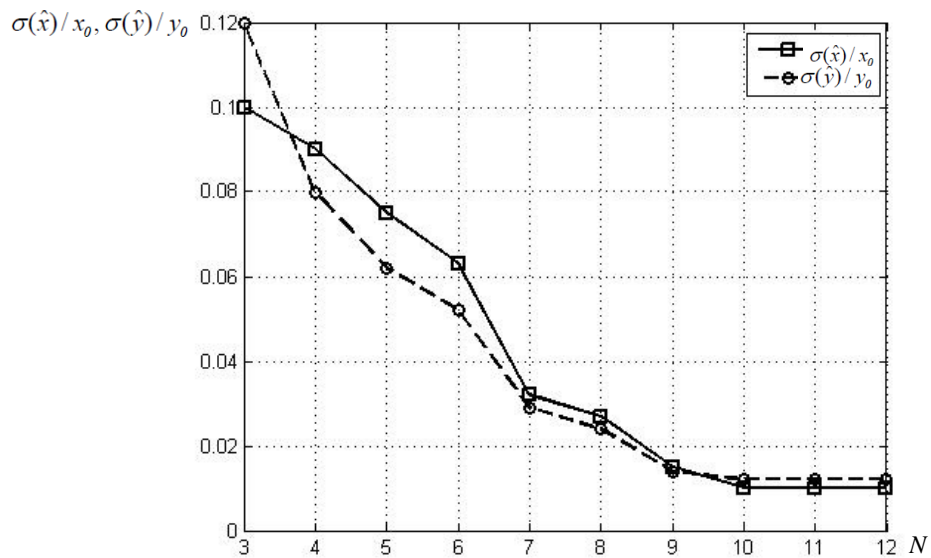
Были проведены вычислительные эксперименты с 60-ю синтезированными траекториями. В процессе экспериментов оценивались смещение оценок кинематических параметров, коэффициенты вариации их

оценок. а также оценок сходимость по вероятности к истинным значениям, которые для синтезированных траекторий известны абсолютно точно.

Типичные результаты экспериментов для траектории, изображенной на рис. 3.3 показаны ниже на рис. 3.4–3.6.



а)



б)

Рисунок 3.4 – Зависимости нормированного смещения оценок координат  $x$  (а) и  $y$  (б) от числа усредняемых последовательных кадров  $N$  для траектории, показанной на рис.3.3



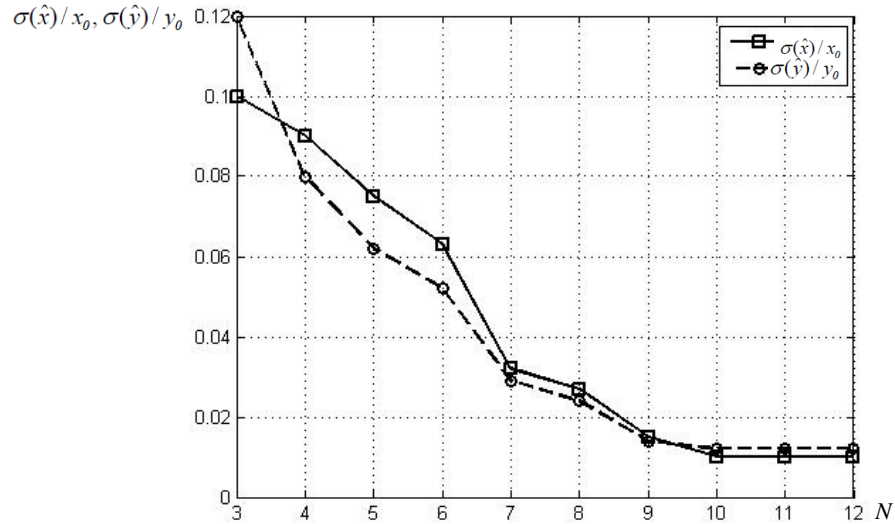


Рисунок 3.5 – Зависимость коэффициента вариации оценок координат от числа усредняемых последовательных кадров  $N$  для траектории, показанной на рис.3.3

В целом для экспериментов с 60 синтезированными траекториями становлено, что смещение оценок координат уменьшается с увеличением числа усредняемых видео фрагментов и не превышает 1,9 %.

Коэффициенты вариации оценок координат уменьшаются с увеличением числа усредняемых видео фрагментов. В целом установлено, что коэффициенты вариации оценок координат ХТ не превышают 3,2%,

На рис. 3.6 приведены результаты усреднения оценок кинематических параметров как скользящего среднего (МА):

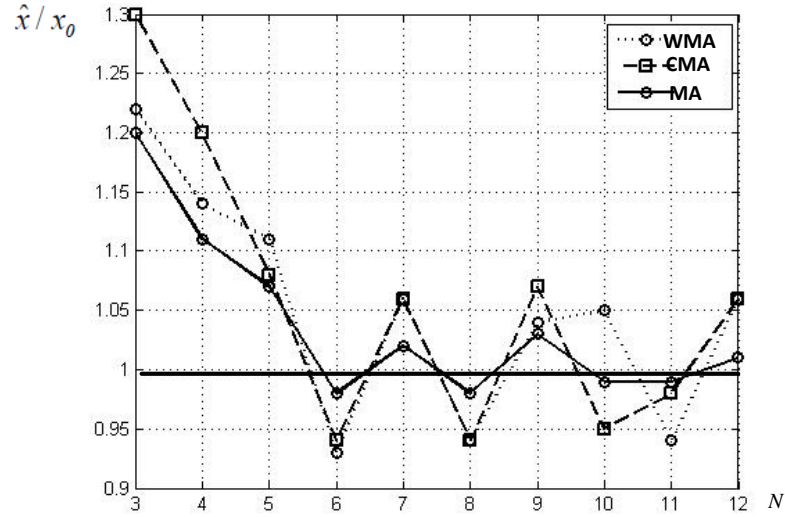
$$MA(k\hat{p}_j) = 1/W \sum_{k=j-W+1}^j k\hat{p}_k, \quad (3.31)$$

где  $j = \lceil W/2 \rceil, \dots, L = N - \max\{2, \lceil W/2 \rceil\}$ ,

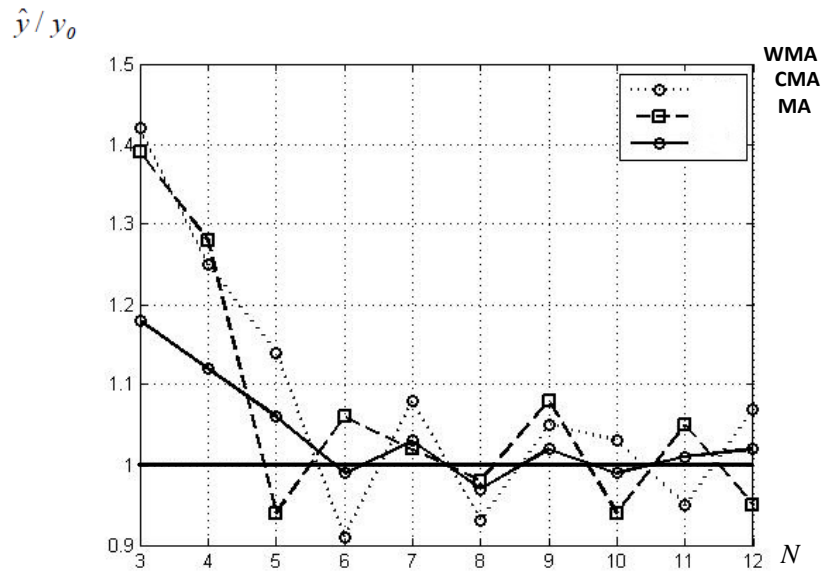
взвешенного скользящего среднего (WMA):

$$WMA(k\hat{p}_j) = 1/j(j-1) \sum_{k=j-W+1}^j k\hat{p}_k(j-k), \quad (3.32)$$

где  $j = \lceil W/2 \rceil, \dots, L = N - \max\{2, \lceil W/2 \rceil\}$ ,



а)



б)

Рисунок 3.6 – Зависимости сходимости оценки координат  $x$  (а) и  $y$  (б) к истинному значению в зависимости от числа усредняемых последовательных кадров  $N$  для траектории, показанной на рис.3.3

и кумулятивного среднего (СМА)

$$CMA(k\hat{p}_j) = 1/j \sum_{k=j}^j k\hat{p}_k, \quad (3.33)$$

$$j=0, \dots, L=N-2.$$

Аналогичные зависимости наблюдаются и для кинематических параметров  $v_{\tau}, a_{\tau}$ .

В процессе моделирования статистического усреднения оценок кинематических параметров установлено, что:

- смещения оценок координат, скорости и ускорения уменьшаются с увеличением числа усредняемых видео фрагментов и составляют в целом не более 1,9% по координатам, 2,1% по скорости и 5,1% по ускорению по отношению к истинным значениям при усреднении 10 фрагментов потока и более;

- коэффициенты вариации оценок координат, скорости и ускорения уменьшаются с увеличением числа усредняемых видео фрагментов и не превосходят 3,2% по координатам, 4,1% по скорости и 4,8 % по ускорению при усреднении 10 фрагментов потока и более.

- оценки координат, скорости и ускорения движущейся точки по кинематической модели сходятся по вероятности к истинному значению быстрее других исследованных моделей усреднения (СМА, WMA), что подтверждает состоятельность оценок.

Т.е. показано, что при расчете скользящего среднего не менее, чем по 10 фрагментам (или 12 последовательным кадрам) видеопотока оценки параметров практически не смещены, состоятельны и эффективны. Таким образом экспериментально подтверждены адекватность построенной кинематической модели и метода построения статистических оценок кинематических параметров.

### 3.4 Базовая информационная технология построения кинематической модели и оценивания кинематических параметров

На основании предложенной кинематической модели и метода оценки кинематических параметров можно сформулировать разработанную и апробированную базовую информационную технологию №2 (БИТ2) –

технологии преобразования кадров видеопотока с изображением движения тела в кинематическую матрицу параметров движения.

**Исходные данные:** Структура данных, содержащая массивы координат характерных точек тела для  $N$  последовательных кадров видеопотока.

**Последовательно выполняемые операции:**

1. Аппроксимация сегмента траектории каждой последовательной тройки характерных точек дугами окружностей.

2. Определение центров и радиусов окружности для каждого сегмента для каждой характерной точки.

3. Оценка тангенциальной скорости и тангенциального ускорения каждой характерной точки в каждом последовательном трехкадровом сегменте видеопотока.

4. Усреднение полученных оценок кинематических параметров характерных точек скользящим средним по последовательности длиной не менее 10 кадров.

**Результат:** Матрица статистических оценок кинематических параметров каждой наблюдаемой в видеопотоке характерной точки.

### 3.5 Оценка энтропии оценивания кинематических параметров движения по разработанной модели

С точки зрения построения ИТ ППР при ДДР представляет интерес величина уменьшения энтропии оценивания параметров движения с применением разработанной кинематической модели. Величина уменьшения энтропии является косвенной оценкой повышения достоверности принимаемого решения о правильности выполненного движения и его коррекции.

Для определения величины уменьшения энтропии оценивания параметров движения был поставлен следующий эксперимент. Для 50

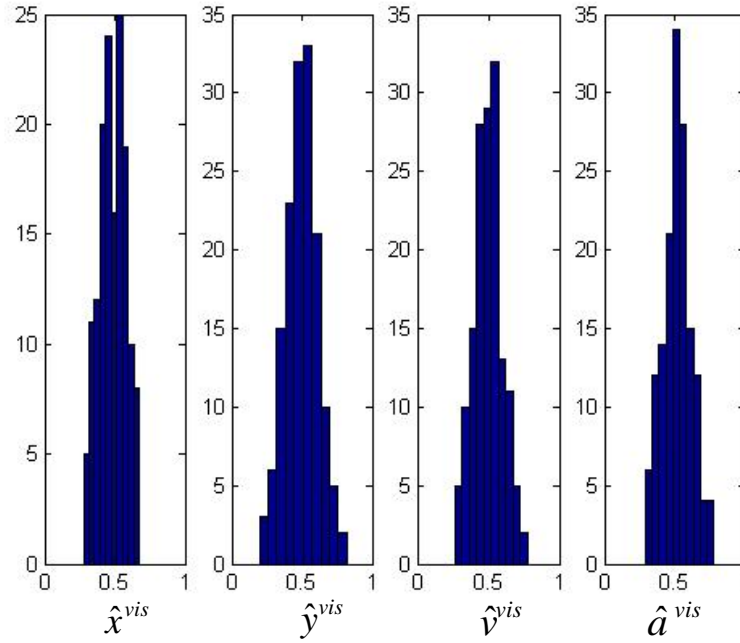
синтезированных траекторий были оценены ошибки оценивания кинематических параметров врачом-экспертом путем визуального наблюдения и автоматически с применением разработанной модели.

Были проверены законы распределения ошибок при оценивании кинематических параметров движения координат  $(x_j, y_j)$  (они рассматривались отдельно, поскольку врач-эксперт в практической деятельности оценивает  $x$  – координату как размах движения, а  $y$  – координату как высоту движения), скорость  $v_j$  и ускорение  $a_j$  ( $j=1, \dots, N$ ,  $N$  – число наблюдаемых ХТ в кадрах видеопотока ХТ). Ошибки оценивались с помощью описанной методики синтезированных траекторий.

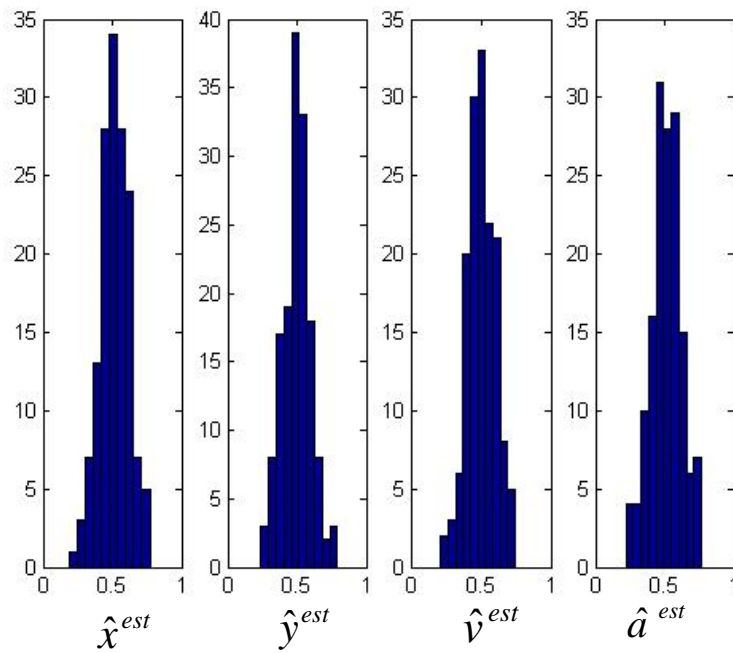
Поскольку кинематические параметры синтезированной траектории известны абсолютно точно, можно оценить ошибки оценки кинематических параметров при оценивании их врачом-экспертом и по предложенной кинематической модели с усреднение оценок, рассматривая и врача-эксперта и автомат, как системы оценивания координат, скоростей и ускорений. На рис.3.7 показаны гистограммы этих ошибок.

Вид гистограмм дает основания предполагать нормальный закон распределения ошибок. Гипотеза о нормальности закона распределения проверена по тесту Шапиро-Уилка ( $W$ -критерию) [104,105]. Как указано в [105],  $W$ -критерий – один из наиболее эффективных критериев проверки нормальности выборки. По общей методике применения критериев согласия, проверка проводилась в два этапа:

- расчет  $W$ -статистики;
- сравнение  $W$ -статистики с пороговым значением критерия Шапиро-Уилка  $W(\alpha)$ , если  $W < W(\alpha)$ , то гипотеза о нормальности распределения отклоняется на уровне значимости  $\alpha$ .



а)



б)

Рисунок 3.7 – Гистограммы распределения ошибок оценивания кинематических параметров путем визуального оценивания врачом-экспертом (а) и автоматического оценивания по кинематической модели (б)

Результаты проверки приведены в таблице 3.1.

Таблица 3.1 – Статистики Шапиро-Уилка распределения ошибок оценивания кинематических параметров

Визуальное оценивание Выборка: 50 синтезированных траекторий		
Параметр	<i>W</i> -статистика	Пороговая статистика $W_{50}(0,05)$
Ошибка оценивания координаты <i>x</i>	0,983	0,947
Ошибка оценивания координаты <i>y</i>	0,992	0,947
Ошибка оценивания скорости	0,971	0,947
Ошибка оценивания ускорения	0,959	0,947
Автоматическое оценивание Выборка: 50 синтезированных траекторий		
Ошибка оценивания координаты <i>x</i>	0,982	0,947
Ошибка оценивания Координаты <i>y</i>	0,977	0,947
Ошибка оценивания скорости	0,969	0,947
Ошибка оценивания ускорения	0,986	0,947

Как видно из таблицы 3.1, для всех выборок ошибок выполняется условие:

$$W > W_{50}(0,05),$$

т.е. гипотеза о нормальности распределения ошибок подтверждается.

Поскольку установлено, что случайные оценки кинематических параметров, получаемые ЛПР при визуальном наблюдении, распределены по нормальному закону:

$$p(x_j^{vis}) = \frac{1}{\sigma_{xj}^{vis} \sqrt{2\pi}} \exp\left(-\frac{x_j^{vis2}}{2\sigma_{xj}^{vis2}}\right), p(y_j^{vis}) = \frac{1}{\sigma_{yj}^{vis} \sqrt{2\pi}} \exp\left(-\frac{y_j^{vis2}}{2\sigma_{yj}^{vis2}}\right), \quad (3.34)$$

$$p(v_j^{vis}) = \frac{1}{\sigma_{vj}^{vis} \sqrt{2\pi}} \exp\left(-\frac{v_j^{vis2}}{2\sigma_{vj}^{vis2}}\right), p(a_j^{vis}) = \frac{1}{\sigma_{aj}^{vis} \sqrt{2\pi}} \exp\left(-\frac{a_j^{vis2}}{2\sigma_{aj}^{vis2}}\right), \quad (3.35)$$

где  $\sigma_{...j}^{vis2}$  - дисперсии оценок соответствующих кинематических параметров.

Согласно [106] энтропия соответствующих оценок кинематических величин для  $j$ -й ХТ равна:

$$H(x_j^{vis}) = \log \left[ \frac{\sqrt{2\pi e \sigma_{xj}^{vis}}}{\Delta x} \right], \quad (3.36)$$

$$H(y_j^{vis}) = \log \left[ \frac{\sqrt{2\pi e \sigma_{yj}^{vis}}}{\Delta y} \right], \quad (3.37)$$

$$H(v_j^{vis}) = \log \left[ \frac{\sqrt{2\pi e \sigma_{vj}^{vis}}}{\Delta v} \right], \quad (3.38)$$

$$H(a_j^{vis}) = \log \left[ \frac{\sqrt{2\pi e \sigma_{aj}^{vis}}}{\Delta a} \right], \quad (3.39)$$

где  $\Delta x$ ,  $\Delta y$ ,  $\Delta v$ ,  $\Delta a$  - разрешающие способности системы по соответствующим кинематическим параметрам. В предположении равенства



дисперсий оценок одноименных параметров для всех ХТ и в силу аддитивности энтропии получаем общую энтропию системы при визуальной оценке кинематических параметров движения пациента врачом в виде:

$$\begin{aligned}
 H_1 &= N \log \left[ \frac{\sqrt{2\pi e \sigma_x^{vis}}}{\Delta x} \right] + N \log \left[ \frac{\sqrt{2\pi e \sigma_y^{vis}}}{\Delta y} \right] + N \log \left[ \frac{\sqrt{2\pi e \sigma_v^{vis}}}{\Delta v} \right] + \\
 &+ N \log \left[ \frac{\sqrt{2\pi e \sigma_a^{vis}}}{\Delta a} \right] = N \log \left[ \frac{\sqrt{2\pi e \sigma_x^{vis}}}{\Delta x} \cdot \frac{\sqrt{2\pi e \sigma_y^{vis}}}{\Delta y} \cdot \frac{\sqrt{2\pi e \sigma_v^{vis}}}{\Delta v} \cdot \frac{\sqrt{2\pi e \sigma_a^{vis}}}{\Delta a} \right] = \\
 &= N \log \left[ \sqrt{2\pi e \sigma_x^{vis}} \sqrt{2\pi e \sigma_y^{vis}} \sqrt{2\pi e \sigma_v^{vis}} \sqrt{2\pi e \sigma_a^{vis}} \right] - N \log [ \Delta x \Delta y \Delta v \Delta a ] \quad (3.40)
 \end{aligned}$$

Обычно в практике оценки энтропии измерительных систем полагают [106], что второе слагаемое существенно меньше первого и им можно пренебречь, поэтому получим, что

$$H_1 = N \log \left[ \sqrt{2\pi e \sigma_x^{vis}} \sqrt{2\pi e \sigma_y^{vis}} \sqrt{2\pi e \sigma_v^{vis}} \sqrt{2\pi e \sigma_a^{vis}} \right]. \quad (3.41)$$

При тех же предположениях о нормальном законе распределения оценок кинематических параметров получим выражение для информационной энтропии системы при оценке параметров движения тела пациента с помощью системы ППР:

$$\begin{aligned}
 H_2 &= N \log \left[ \frac{\sqrt{2\pi e \sigma_x^{est}}}{\Delta x} \cdot \frac{\sqrt{2\pi e \sigma_y^{est}}}{\Delta y} \cdot \frac{\sqrt{2\pi e \sigma_v^{est}}}{\Delta v} \cdot \frac{\sqrt{2\pi e \sigma_a^{est}}}{\Delta a} \right] = \\
 &= N \log \left[ \sqrt{2\pi e \sigma_x^{est}} \sqrt{2\pi e \sigma_y^{est}} \sqrt{2\pi e \sigma_v^{est}} \sqrt{2\pi e \sigma_a^{est}} \right] - N \log [ \Delta x \Delta y \Delta v \Delta a ] \quad (3.42)
 \end{aligned}$$

$$H_2 = N \log \left[ \sqrt{2\pi e \sigma_x^{est}} \sqrt{2\pi e \sigma_y^{est}} \sqrt{2\pi e \sigma_v^{est}} \sqrt{2\pi e \sigma_a^{est}} \right]. \quad (3.43)$$

В определенной степени повышение достоверности принятия решения определяется степенью устранения неопределенности о движении ХТ тела (степенью "снятия энтропии").

Определим относительное уменьшение информационной энтропии как:

$$K_{\text{умен\_энтр}} = \frac{H_1 - H_2}{H_1} = 1 - \frac{H_2}{H_1},$$

$$K_{\text{умен\_энтр}} = 1 - \frac{\log \left[ \sqrt{2\pi e \sigma_y^{\text{vis}}} \sqrt{2\pi e \sigma_y^{\text{est}}} \sqrt{2\pi e \sigma_v^{\text{vis}}} \sqrt{2\pi e \sigma_a^{\text{vis}}} \right]}{\log \left[ \sqrt{2\pi e \sigma_y^{\text{est}}} \sqrt{2\pi e \sigma_y^{\text{est}}} \sqrt{2\pi e \sigma_v^{\text{est}}} \sqrt{2\pi e \sigma_a^{\text{est}}} \right]}. \quad (3.44)$$

Видно, что в полученном выражении информационный выигрыш определяется исключительно уменьшением дисперсии оценок кинематических параметров при использовании разработанной модели автоматического оценивания параметров движения системы ППР. Оцененные СКО  $\sigma_x^{\text{vis}}$ ,  $\sigma_y^{\text{vis}}$ ,  $\sigma_v^{\text{vis}}$ ,  $\sigma_a^{\text{vis}}$  и  $\sigma_x^{\text{est}}$ ,  $\sigma_y^{\text{est}}$ ,  $\sigma_v^{\text{est}}$ ,  $\sigma_a^{\text{est}}$  приведены в таблице 3.2.

Подстановка экспериментальных данных в соотношение (3.44) дает оценку уменьшения энтропии:

$$K_{\text{умен\_энтрапии}} = 0.617.$$

### 3.6 Метод поддержки принятия решений по коррекции движений пациента

Для ППР при ДДР необходимо количественно сравнить модельные матрицы обучающего движения врача и движения, выполненного пациентом.

Таблица 3.2 – СКО оценок кинематических параметров

Э1. Визуальное оценивание		
Выборка: 50 синтезированных траекторий		
Параметр	Диапазон изменения	Усредненное СКО оценки параметра
Координата $x$ , м	[0;2,5]	0,09±0,02
Координата $y$ , м	[0;2,5]	0,09±0,02
Скорость, м/с	[0;5]	0,19±0,04
Ускорение, м/с <sup>2</sup>	[0;3]	0,26±0,04
Э2. Автоматическое оценивание		
Выборка: 150 синтезированных траекторий		
Координата $x$ , м	[0;2,5]	0,04±0,01
Координата $y$ , м	[0;2,5]	0,04±0,01
Скорость, м/с	[0;5]	0,07±0,02
Ускорение, м/с <sup>2</sup>	[0;3]	0,11±0,03

Выбор метрики различия векторов модельной матрицы (3.29) является достаточно сложной задачей. Хотя задачи выбора метрик достаточно детально исследованы в задачах распознавания образов, в частности в кластерном анализе [107], в задаче ППР покоррекции движений к метрике предъявляются особые требования:

1. Метрика должна быть достаточно чувствительной к различию векторов;
2. С точки зрения задачи реального времени метрика должна обладать минимальной вычислительной сложностью;
3. Желательно, чтобы метрика была нормированной, т.е. приведенной к интервалу [0,1];

4. Поскольку алгоритмы выделения ХТ и оценки кинематических параметров обладают вычислительными погрешностями, данные содержащиеся в векторах кинематических параметров являются зашумленными, поэтому выбранная метрика должна быть наименее чувствительной к шумам и не требовать дополнительного сглаживания векторов.

Для количественной оценки отличия двух матриц модели исследовались следующие метрики различия векторов  $\mathbf{a}_{j\text{эт}}^n$  и  $\mathbf{b}_{j\text{польз}}^n$  (в качестве  $\mathbf{a}_{j\text{эт}}^n$  и  $\mathbf{b}_{j\text{польз}}^n$  могут быть использованы вектор-столбцы модели (3.29) одноименной физической размерности  $x_j^n, y_j^n, v_{\dot{q}}^n, a_{\ddot{q}}^n$ ) [108-110]:

метрика Хаусдорфа

$$l_{Haus}^{ab} = \max(\max_i \min_j |\mathbf{a}_{i\text{эт}}^n - \mathbf{b}_{j\text{польз}}^n|, \max_j \min_i |\mathbf{a}_{i\text{эт}}^n - \mathbf{b}_{j\text{польз}}^n|), \quad (3.45)$$

метрика Камберра

$$l_{Camb}^{ab} = \sum_{j=1}^K \frac{|\mathbf{a}_{j\text{эт}}^n - \mathbf{b}_{j\text{польз}}^n|}{|\mathbf{a}_{j\text{эт}}^n + \mathbf{b}_{j\text{польз}}^n|}, \quad (3.46)$$

метрика Минковского

$$l_{Mink}^{ab} = \left( \sum_{j=1}^K |\mathbf{a}_{j\text{эт}}^n - \mathbf{b}_{j\text{польз}}^n|^p \right)^{1/p} \text{ для } p = 3, 4, 5, \quad (3.47)$$

чебышевская мера (представляющая собой предельную меру Минковского для  $p = \infty$ )

$$l_{\infty}^{ab} = \max_{\substack{j=1, \dots, K \\ \text{для всех } n}} |\mathbf{a}_{j\text{эт}}^n - \mathbf{b}_{j\text{польз}}^n|, \quad (3.48)$$

и косинусная мера

$$l_{\cos}^{ab} = \frac{\mathbf{a}_{j\text{эт}}^n \times \mathbf{b}_{j\text{польз}}^n}{|\mathbf{a}_{j\text{эт}}^n| \times |\mathbf{b}_{j\text{польз}}^n|}. \quad (3.49)$$

$j=1, \dots, K$   
для всех  $n \in N_\infty$

В результате исследования всех перечисленных метрик путем программной реализации и применения к экспериментально зарегистрированным кадрогам установлены их характеристики, приведенные в таблице 3.3.

Таблица 3.3 – Результаты исследования мер близости векторов

Наименование метрики	Среднее время счета, мс	Минимально разрешаемые кинематические параметры (в пиксельных координатных единицах)					
		По координатам (x, y), пикс		По скорости $v_r$ , пикс/с		По ускорению $a_r$ , пикс/с <sup>2</sup>	
		Без шума	AGN ОСШ= 5дБ	Без шума	AGN ОСШ= 5 дБ	Без шума	AGN ОСШ= 5 дБ
Метрика Хаусдорфа $l_{Haus}^{ab}$	155-220 мс	2-3	4-5	3-4	8-10	2-3	5-6
Метрика Камбера $l_{Camb}^{ab}$	23-27 мс	2-4	8-10	8-9	14-18	5-8	10-12
Метрика Миньковського $l_{Mink}^{ab}$	42-52 мс	1-2	2-3	3-4	4-5	2-3	2-4
Чебишовська метрика $l_{\infty}^{ab}$	2-3 мс	1-2	1-2	8-9	8-9	6-9	8-9
Косинусна міра $l_{\cos}^{ab}$	16-18 мс	2-3	2-4	5-7	5-8	2-3	2-4

Перечисленные метрики исследованы экспериментально путем программной реализации и применения к экспериментально зарегистрированным видеопотокам. Метрики близости исследованы без

шума и с аддитивным гауссовым шумом, наложенным на сравниваемые векторы кинематической матрицы с результирующим отношением сигнал/шум 5 дБ (столбцы таблицы 3.3 "AGN ЗСШ = 5 дБ").

В результате исследования по перечисленным выше критериям в качестве метода предлагается следующая двухэтапная процедура. На первом этапе для координатных вектор-столбцов обеих матриц  $\mathbf{x}_j^n, \mathbf{y}_j^n$  находится чебышевская мера различия векторов:

$$l_\infty^x = \max_{\substack{j=1, \dots, K \\ \text{для всех } n}} |\mathbf{x}_{j \text{ эт}}^n - \mathbf{x}_{j \text{ польз}}^n|, \quad (3.50)$$

$$l_\infty^y = \max_{\substack{j=1, \dots, K \\ \text{для всех } n}} |\mathbf{y}_{j \text{ эт}}^n - \mathbf{y}_{j \text{ польз}}^n|. \quad (3.51)$$

Если рассчитанные меры близости не превышают допустимого порога отклонения:

$$l_\infty^x \leq Thesh(l_\infty^x), \quad (3.52)$$

$$l_\infty^y \leq Thesh(l_\infty^y), \quad (3.53)$$

где  $Thesh(l_\infty^x)$  и  $Thesh(l_\infty^y)$  – экспериментально установленные пороги, то движение считается повторенным правильно. При этом чебышевская мера нормируется к максимальному радиусу  $R_j^{\max}$  для соответствующей ХТ  $j$ . Далее применяется нормированная чебышевская мера близости (НЧМБ). Чебышевская мера практически не требует вычислительных затрат по сравнению с остальными и вынесение решения на первом этапе происходит быстро. В случае невыполнения условий (3.52-3.53) по чебышевскому отклонению быстро уставливаются номера ХТ тела, координаты которых отличаются от требуемых. На основании первого этапа оценки пользователю выдаются рекомендации первого уровня по координатам движений (например, "правая рука - выше", "наклон головы вправо - ниже").

На втором этапе для установленного множества ХТ с отклонениями по координатам  $N_\infty$  рассчитывается косинусная мера близости по векторам скорости и ускорения. Напомним, что косинусная мера близости векторов есть отношение скалярного произведения векторов к произведению евклидовых норм этих векторов. Для совпадающих векторов угол между ними равен нулю и соответственно косинусная мера близости равна 1. Косинусные меры близости оцениваются по формулам:

$$l_{\cos}^v = \frac{\mathbf{v}_{\text{т}}^n \times \mathbf{v}_{\text{польз}}^n}{|\mathbf{v}_{\text{т}}^n| \times |\mathbf{v}_{\text{польз}}^n|}, \quad (3.54)$$

$j=1, \dots, K$   
для всех  $n \in N_\infty$

$$l_{\cos}^a = \frac{\mathbf{a}_{\text{т}}^n \times \mathbf{a}_{\text{польз}}^n}{|\mathbf{a}_{\text{т}}^n| \times |\mathbf{a}_{\text{польз}}^n|}. \quad (3.55)$$

$j=1, \dots, K$   
для всех  $n \in N_\infty$

На основании сравнения векторов скорости и ускорения "отклоняющихся точек" формируется второй уровень рекомендаций пользователю:

Если

$$1 - l_{\cos}^v \leq Thesh(l_{\cos}^v), \quad (3.56)$$

$$1 - l_{\cos}^a \leq Thesh(l_{\cos}^a), \quad (3.57)$$

где  $Thesh(l_{\cos}^v)$  и  $Thesh(l_{\cos}^a)$  – экспериментально установленные пороги скорости ускорения соответственно, то движение считается повторенным правильно. В противном случае выдаются рекомендации по коррекции движений пользователя по скорости (например, "правая рука - быстрее", "наклон головы вправо - медленнее") и по ускорению - (например, "правая рука - резче", "наклон головы вправо - менее резко").

Предварительные оценки, полученные в результате тестирования прототипа системы, показывают, что введенные выше пороги отклонения по

координатам составляют 8-12% от максимального радиуса вращения, а по косинусной мере лежат в диапазоне абсолютных величин 0,09-0,11.

3.7. Базовая информационная технология формирования результата сравнения движений и вынесения корректирующего решения

На основании разработанного метода количественной оценки меры близости двух движений построена базовая информационная технология № 3 (БИТ№3) – технология сравнения видеопотоков двух движений и формирования результата сравнения в виде массива количественных оценок и рекомендаций по коррекции движения пациента.

**Исходные данные:** Матрицы оценок кинематических параметров обучающего и выполненного движений для  $N$  последовательных кадров видеопотока.

**Последовательно выполняемые операции:**

1. Расчет чебышевской меры различия векторов кинематической матрицы соответствующей физической размерности и сравнение меры с экспериментально установленным порогом.

2. Вынесение решений 1-го уровня по координатам характерных точек выполненного движения, отличающихся от обучающего движения.

3. Расчет косинусной меры различия векторов соответствующей физической размерности для точек, определенных в п.2, сравнение косинусной меры с экспериментально установленным порогом.

4. Вынесение решений 2-го уровня по скорости и ускорению характерных точек выполненного движения, отличающихся от обучающего движения.

**Результат:** Структура, содержащая перечень характерных точек, движение которых отличается от обучающего, массив кинематических параметров этих точек, массив значений отклонений по чебышевской и косинусной мерам, рекомендации по коррекции движения каждой точки.



## Выводы по третьему разделу

1. Разработана кинематическая модель тела человека в видеопотоке, которая в отличие от существующих построена на аппроксимации траекторий характерных точек в 3-х кадровых фрагментах видеопотока дугами окружностей, что позволяет получить точечные оценки мгновенных тангенциальных скоростей и ускорений характерных точек тела.

2. Получил дальнейшее развитие метод построения статистических оценок кинематических параметров, который основан на расчете их скользящего среднего по последовательным фрагментам видеопотока, что позволяет получить несмещенные, состоятельные и эффективные оценки координат, тангенциальных скоростей и тангенциальных ускорений ХТ тела.

3. Проведена проверка адекватности кинематической модели с применением методики синтезирования траекторной видеопоследовательности. Установлено, что при расчете скользящего среднего оценок кинематических параметров не менее, чем по 10 фрагментам (или 12 последовательным кадрам) видеопотока оценки параметров практически не смещены, коэффициенты вариации оценок координат ХТ не превышают 3,2%, скорости – 4,1 %, ускорения – 4,8%.

4. На основании предложенной кинематической модели и метода оценки кинематических параметров разработана базовая информационная технология № 2 (БИТ№2) – технология преобразования кадров видеопотока с изображением движения тела в кинематическую матрицу параметров движения.

5. Экспериментальным путем установлено, что энтропия системы как системы оценивания кинематических параметров с применением разработанной модели и метода усреднения уменьшается на 61,7 %, что косвенно указывает на степень повышения достоверности решения.

6. Разработан метод поддержки принятия решения по оценке правильности и коррекции движения пациента, которые отличаются возможностью количественного сравнения параметров обучающего и текущего движений, что позволяет повысить достоверность принятия правильного решения.

7. На основании разработанного метода поддержки принятия решения построена базовая информационная технология №3 (БИТ №3) – технология сравнения видеопотоков двух движений и формирования результата сравнения в виде массива количественных оценок и рекомендаций по коррекций движения пациента.

## РАЗДЕЛ 4

### ИНФОРМАЦИОННАЯ ТЕХНОЛОГИЯ ПОСТРОЕНИЯ СИСТЕМЫ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ ДЛЯ ДИСТАНЦИОННОЙ ДВИГАТЕЛЬНОЙ РЕАБИЛИТАЦИИ И ОЦЕНКА ХАРАКТЕРИСТИК РАЗРАБОТАННОЙ СИСТЕМЫ

Разработанные в разделах 2 и 3 модели и методы и базовые информационные технологии БИТ №1, БИТ №2, БИТ №3 легли в основу разработки интегрированной информационной технологии поддержки принятия решений при ДДР. Разработанная интегрированная информационная технология ППР должна функционировать в трех режимах: подготовительном, автоматическом (без участия врача-реабилитолога) и автоматизированном (с участием врача).

На базе разработанной информационной технологии необходимо разработать программный проект СППР для ДДР.

Разработанная СППР должна быть протестирована, необходимо оценить ее показатели по достоверности принятия решений.

#### 4.1 Информационная технология поддержки принятия решений

Разработанная информационная технология имеет три режима функционирования:

1. Подготовительный режим, в котором участвует только врач-реабилитолог, в этом режиме врач регистрируется в системе, выполняет обучающее упражнение и записывает инструкцию для пациента решения по коррективке движения (рис.4.1).

Выполненное врачом обучающее упражнение (ОУ) в виде видеопотока передается на сервер, где над видеопотоком ОУ выполняются следующие операции: покадрово формируется двумерная проективная точечная модель движения, строится кинематическая модель движения, по методу построения

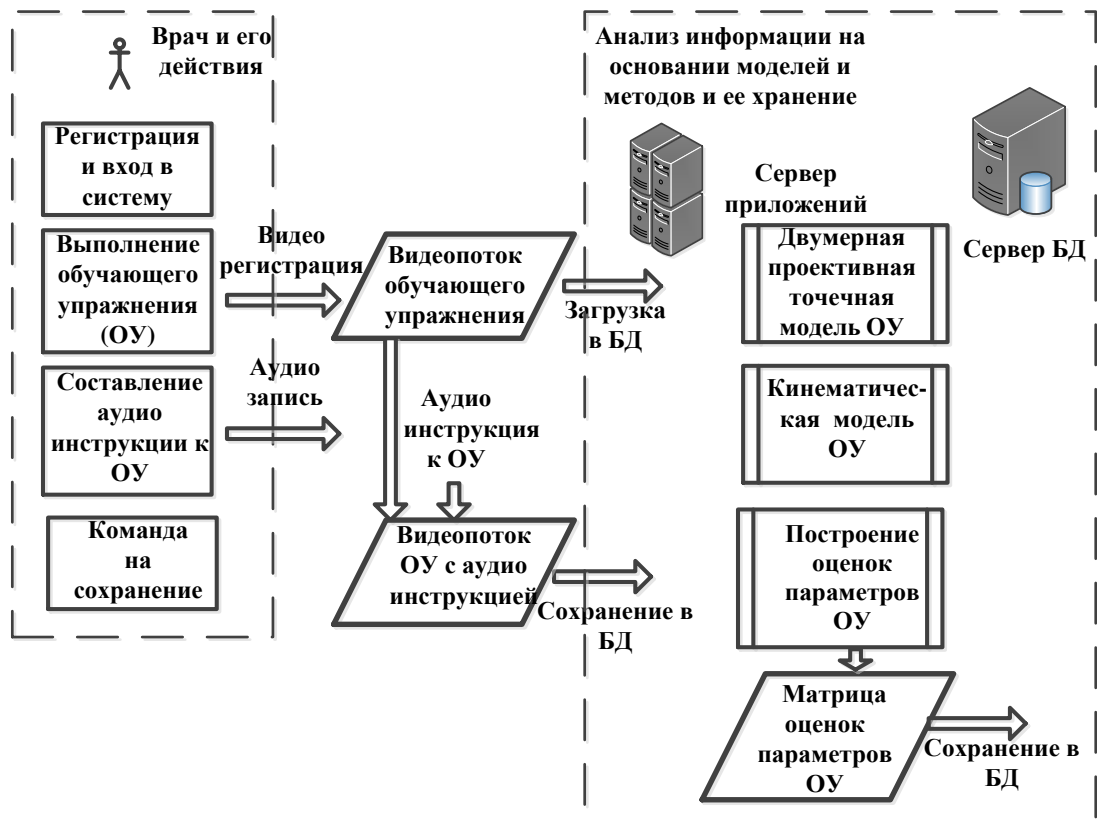


Рисунок 4.1 – Подготовительный режим ИТ

статистических оценок кинематических параметров строится матрица обучающего движения  $M_{обуч}$ . Матрица для данного упражнения сохраняется в БД СППР.

2. Автоматический режим ДДР, в котором участвует только пациент, решения по коррекции движений принимаются автоматом (рис.4.2). Пациент регистрируется в системе, выбирает требуемое упражнение, изучает видео инструкцию и выполняет упражнение. Видеозапись выполненного движения (ВД) пациента в виде видеопотока передается на сервер, где над видеопотоком ВД выполняются операции, перечисленные в режиме 1. В результате строится матрица выполненного движения  $M_{польз}$ . Затем по методу формирования решений по коррекции движений сравниваются кинематические матрицы обучающего  $M_{обуч}$  и выполненного движений  $M_{польз}$ , выносятся решение по коррекции движения и сообщается пациенту.

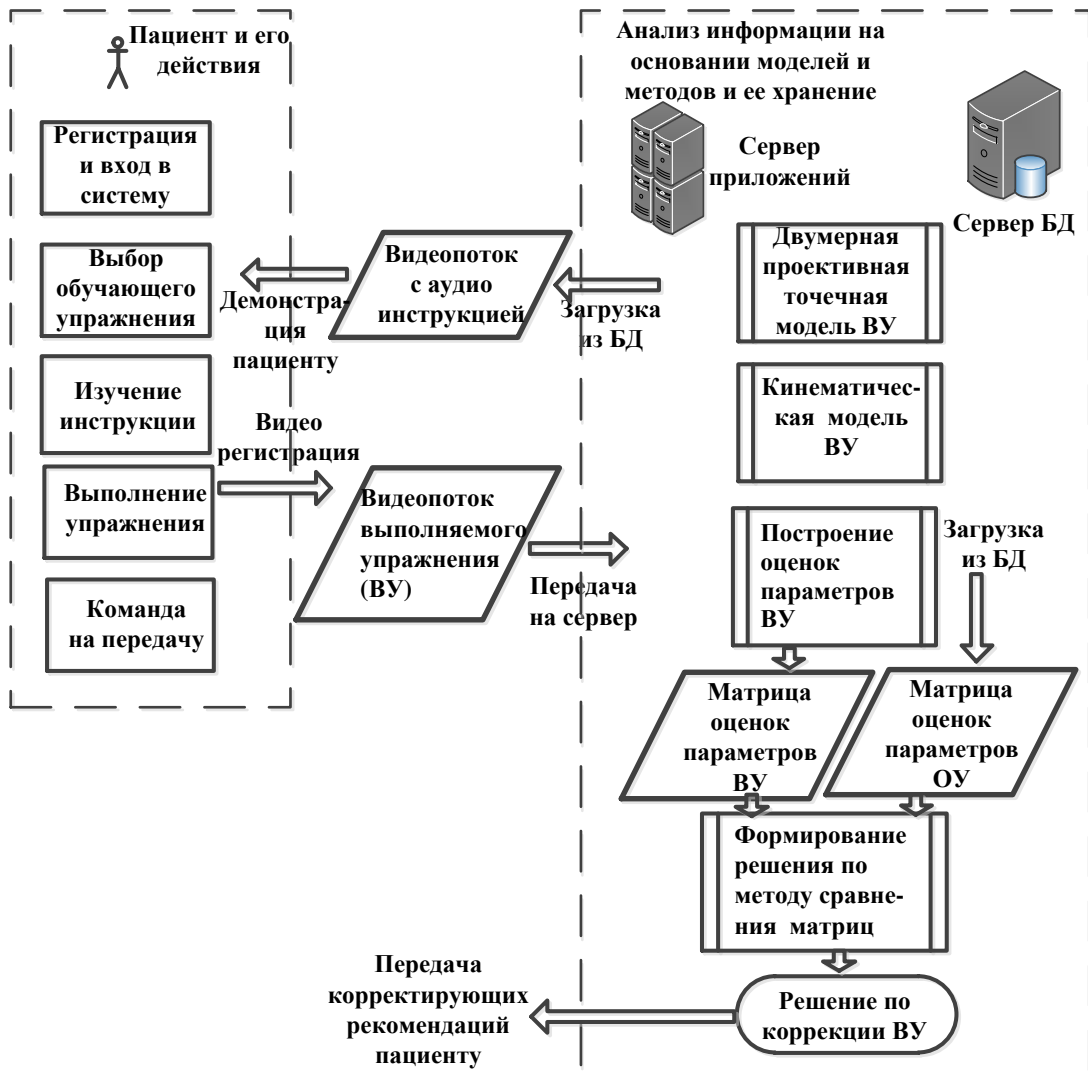


Рисунок 4.2 – Автоматический режим ИТ

На рис.4.3 показан алгоритм функционирования ИТ в автоматическом режиме.

При выполнении комплекса упражнений независимо от правильности их выполнения задано максимальное число упражнений *Limit*, определяемое врачом для данного пациента с учетом индивидуальных особенностей его организма, возраста и состояния здоровья.

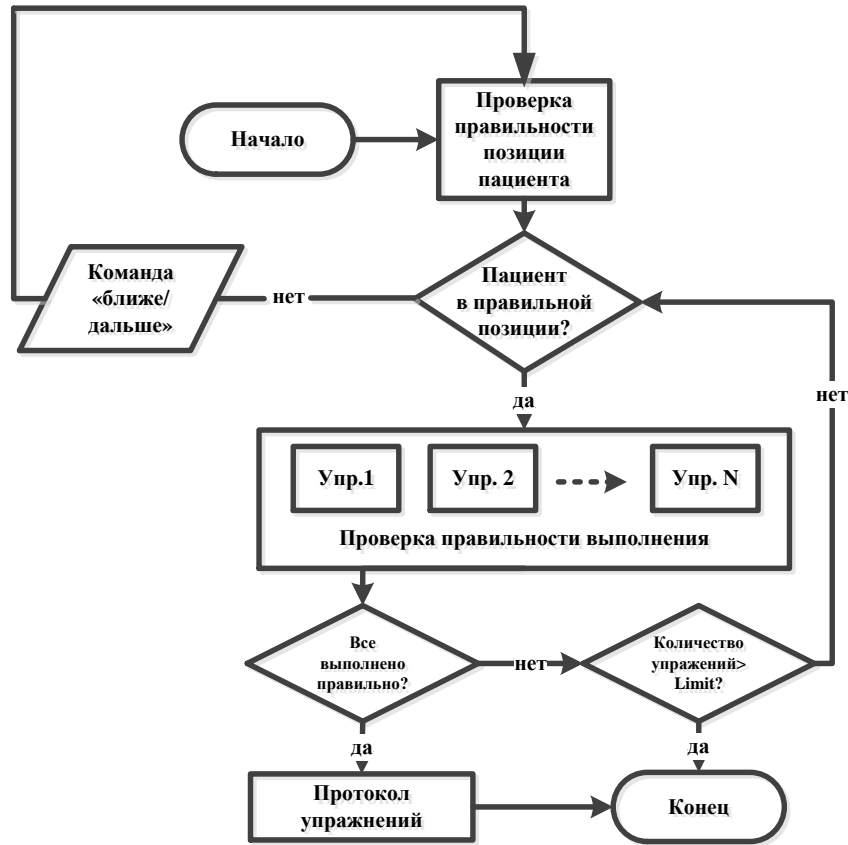


Рисунок 4.3 – Алгоритм функционирования ИТ в автоматическом режиме

3. Автоматизированный режим. В автоматизированном режиме ЛПР является врач, который анализирует видеозапись выполненного движения пациента, результаты сравнения кинематических матриц обучающего  $M_{обуч}$  и выполненного движений  $M_{польз}$ , корректирует решение и направляет его пациенту. Схема автоматизированного режима ИТ приведена на рис. 4.4.

4.2 Система поддержки принятия решений, разработанная на основании ИТ и ее архитектура

На базе разработанных ИТ ППР разработана распределенная СППР для ДДР (рис.4.5), которая состоит из интерфейсной подсистемы для пользования системой пациентами и врачами-реабилитологами, и подсистемы анализа данных, принятия решения и хранения информации

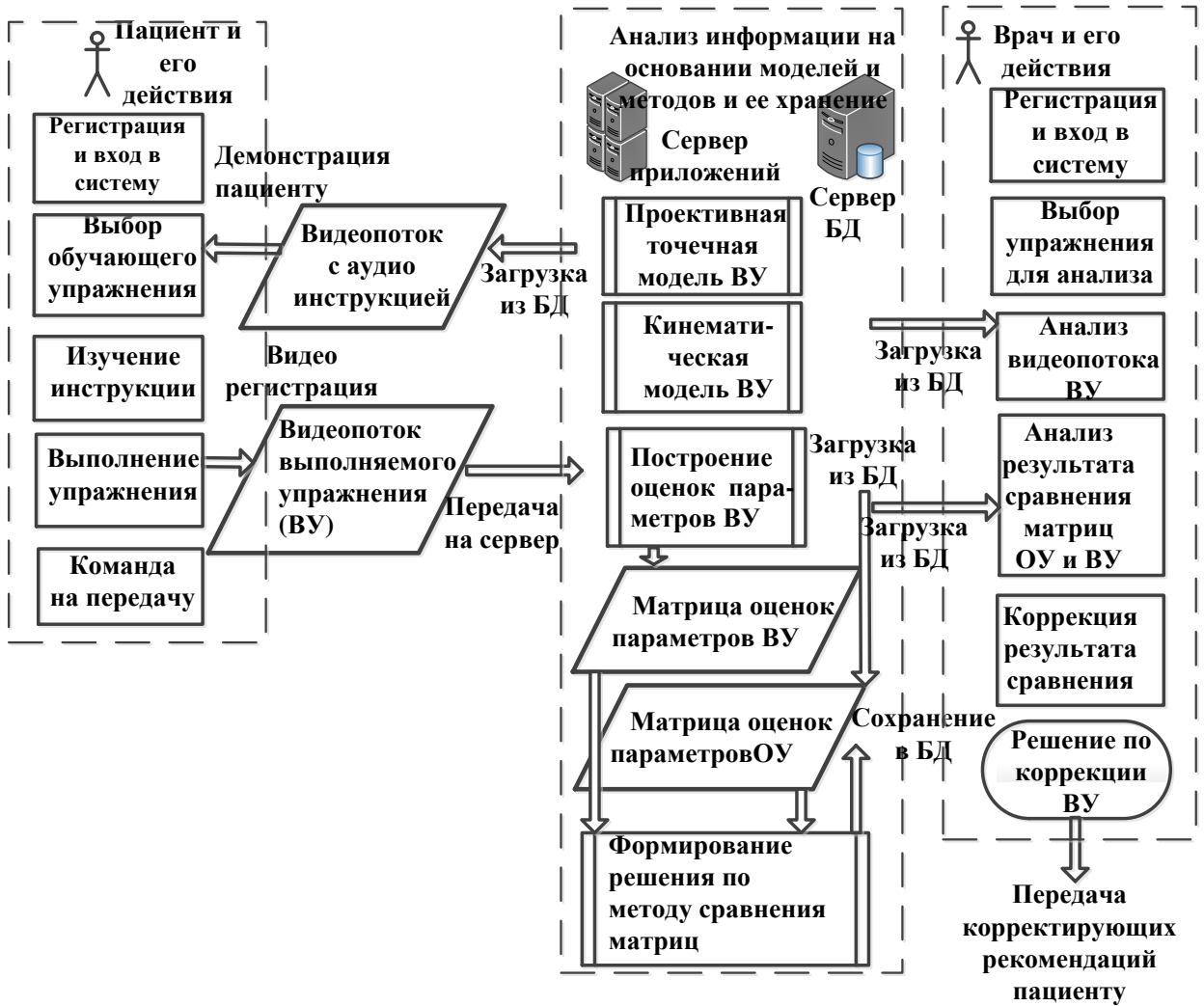


Рисунок 4.4 – Схема информационной технологии автоматизированной поддержки принятия решений

(АДПРХИ). СППР реализована в виде веб-приложения в клиент-серверной архитектуре. Подсистема АДПРХИ реализована в виде четырех компонентов: веб-сервера, организующего работу приложений (принимающего HTTP-запросы от клиентов из веб-браузеров), и выдающий им HTTP-ответы, организующего работу приложений, сервера приложений, на котором выполняются а все вычислительно емкие процедуры по обработке видеопотоков, построению моделей и анализу результатов, сервера базы данных (БД), где размещается общая БД СППР, в ней сохраняются все данные, обработывавшиеся в процессе принятия решений и результаты функционирования системы.

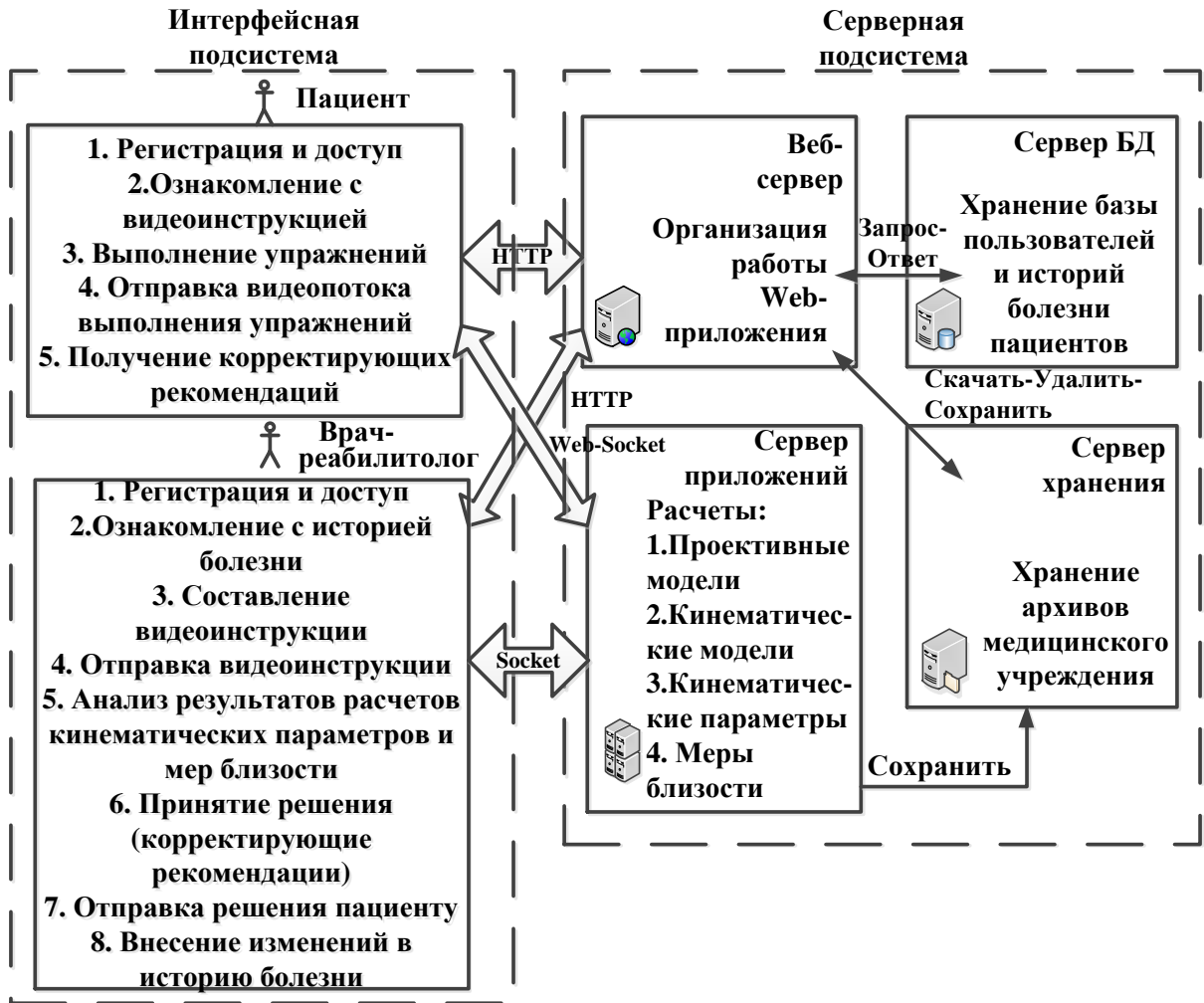


Рисунок 4.5 – Структура разработанной СППР

Веб-сервер и сервер БД выполнены как единый веб-сервер управления WAMP. Для управления пользователями (пациентом, врачом) и управления упражнениями использована технология OpenSource CMS Wordpress, выполненная на PHP. Сервер приложений обрабатывает выполнен на C++ с использованием библиотек Boost и OpenCV. Обмен видеопотоками осуществляется с использованием протокола WebSocket целью облегчения нагрузки на веб-сервер. Кроме того, для ускорения обмена видеопотоками видеофайлы перекодируются в формат Base64 и сжимаются, так что передача их осуществляется в формате сжатых строк.

Выполненная в такой архитектуре СППР при ДДР реализована в виде опытного образца и испытана на реальных двигательных видеоинструкциях



врача и упражнениях, выполняемых пациентами. В качестве клиентской платформы инструктора и пациента были использованы компьютеры с характеристиками CPU: Intel® Core™2 Duo CPU E8400 @3.00 ГГц; RAM: 12 ГБ DDR3 1300 МГц; ОС: Windows 10 Pro x64 и бытовой камерой Logitech HD Webcam C310 с используемым разрешением 640×480 пикселей. Серверная часть с описанной выше архитектурой размещена в облачном датацентре. Описанная реализация СППР для ДДР обеспечивает функционирование системы в реальном времени, наблюдаемые задержки в обмене данными и в поступлении корректирующих рекомендаций не превышают 3-5 с.

Экранная форма СППР показана на рис.4.6.



**Exercise #1: Right Hand**

SHOW CAMERA START PROCESS STOP PROCESS

**Decision.**

№	Name	Decision
9	Right Wrist	Lower and Slower
6	Right Elbow	Upper and Faster
3	Right Shoulder	Correct!

**Instruction:** This is the simplest exercise to practise with your hands. It will help to rehabilitate your Wrist, Elbow and Shoulder. You can start with one hand and continue with another hand periodically. It takes not much time to repeat exercise about 50-100 times. Step to implement:

1. Stand straight

**Clinical history #####**

Age	30	Gender	Man	Doctor	#####
Ex. Group	Practise with Hand	Date	01/08/2015	Begin Time	16h30
End Time	16h05	Duration	2min 10s	Motions Number	100
Temp	1.3s	Observating Points	3,6,9		
Cosine Measures	0.980;0.932;0.899	Chebyshev Measu.	0.950;0.935;0.897		

Рисунок 4.6 – Экранная форма СППР

### 4.3 Программная разработка СППР

#### 4.3.1 Технология и программные средства

При разработке СППР использованы следующие технологии программирования и программные средства:

##### 1. Языки программирования:

- C++ – для реализации основных модулей и приложений обработки видеопотоков и изображений;
- PHP – для реализации веб-приложения;
- Javascript – для передачи данных через web-socket;
- HTML/CSS – для отображения веб-контента в интерфейсах.

## 2. Библиотеки:

– OpenCV 2.4.9 [111,112] с открытыми реализациями базовых алгоритмов обработки с видеопотоков и изображений . Intel® Threading Building Blocks обеспечивает конструкции для распараллеливания программ для систем с общей памятью;

– Boost 1.57.0 — собрание библиотек классов, использующих функциональность языка C++ и предоставляющих удобный, кроссплатформенный, высокоуровневый интерфейс для кодирования работы с данными, алгоритмами, файлами, потоками и т. п.);

– Boost Asio – Суб-библиотека библиотеки Boost для создания сервера приложений;

– Boost JSON-Spirit – суб-библиотека Boost, которая поможет легко читать и писать от/в файлы или потоки по формату JSON.

3. Операционная система: Microsoft® Windows 10 Pro x64.

4. Открытый CMS ресурс: Wordpress 4.2.5.

5. Среды разработки: Microsoft® Visual Studio 2012, Notepad++ v6.8.3, SciLab 5.5.2.

6. Инструмент построения диаграмм: BPWin4.1, MS Visio 2010.

### 4.3.2 Описание основных программных классов, модулей и библиотек

В состав основного программного обеспечения входит несколько программных модулей и приложений, которые разделены на две группы в соответствии с общей архитектурой:

Классы и модули для обработки изображений с классами:

– `Common` – модуль, который содержит часто используемые функции для обработки изображений и видеопотоков;

– `Rehabilitation` – главный модуль обработки, который содержит основные функции, соответствующие этапам процесса выполнения обработки видеопотока, сравнения кинематических характеристик выдачи рекомендации:

– `getBackground()`: Оценка фона.

– `subtractVideoImage()`: Вычитание фона

– `defineObjectArea()`: Бинарное изображение человека

– `defineObjectSkelet()`: Скелетизация

– `defineSpecialPoints()`: Определения особенных точек скелета

– `getResult()`: Сравнение кинематических параметров движения пациента и врача и формирование решения.

Классы и модули для передачи и обработки данных между клиентом и сервером с классами:

– `Server` – класс, использующий для создания сервера, который отслеживает приходящий от клиента сигнал с определенного адреса (`socket`);

– `Session` – класс, представляющий каждое подключение клиента;

– `Dataframe` – структура для сохранения данных `Websocket`;

– `Dataframe_parser` – грамматический разбор входящего `Dataframe`;

– `Request` – класс, представляющий запрос от клиента;

– `Request_handler` – обработчик входящих `http` запросов;

– `Request_parser` – грамматический разбор входящих запросов;

– `Reply` – структура ответа, отправляемого клиенту;

– Room – класс, использующийся в случае, когда требуется обмен данными между клиентами;

– Participant – класс, представляющий участника в классе Room.

На рис. 4.7 представлена диаграмма классов системы.

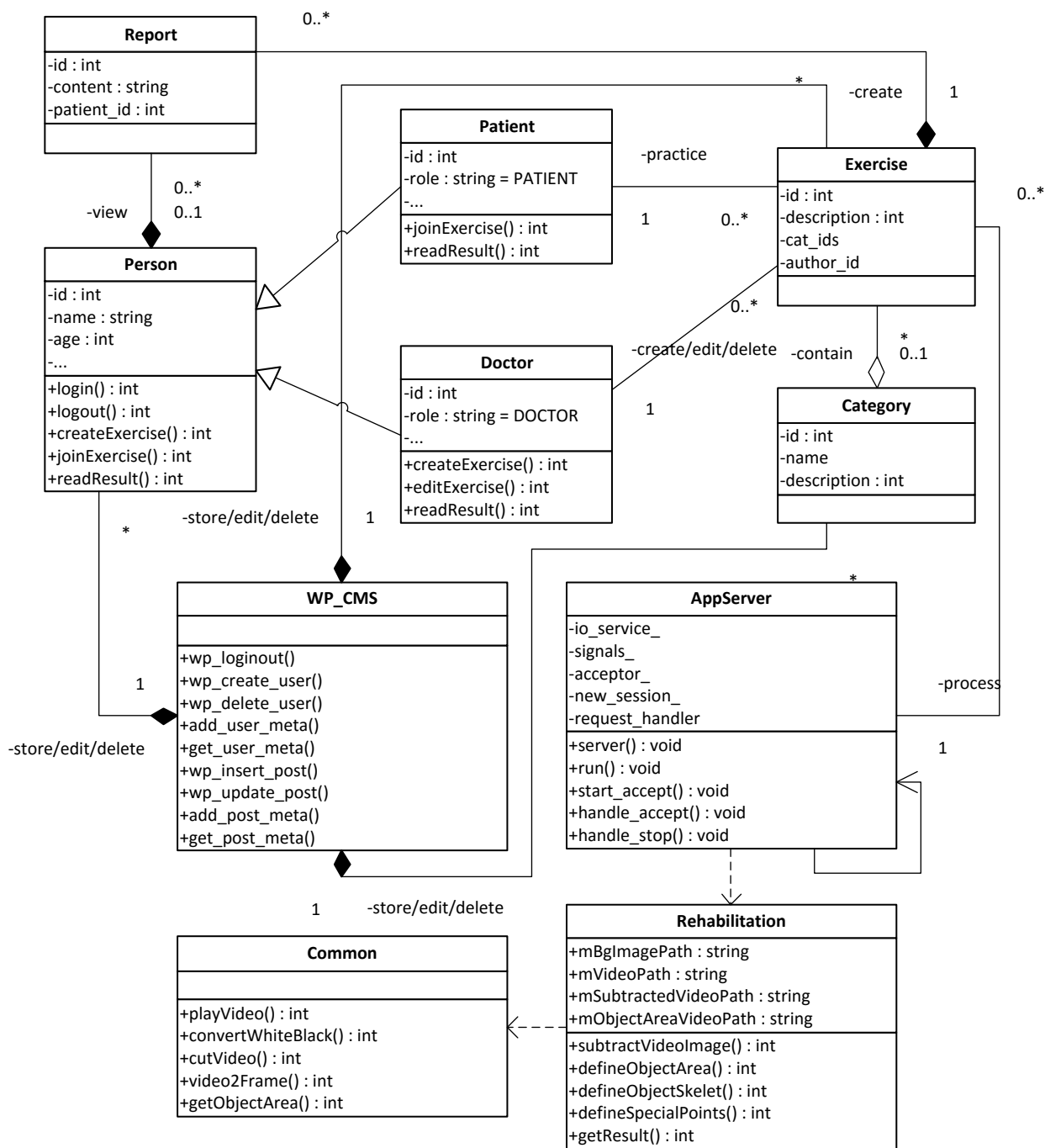


Рисунок 4.7 – Диаграмма классов системы

### 4.3.3 База данных системы

Рассмотрим организацию БД системы. СУБД выбиралась из требований: поддержка не только языка веб-программирования, но и языка C/C++, выполнение базовых запросов (INSERT, SELECT, UPDATE, DELETE) с высокой скоростью, легкое расширение, возможность сохранения и скачивания как локальных, так и удаленных данных. С этой точки зрения была выбрана СУБД MySQL Community Server.

Структура БД СППР при ДДР представлена на рис.4.8.

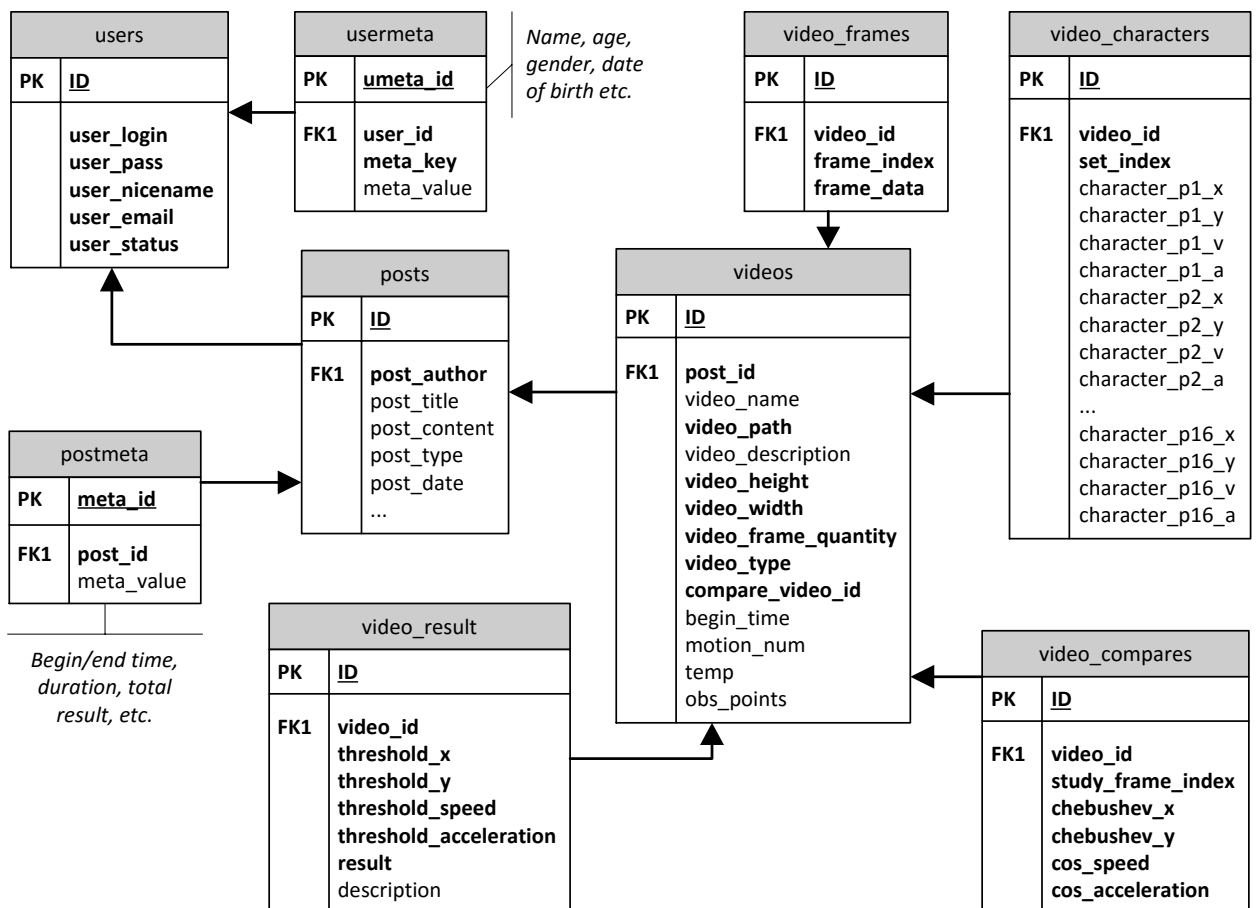


Рисунок 4.8 – Структура БД системы дистанционной двигательной реабилитации

### Основные таблицы БД:

- таблица **user** содержит основную информацию о пользователе (имя, адрес электронной почты, пароль для входа в систему, фамилию, имя и отчество, и статус пользователя (активный / пассивный));

- таблица **usermeta** содержит дополнительную информацию о пользователе: роль (врач/пациент), пол (ж/м), возраст, высота, вес, корректирующие масштабные и антропометрические коэффициенты и т.д., использованная EAV model позволяет легко расширять эту и другие таблицы БД при необходимости;

- таблица **posts** – тема, создающая пользователем: каждый пользователь может создать тему для себя и других пользователей (блог), содержание которого может быть комментариями, замечаниями или просто описанием своего выполнения комплекса упражнений;

- таблица **postmeta** – также как **usermeta**, содержит дополнительную информацию по теме **posts**;

- таблица **videos** – содержит информацию о каждой видеозаписи;

- таблица **video\_frames** – содержит данные каждого из кадров видеопотока;

- таблица **video\_characters** – содержит кинематические параметры после обработки некоторой комбинации кадров видеопотока;

- таблица **video\_results** – содержит окончательные результаты обработки видеопотока;

- таблица **video\_compares** содержит результаты сравнения кинематических параметров (координаты, скорость, ускорение) движения пациента и обучающего движения врача-инструктора по различным мерам подобию.

## 4.4 Тестирование, экспериментальная проверка системы и оценка характеристик системы

#### 4.4.1 Калибровка системы

Разработаны и испытаны две процедуры калибровки системы:

- масштабная калибровка;
- антропологическая пропорциональная калибровка.

**Масштабная калибровка.** Врач и пациент (или его ассистент), поочередно, располагаясь на расстоянии 2,5 м от камеры, принимают Т-позу с мерной рулеткой в руках (рис. 4.9). По крайним точкам размаха рук определяются масштабы  $m = \frac{L_{\text{пикс}}}{L_{\text{м}}}$ , показывающие истинное число пикселей, приходящихся на одну метрическую единицу.



Рисунок 4.9 – Масштабная калибровка системы

Если масштабы изображений врача  $m_{\text{вр}}$  и пациента  $m_{\text{пац}}$  численно отличаются более, чем на 5%, то координаты точек изображения с меньшим масштабом умножаются на корректирующий масштабный множитель

$$Corr_{\text{масш}} = \frac{\max(m_{\text{вр}}, m_{\text{пац}})}{\min(m_{\text{вр}}, m_{\text{пац}})}.$$

Множитель  $Corr_{маси}$  вносится отдельной записью в базу данных пациента.

### **Антропометрическая пропорциональная калибровка.**

Регистрируется рост врача-инструктора и пациента  $H_1$  и  $H_2$  соответственно.

Для каждого фрагмента тела оцениваются коэффициенты  $k_i^1 = \frac{l_i^1}{H_1}, k_i^2 = \frac{l_i^2}{H_2}$ ,

$i = 1, \dots, M$  ( $M$  – число наблюдаемых точек тела), эти коэффициенты

сравниваются со средними антропометрическими коэффициентами  $k_i^0 = \frac{l_i^0}{H_0}$ ,

$i = 1, \dots, M$  [113-115] (рис. 4.10).

Если для какого-то фрагмента тела  $i$  врача или инструктора отличаются от средних антропометрических более, чем на 5%, то координаты характерных точек, ограничивающих этот фрагмент тела, далее всегда умножаются на коэффициент коррекции:

$$Corr_i^1 = \frac{l_i^1}{H_1} / \frac{l_i^0}{H_0}$$

$$Corr_i^2 = \frac{l_i^2}{H_2} / \frac{l_i^0}{H_0}$$

этот факт регистрируется в личной записи пациента (или врача) в БД СППР .

#### 4.4.2 Экспериментальная оценка рабочей зоны системы

Экспериментально оценена рабочая зона системы, в пределах которой сохраняется допустимый процент ошибочных решений (рис. 4.11).

Для оценки рабочей зоны точка расположения пациента отклонялась от оптической оси системы и вдоль оптической оси на различные расстояния. При этом в автоматическом режиме оценивался процент безошибочных решений. Результаты приведены в таблице 4.1.



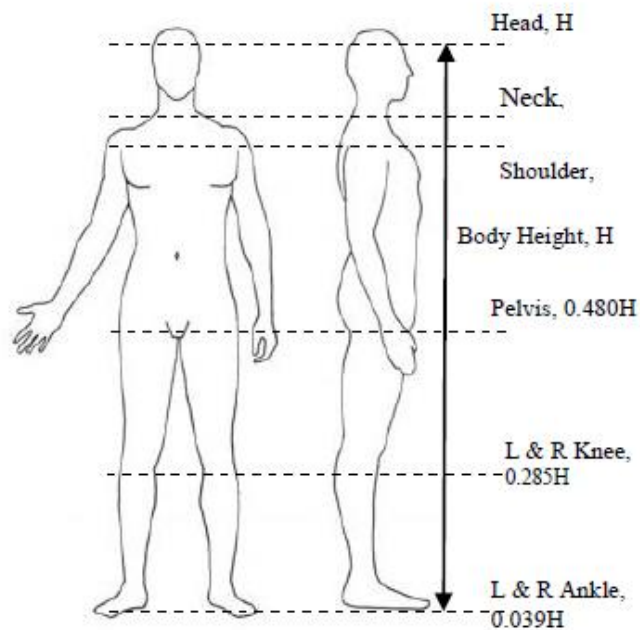


Рисунок 4.10 – Классические антропометрические пропорции [115]

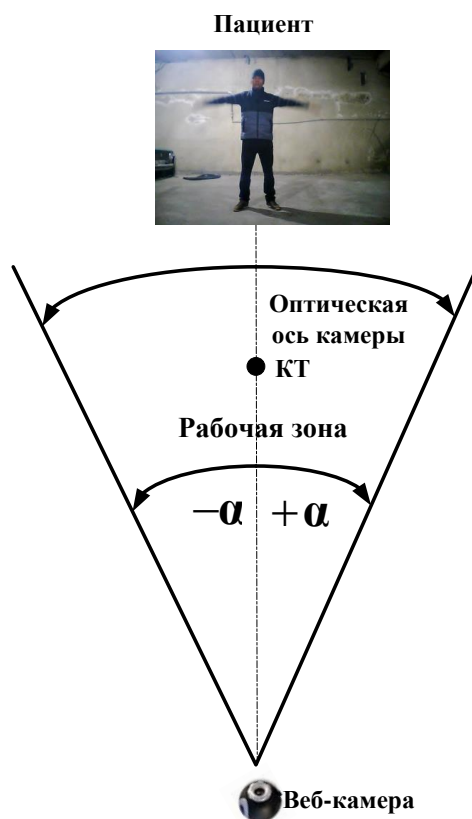


Рисунок 4.11 – Рабочая зона системы (калибровочная точка КТ соответствует расстоянию "камера-пациент" – 2,5 м и углу отклонения от оптической оси  $0^\circ$ )

Таблица 4.1 – Определение границ рабочей зоны

Зависимость от расстояния		Зависимость от угла	
Расстояние от калибровочной точки	Процент ошибочных решений, %	Отклонение от оптической оси	Процент ошибочных решений, %
- 0,7 м (1,8 м)	13	-14°	19
- 0,6 м (1,9 м)	8	-12°	16
-0,5 м (2,0 м)	8	-10°	11
- 0,4 м (2,1 м)	6	-8°	6
- 0,3 м (2,2 м)	6	-6°	6
- 0,2 м (2,3 м)	6	-4°	6
- 0,1 м (2,4 м)	6	-2°	6
КТ(2,5 м)	6	0°	6
+0,1 м (2,6 м)	6	+2°	6
+0,2 м (2,7 м)	6	+4°	6
+0,3 м (2,8 м)	6	+6°	6
+0,4 м (2,9 м)	6	+8°	6
+0,5 м (3,0 м)	9	+10°	9
+0,6 м (3,1 м)	14	+12°	13

Из таблицы 4.1 видно, что рабочая зона системы ограничивается по углу секторами  $\pm 8^\circ$  относительно оптической оси, а по расстоянию ограничена смещениями относительно калибровочной точки  $КТ \pm 0,4$  м (2,1-2,9 м от камеры в абсолютном выражении). За пределами рабочей зоны процент ошибочных решений резко увеличивается.

Из таблицы видно, что рабочая зона системы ограничивается по углу секторами  $\pm 8^\circ$  относительно оптической оси, а по расстоянию ограничена смещениями относительно калибровочной точки  $KT \pm 0,4$  м (2,1-2,9 м от камеры в абсолютном выражении). За пределами рабочей зоны процент ошибочных решений резко увеличивается.

#### 4.4.3 Оценка показателей достоверности принятия решений

Для экспериментального определения порогов, по которым оценивается близость обучающего и выполненного движений, врачом-инструктором выполнены 7 групп типовых движений с 10-кратным повторением каждого движения. Экспериментально установленные пороги сохранены в БД системы.

Показатели достоверности принятия решения разработанной СППР в автоматическом режиме оценены на видеопотоках с эталонными и текущими движениями с объемом выборки объемом 100 парных записей упражнений ДДР, выполненных по методикам [116,117] в автоматическом режиме и с участием врача-эксперта, решения которого считаются абсолютно правильными, установлено, что вероятность ошибочного решения в автоматическом режиме не превышает 0,06. Это на 0,06 ниже, чем в известной аналогичной системе [118].

Для оценки коэффициента повышения достоверности принятия решений с применением разработанной СППР в автоматизированном режиме был поставлен эксперимент с участием четырех врачей: эксперта-реабилитолога и трех врачей-реабилитологов с опытом работы менее трех лет. Экспертном и врачами рассмотрена та же выборка эталонных и текущих двигательных упражнений объемом 100 парных записей. Решения по оценке правильности выполнения движений и их коррекции, принятые экспертом, считались абсолютно правильными.

В таблице 4.2 приведены показатели оценки этих же движений тремя врачами-участниками эксперимента.

Таким образом, достоверность принятия решений с применением разработанной СППР увеличивается на величину

$$K_{нов.досм}^{СППР} = \left(1 - \frac{K_{right}^{DSS}}{K_{right}^{vis}}\right) \cdot 100\% = (1 - 0,85/0,54) \cdot 100\% = 57\%$$

по сравнению с достоверностью решений, принимаемых только на основании визуального анализа и опыта ЛПР.

Таблица 4.2 – Оценка достоверности принятия решений без применения и с применением разработанной СППР

Количество ошибочных решений, допущенных участниками эксперимента				
Участники эксперимента	Визуальный анализ без применения СППР	Коэффициент достоверности (доля безошибочных решений) $K_{right}^{vis}$	Анализ с применением СППР (автоматизированный режим)	Коэффициент достоверности (доля безошибочных решений) $K_{right}^{DSS}$
Врач_1	34	0,66	11	0,89
Врач_2	41	0,59	17	0,83
Врач_3	37	0,63	14	0,86
Средние показатели		0,54		0,85

## Выводы по разделу 4

1. На основании разработанных базовых информационных технологий БИТ1, БИТ2, БИТ3 создана интегрированная информационная технология поддержки принятия решений для дистанционной двигательной реабилитации.

2. Разработана информационная технология ППР при ДДР. ИТ предусматривает три режима функционирования:

- подготовительный,
- автоматический,
- автоматизированный.

3. На основании разработанной информационной технологии разработана распределенная система ППР для ДДР. Распределенная СППР выполнена в виде клиент-серверного приложения.

4. Проведено тестирование разработанной СППР для ДДР, проведены процедуры масштабной и антропометрической калибровки системы. Оценена рабочая зона системы. Определены пороги для сравнения кинематических моделей при вынесении решения.

5. Проведена оценка показателей качества разработанной СППР с участием врачей-реабилитологов. Установлено, что:

- в автоматическом режиме вероятность ошибочного решения в автоматическом режиме не превышает 0,06, что на 0,06 ниже, чем в известных аналогичных системах;

- в автоматизированном режиме достоверность принятия решения увеличивается на 57 % по сравнению с достоверностью решений, принимаемых только на основании визуального анализа и опыта ЛПР.

## ВЫВОДЫ

В диссертационной работе решена актуальная научно-практическая задача разработки моделей и методов для компьютерной поддержки принятия решений при двигательной реабилитации пациентов. Автором получены следующие научные и практические результаты.

1. Анализ существующих систем ППР при ДРР показал, что они выдают как правило бинарное решение ("верно / неверно") без оценки количественных показателей, модели движения человека в видеопотоке сложны, и вместе с тем слабо формализованные, недостаточно точно описывают динамику движения и особенности построения двигательного образа. Методы количественного сравнения движений сложные и практически неприменимы для врача, который является лицом, принимающим решения. На основании результатов анализа сформулированы и обоснованы цели и задачи, проблемы и пути их решения, выбрано направления разработки системы поддержки принятия решения и информационных технологий для ее реализации.

2. Разработана двумерная 16-ти точечную проективную модель и информационная технология для покадрового представления элемента движения человека в видеопотоке, основанная на покадровой обработке скелетов изображения тела. Исследованы показатели качества обнаружения, разработанная модель обеспечивает вероятности правильного обнаружения при обнаружении ХТ не ниже 0,97 при вероятности ложных тревог не более 0,04, она позволяет увеличить вероятность правильного выделения ХТ на 0,01 и уменьшить вероятность выделения ложных точек на 0,02 по сравнению с известными аналогами.

3. Разработана кинематическая модель движения человека в видеопотоке, основанная на аппроксимации траекторий характерных точек в 3-х кадровых фрагментах видеопотока дугами кругов, что позволило

сформулировать метод оценки тангенциальных скоростей и тангенциальных ускорений ХТ тела человека.

4. Разработан метод построения статистических оценок кинематических параметров путем усреднения фрагментов кинематической матрицы методом сдвига среднего. Установлены состоятельность, несмещенность и эффективность оценок кинематических параметров, получаемых с помощью метода. Показано, что при расчете скользящего среднего не менее, чем по 10 трехкадровым фрагментам видеопотока оценки параметров практически не смещены, коэффициент вариации оценок координат ХТ не превышает 3,2%, скорости - 4,1%, ускорение - 4,8%. Таким образом экспериментально подтверждены адекватность построенной кинематической модели движения тела и метода построения статистических оценок кинематических параметров.

5. На основе предложенных моделей и методов разработан информационную технологию поддержки принятия решений при дистанционной двигательной реабилитации, предназначенная для функционирования в автоматическом и автоматизированном режимах. ИТ ППР реализована в виде распределенной СППР, работающей в реальном времени с использованием экономичных компьютерных ресурсов.

6. В результате испытаний системы ППР установлено, что применение разработанной ИТ в автоматическом режиме дает не более 6% ошибочных решений и позволяет повысить достоверность принятого решения на 57 % в автоматизированном режиме.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Нгуен Г. К. Принципи побудови комп'ютерних систем дистанційного тренування на основі аналізу відеопотоку/ Нгуен Г.К., В. О. Болтънков, Д. В. Малявін. – Восточно-Европейский журнал передовых технологий. – 2014. – №5/2 (71). – С. 25-33.
2. Болтенков В.А. Двумерная проективная модель движения тела человека и ее применение в задачах телемедицины / В.А. Болтенков , Нгуен Гуи Кионг. – Информатика та математичні методи в моделюванні. – 2014. – Том 4. – №4. – С. 312–323.
3. Болтенков В.А. Анализ алгоритмов скелетизации бинарных изображений. / В.А. Болтенков, Нгуен Гуи Кионг, Д.В. Малявин – 2015. – Электротехнические и компьютерные системы. – № 17 (93). – С. 102–109.
4. Нгуен Г. К. Информационная технология дистанционной двигательной реабилитации / Г. К. Нгуен. – 2015. – Электротехнические и компьютерные системы. – № 18 (94). – С. 21–27.
5. Болтенков В.А. Архитектура компьютерной системы дистанционной двигательной реабилитации / В.А. Болтенков, Г. К. Нгуен. – 2015. – Комп'ютерно-інтегровані технології: освіта, наука, виробництво. – Вип. 19. – С. 91–95.
6. Нгуен Г. К. Применение систем компьютерного зрения в задачах реабилитации пациентов с болезнями опорно–двигательного аппарата . / Г. К. Нгуен // Труды Междун. научно-практ. конференции «Современные информационные и электронные технологии» (СИЭТ-2013). – Одесса, 2013. – Т.1. – С. 53–54.
7. Нгуен Г. К. Анализ двигательной активности человека в видеопотоке/ Г. К. Нгуен, В.А. Болтенков // Інформаційні управляючі системи та технології (ПУСТ-2013). – Одеса, ОНМА, 2013. – С.69–71.
8. Болтенков В.А., Малявин Д.В., Нгуен Г. К. Сравнительный анализ алгоритмов скелетизации бинарных изображений / В.А. Болтенков , Д.В.



- Малявин, Г. К. Нгуен // Труды Междун. научно-практ. конференции «Современные информационные и электронные технологии» (СИЭТ-2014). – Одесса, 2014. – Т.1. – С.40–41.
9. Малявин Д.В. Траекторное сопровождение движений человека / Д.В. Малявин, Г. К. Нгуен, В.А. Болтенков // Сучасні інформаційні технології 2014 (МІТ-2014) / Матеріали четвертої Міжн. конференції студентів і молодих науковців, 22–26 квітня 2014 р./ – Одеса, , 2014. – С.99–100.
10. Antoshchuk S.G. The computer system for distance medical motion rehabilitation / S.G. Antoshchuk, V.A. Boltenkov, N. H. Kiong // 8-th International Forum on Innovative Technologies for Medicine (ITMED 2014) – Poland, Suprasl, 2014. – P.13.
11. Нгуен Г.К. Информационная технология дистанционной двигательной реабилитации / Г. К. Нгуен , В.А. Болтенков // Третя українсько-німецька конференція «Інформатика. Культура. Техніка»: Збірник тез доповідей. — Одеса: ОНПУ, 2015. — С. 17–18.
12. Назаренко Г. И. Медицинские информационные системы: теория и практика / Г. И. Назаренко, Я. И. Гулиев Д. Е.Ермаков – М. : Физматлит, 2005. – 320 с.
13. Кобринский Б.А. Медицинская информатика / Б.А. Кобринский, Т.В. Зарубина – М.: Издат. центр "Академия", 2009. – 192 с.
14. Zhou H. Human motion tracking for rehabilitation — A survey / H. Zhou, H. Hu // Biomedical Signal Processing and Control. – 2008. – No.3. – P. 1–18.
15. Rosen M. Telerehabilitation / M. Rosen // NeuroRehabilitation. – 1999. – Vol. 12, No.1. – P. 11-26.
16. E-health care information systems: an introduction for students and professionals. – N.-Y. : John Wiley and Sons, 2005. – 219 p.
17. Jaume-i-Capó A. Design Issues for Vision-Based Motor-Rehabilitation Serious Games / A. Jaume-i-Capó, B. Moyà-Alcover, J. Varona. // Technologies of Inclusive Well-Being. Studies in Computational Intelligence No. 536.– Berlin-Heidelberg : Springer-Verlag, 2014. – P. 13-24.

18. Keaton L. An E-rehabilitation Team Helps Caregivers Deal with Stroke / L. Keaton, L. L. Pierce, V. Steiner, K.Lance // *The Internet Journal of Allied Health Sciences and Practice* . – 2004. – Vol. 2 , No. 4. – P.1-9.
19. C.-J. Su. Personal Rehabilitation Exercise Assistant with Kinect and Dynamic Time Warping / C.-J. Su // *International Journal of Information and Education Technology*. – 2013. – Vol. 3, No. 4. – P. 448-454.
20. Pianykh O.S. *Digital Imaging and Communication in Medicine (DICOM)*. – Berlin: Springer-Verlag, 2008. – 383 p.
21. Тоценко В.Г. Методы и системы поддержки принятия решений. Алгоритмический аспект / В.Г. Тоценко .– К. : Наукова думка, 2002.– 382 с.
22. Щаденкова М.Ф. Системы анализа движений на основе видеозахвата / М.Ф. Щаденкова // *Вестник Полоцкого государственного университета. Серия С. Фундаментальные науки. Информатика*. – 2010. – № 3. – С. 30-33.
23. H. Zeng. Sensing Movement: Microsensors for Body Motion Measurement // H. Zeng and Y. Zhao // *Sensors*. – 2011. – No. 11. – P. 638-660.
24. Pan J.-I. Intelligent Shoulder Joint Home-Based Self-Rehabilitation Monitoring System / J -I. Pan, H.-W.Chung, J.-J. Huang // *International Journal of Smart Home* . – 2013. – Vol.7. – No.5.– P.395-404.
25. Doshi S. S. Applications of Inertial Measurement Units in Monitoring Rehabilitation Progress of Arm in Stroke Survivors / Master's thesis. – 2011. – Colorado State University. – 123 p.
26. Macedo P. A Telerehabilitation System based on Wireless Motion Capture Sensors / P. Macedo, J.Afonso, L.A. Rocha, R. Simoes . // *4th International Conference on Sensor Networks (SENSORNETS 2015)*. – 2015. –Angers, France. – P. 55-62.
27. Amirabdollahian F. Design, development and deployment of a hand/wrist exoskeleton for home-based rehabilitation after stroke - SCRIPT project F. Amirabdollahian, S. Ates, A. Basteris, A. Cesario at al. // *Robotica* . – 2014. –Vol. 32, No.8 – P. 1331-1346.

28. Gypsy Motion Capture System [Электронный ресурс]. – Режим доступа: <http://www.metamotion.com/gypsy/gypsy-motion-capture-system.htm>.
29. Уменьшение случайных ошибок измерений, связанных с креплением маркеров на теле испытуемых [Электронный ресурс]. – Режим доступа: <http://biosoftvideo.ru/marker/>
30. CMU Graphics Lab Motion Capture Database [Электронный ресурс]. – Режим доступа: <http://mocap.cs.cmu.edu/info.php>
31. Pawlik P. Visual Surveillance for Movement Rehabilitation / P. Pawlik , Z.Bublinski // Image Processing & Communication – 2012 – Vol. 17, No. 4 – P. 173-178.
32. Воронов А.В. Компьютерный видеоанализ движений в научных исследованиях и клинической практике // А.В. Воронов, В.И. Доценко , Н.Ю. Титаренко , К.Е.Титаренко // Труды Седьмой Всероссийской конференции по биомеханике «Биомеханика-2004». – М., Изд-во МГУ, 2004. – С.36-38.
33. Бобылев А.Н. Восстановление утерянных показаний системы видеоанализа движений с использованием измерений акселерометра / А.Н. Бобылев А.Н., А.В. Воронов, П.А. Кручинин // Всерос. конф. по биомеханике «Биомеханика-2010»: тез. докл. – Саратов: Изд-во СГУ, 2010. – С. 46–47.
34. Воронов А.В. Методические аспекты анализа локомоций с использованием аппаратно-программного комплекса «видеоанализ движений / А.В. Воронов [Электронный ресурс]. – Режим доступа: <http://www.ordas.ru/index.php?id=50>.
35. Воронов А.В. История биомеханической видеосъемки / А.В. Воронов // [Электронный ресурс]. – Режим доступа: <http://www.biosoftvideo.ru/history>
36. Ferrigno G. ELITE: a digital dedicated hardware system for movement analysis via real-time TV signal procession / G. Ferrigno , A. Pedotti // IEEE Transactions on Biomedical Engineering. – 1985. – Vol. BME-32, No. 11. – P. 943-950.

37. Bruderlin A. Motion Signal Processing / A. Bruderlin, L. Williams // Proceedings of the 22nd Annual Conference on Computer Graphics and Interactive Techniques SIGGRAPH '95 . – N. Y.: ACM, 1995. P. 97-104
38. Rafiuzzaman M. Distance Physical Rehabilitation System Framework with Multi-Kinect Motion Captured Data / M. Rafiuzzaman, C. Öz // Communications on Applied Electronics (CAE) – 2015. – Vol. 1 , No.5 – P. 29-39.
39. Webster D. Systematic review of Kinect applications in elderly care and stroke rehabilitation / D. Webster, O. Celik // Journal of NeuroEngineering and Rehabilitation. – 2014. – Vol.11, No.108 – P.1-24.
40. Wang J.J. Video analysis of human dynamics—a survey / J.J. Wang, , S.Singh // Real-Time Imaging. – 2003. – Vol. 9, No. 5. – P. 321–346.
41. Moeslund T.B. A survey of advances in vision-based human motion capture and analysis / T. B. Moeslund, A. Hilton, V. Kruger // Computer Vision and Image Understanding. – 2006.– Vol. 104, No. 2–3. – P. 90–126.
42. Perez-Sala X. A Survey on Model Based Approaches for 2D and 3D Visual Human Pose Recovery / X. Perez-Sala, S. Escalera, C. Angulo at al. // Sensors. – 2014. – No. 14. – P. 4189-4210.
43. Rosenhahn B. A Silhouette Based Human Motion Tracking System / B. Rosenhahn, U. G. Kersting, L. He, A W. Smith,T. Brox, R. Klette, H.-P Seidel // Lecture Notes in Computer Science, Vol. 3175.– Berlin: Springer, 2004. – P. 294-304.
44. 17a6B Oinam B.D. Robust Statistical Approach for Extraction of Moving Human Silhouettes from Videos / B . D. Oinam , S. P. Nissi , Y. J. Singh // International Journal on Information Theory (IJIT). – 2014. – Vol.3, No.3. – P. 55-64.
45. Panagiotakis . Unsupervised Human Members Tracking Based on an Silhouette Detection and Analysis Scheme / C. Panagiotakis, A. Doulamis // AIAI Workshops, Vol. 475 of CEUR Workshop Proceedings. CEUR-WS.org, 2009. – CEUR-WS.org . – P. 134-141.

46. Lee M.W. Human Pose Tracking in Monocular Sequence Using Multilevel Structured Models / M. W. Lee, R. Nevatia // IEEE Transactions on Pattern Analysis and Machine Intelligence. – 2009. – Vol. 31, No. 1. – P. 27-38.
47. Muthukumar B. Tracking the Human Motion in Real Time Using Star Skeleton Model / B. Muthukumar, S.Ravi // International Journal of Engineering and Advanced Technology. – 2012. – Vol.1, No. 3 – P. 2249 – 2958.
48. Uddin M. Z. Human Activity Recognition Using Body Joint-Angle Features and Hidden Markov Model / M. Z. Uddin, N. D. Thang, J. T. Kim // ETRI Journal. – 2011. – Vol. 33, No.4. – P.569-579.
49. Caillette F. Real-Time 3-D Human Body Tracking using Learnt Models of Behaviour / F. Caillette, A. Galata, T. // Computer Vision and Image Understanding (CVIU) Journal. – 2008. – Vol. 109, No. 2. – P.112 - 125.
50. Bharathram C. Human Action Recognition Based on Enhanced Dynamic Time Warping / C. Bharathram C., S. Chitrakala // International Journal of Advanced Technology in Engineering and Science. – 2015. – Vol. 3, No. 2. – P. 746-755
51. Devi O. B. Robust Statistical Approach for Extraction of Moving Human Silhouettes From Videos // O. B. Devi, N. S. Paul, Y. J Singh // International Journal on Information Theory (IJIT). – 2014. – Vol.3, No.3. –P. 55-64.
52. He X. Defined Human Pose Detection or Video Surveillance / X. He, Zh. Wang, B. -S. Shin, R.Klette // Proceedings of the 29th International Conference on Image and Vision Computing. New Zealand. – 2014. – P. 160-165.
53. Chang Y. J. A Kinect-based system for physical rehabilitation: A pilot study for young adults with motor disabilities / Y. J. Chang, S. F. Chen, A. F. Chuang // Research in Developmental Disabilities. – 2011. – Vol. 32, No. 6. – P.2566-2570.
54. Lange B. Development and Evaluation of Low Cost Game-Based Balance Rehabilitation Tool Using the Microsoft Kinect Sensor / B. Lange, C. Y. Chang, E. Suma, B. Newman, A. S. Rizzo, M. Bolas // Proc. 33rd Annual International Conf. of the IEEE EMBS, Boston . – 2011. – P. 1831-1834.

55. Wu K. K. Using Human Skeleton to Recognizing Human Exercise by Kinect's Camera / K. K. Wu . Master thesis. National Taipei Univ. of Technology, Taiwan. –2011. – 184 p.
56. Гонсалес Р. Цифровая обработка изображений / Р. Гонсалес, Р. М. Вудс. – М.: Техносфера, 2005. – 606 с.
57. Яне Б. Цифровая обработка изображений / Б. Яне / – М.: Техносфера, 2007. – 584с.
58. Фисенко В.Т. Компьютерная обработка и распознавание изображений / В.Т. Фисенко, Т.Ю. Фисенко. – СПб: СПбГУ ИТМО, 2008. – 192 с.
59. Методы компьютерной оптики / А.В. Волков, Д.Л. Головашкин, Л.Л. Досколович, Н.Л. Казанский и др. Под ред. В.А. Сойфера. – М. : ФИЗМАТЛИТ, 2003. - 688 с.
60. Введение в контурный анализ; приложения к обработке изображений и сигналов / Я.А.Фурман, А.В Кривецкий, А.К. Передреев, А.А. Роженцов и др. – М.: ФИЗМАТЛИТ, 2003. — 592 с.
61. Визильтер Ю.В. Обработка и анализ изображений в задачах машинного зрения / Ю.В. Визильтер, С.Ю. Желтов, А.В. Бондаренко, М.В. Ососков М.В. и др. – М.: Физматкнига, 2010. – 672 с.
62. Blum H. A transformation for extracting new descriptors of shape / H. Blum // Models for the Perception of Speech and Visual Form. – Cambridge, MA.: MIT Press, 1967. – P. 362-380.
63. Serra J. Image Analysis and Mathematical Morphology/ J. Serra. – London–New York: Academic Press, 1982. – 485 p.
64. Местецкий Л. М. Непрерывная морфология бинарных изображений. Фигуры. Скелеты. Циркуляры / Л. М. Местецкий. – М. : ФИЗМАТЛИТ. – 2009. – 287 с.
65. Rosenfeld A. Computer representation of planar regions by their skeletons / A. Rosenfeld, J.L. Pfalz J.L // Communications of ACM. – 1967. – Vol.10, No. 2. – P.119 –125.

66. Wang T. Analysis and Application of Iteration Skeletonization Algorithm in Recognizing Chinese Characters Image / T. Wang , G. Chen , Z. Chen // Proceedings of the Second Symposium International Computer Science and Computational Technology (ISCSCT '09) , Huangshan, 2009. – P. 281-284.
67. Davies E.R. Computer and Machine Vision : Theory, Algorithms, Practicalities / E.R. Davies. – Academic Press, 2012. – 912 p.
68. Лукьяница А.А. Цифровая обработка видеоизображений / А.А. Лукьяница, А.Г. Шишкин. – М.: Ай-Эс-Эс Пресс, 2009. – 518 стр.
69. Saad M. H. M. Essential Human Body Tracking Using Kalman Filter / M. H. M. Saad // Proceedings of the World Congress on Engineering and Computer Science IWCECS 2013. – 2013. –Vol.1 – P. 23-25.
70. Chang I-C. Skeleton-based Walking Motion Analysis Using Hidden Markov Model and Active Shape Models / I-C. Chang , C.-L. Huang //Journal of Information Science and Engineering. – 2001. – Vol. 17. – P.371-403.
71. Lin J. F.-S. Automated Rehabilitation Exercise Motion Tracking / J. F.-S. Lin. Thesis Master of Science. University of Waterloo, 2012. – 98 p.
72. J. Singh. Real Time Tracking of Human Poses using Advanced Kalman Filter /J. Singh, F. Shivani // National Conference on Synergetic Trends in engineering and Technology (STET-2014). Vol.2.– P. 275-277.
73. Win K. Essential Human Body Points Tracking Using Kalman Filter / K.Win, H. Aini , Saad M. H. M. // Proc. of the World Congress on Engineering and Computer Science:. WCECS 2013, San Francisco. Vol. 1. – P. 503-507.
74. Suliman C. Kalman Filter Based Tracking in an Video Surveillance System / C. Suliman, , C. Cruceru, F. Moldoveanu // Advances in Electrical and Computer Engineering. – 2010.– Vol. 10, No. 2. – P. 30-34.
75. Agarwal P. An Optimization Based Framework for Human Pose Estimation in Monocular Videos / Proceedings of International Symposium on Visual Computing, 2012. – P. 12-17.

76. Xu R. Combining Skeletal Pose with Local Motion for Human Activity Recognition / R. Xu, P. Agarwal, S. Kumar at al. // Proceedings of VII Conference on Articulated Motion and Deformable Objects, 2012. – P. 412-416.
77. Agarwal P. Estimating Human Dynamics On-The-Fly Using Monocular Video For Pose Estimation / P. Agarwal, S. Kumar, J. Ryde at al. // Proceedings of Robotics Science and Systems, 2012. – P. 311-317.
78. Fleet D. J. Motion Models for People Tracking / D. J. Fleet, M. A. Brubaker, L. Sigal // Visual Analysis of Humans. Springer. – 2011. – P. 171-198.
79. Fleet D. J. Video-Based People Tracking // Handbook of Ambient Intelligence and Smart Environments . Springer, 2010. – P. 57-87.
80. Bartlett R. Introduction to Sports Biomechanics: Analysing Human Movement Patterns / R. Bartlett – N.-Y., Routledge. – 2007.– 292 p.
81. Потемкин С.Б. Восстановление пространственного спортивного движения по его двумерному образу / С. Б. Потемкин // Труды Междисциплинарного научного семинара "Информатика и спорт" – М.: Изд. МГУ , 2010 – С.132-135.
82. Катаев М.Ю. Построение скелетной модели фигуры человека по потоку изображений / М.Ю. Катаев , А.П. Коробко // Доклады ТУСУРа. – 2014. – № 4 (34). – С. 109-113.
83. Waleed A.-A. Skeletonization Algorithm for Binary Images, Procedia Technology / A.-A. Waleed, N. H. S. A. Siti, B. Bilal at al. // Procedia Technology. – 2013. – Vol.11. – P.704 – 709.
84. Zhang T.Y. A Fast Parallel Algorithm for Thinning Digital Patterns / T.Y. Zhang , C.Y. Suen // Communications of ACM. – 1984. – Vol. 27, No. 3.– P. 236 –39.
85. Guo Z. Parallel Thinning with Two Subiteration Algorithms / Guo Z., Hall R.W. // Communications of the ACM. –1989. – Vol. 32, No. 3. – P. 359 – 373.
86. Stentiford F.W.M. Some New heuristics for thinning binary hand printed characters for OCR / F.W.M. Stentiford , R.G. Mortimer // IEEE Transactions on Systems, Man, and Cybernetics. – 1983.– Vol. 13, No. 1. – P. 81 – 84.



87. Щепин Е. В. . К топологическому подходу в анализе изображений / Е. В. Щепин, Г. М. Непомнящий. Геометрия, топология и приложения", Межвузовский сборник научных трудов.– М. : – Московский институт приборостроения , 1990. – С. 13–25.
88. Домахина Л. Г. Регуляризация скелета для задачи сравнения формы / Л. Г. Домахина // Математические методы распознавания образов: доклады XIV Всерос. конф. – М. , 2009. – С. 342 – 346.
89. Рогов А.А. Некоторые методы классификации и поиска в электронной коллекции графических документов / А. А. Рогов, К. А. Рогова, П. В. Кириков, М. Ю. Быстров // Труды 12-й Всероссийской научной конференции «Электронные библиотеки: перспективные методы и технологии, электронные коллекции». – Казань : КГУ, 2010. – С. 409 – 414.
90. Баша Н.С. Методическое обеспечение верификации личности по индивидуальным особенностям сосудистого рисунка и геометрическим пропорциям лица человека в естественных условиях: автореф. дис. на соискание уч. степени канд. техн. наук : спец. 05.13.01 – «Системный анализ, управление и обработка информации» (в науке и промышленности) / Н.С. Баша. – Дубна, Международный университет природы, общества и человека – 2013. – 22 с.
91. Olsen M.A. Convolution Approach for Feature Detection in Topological Skeletons Obtained from Vascular Patterns / M. A. Olsen, D. Hartung, C. Busch and R.Larsen // IEEE Workshop on Computational Intelligence in Biometrics and Identity Management – CIVIM 2011. – P. 163-167.
92. Дронг В. И. Курс теоретической механики / В. И. Дронг – М.: МГТУ имени Н. Э. Баумана, 2005. – 736 с.
93. Умнов А. Е. Аналитическая геометрия и линейная алгебра / А. Е. Умнов. – М. : МФТИ, 2011. – 544 с.
94. Гантмахер Ф. Р. Лекции по аналитической механике / Ф. Р. Гантмахер – М.: Физматлит, 2005. – 264 с.

95. Эльясберг П. Е. Определение движения по результатам измерений / П. Е. Эльясберг. – М. : Книжный дом "ЛИБРОКОМ", 2011. – 416 с.
96. Жданюк Б.Ф. Основы статистической обработки траекторных измерений / Б.Ф. Жданюк. – М.: : Советское радио, 1978. – 384 с.
97. Kustikova V.D. Vehicle Video Detection Method / V.D. Kustikova, I.B. Meyerov, N.Y. Zolotykh // Proceedings of the 11th International Conference "Pattern Recognition and Image Analysis: New information technologies" (PRIA-11-2013). – 2013. – V. 1. – P. 232– 235.
98. Калиткин Н. Н. Численные методы / Н. Н. Калиткин — СПб.: БХВ-Петербург, 2011. – 592 с..
99. Шишкова Е.В. Регуляризация задачи численного дифференцирования // Сб. науч. тр. Механика. Математика, Саратов: Изд-во Саратов. ун-та, 2008. – С. 99-101.
100. Светозаров В.В. Основы статистической обработки результатов измерений / В.В. Светозаров. – М.: Изд. МИФИ, 2005. – 40 с.
101. Боровиков В. STATISTICA : искусство анализа данных на компьютере. – СПб.: Питер, 2001 . – 656 с.
102. Горяинова Е. Р. Прикладные методы анализа статистических данных / Е. Р. Горяинова, А. Р. Панков, Е. Н. Платонов. М.: Изд. ВШЭ, 2012. – 310 с.
103. Закс Л. Статистическое оценивание. / Л. Закс. – М.: Статистика, 1976. — 600 с.
104. Кобзарь А.И. Прикладная математическая статистика / А.И. . Кобзарь. – М.: ФИЗМАТЛИТ, 2006. - 816 с.
105. Нестеров А.В. Проверка отклонения распределения вероятностей от нормального распределения по критерию Шапиро-Уилка в MATLAB / А.В. Нестеров, С.В.Нестеров , Д.А. Нестерова // Некоторые вопросы математики и ее приложений: Сб. науч. тр. Вып. 1. – Магнитогорск: Магнитогорск: Изд-во Магнитогорск. гос. техн. ун-та им. Г.И. Носова, 2013. – С. 41-45.

106. Аграновский К.Ю. Основы теории радиоэлектронных систем морских объектов / К.Ю. Аграновский, П.И. Киселев, Е.А. Свядоц. – Л.: Судостроение, 1974. – 351 с.
107. Дюран Б., Оделл П. Кластерный анализ / Б. Дюран , П. Оделл .– М.: Статистика, 1977. – 128 с.
108. Дуда Р. Распознавание образов и анализ сцен / Р. Дуда, П. Харт – М.: Мир, 1976. – 512 с.
109. Деза Е. Энциклопедический словарь расстояний / Е. Деза , М.М. Деза – М.: Наука, 2008. – 446 с.
110. Уздин Д. З. Новые меры близости, функции состояний и решающие правила в теории распознавания образов ( состояний ) / Д. З. Уздин . – М.: Макс Пресс , .2015.– 95 с.
111. Open CV [Электронный ресурс] // Режим доступа: <http://opencv.org>.
112. Motion Analysis and Object Tracking (OpenCV 2.4.9.0 API) [Электронный ресурс]. – Режим доступа: [http://docs.opencv.org/modules/video/doc/motion\\_analysis\\_and\\_object\\_tracking.html#id1](http://docs.opencv.org/modules/video/doc/motion_analysis_and_object_tracking.html#id1).
113. Gritai A. Tracking Of Human Body Joints Using Anthropometry / A. Gritai , M. Shah.// IEEE International Conference on Multimedia And Expo ICME'2006 Proceedings, Toronto, Ontario, Canada . – P.34-41.
114. 66. Barron C. Estimating Anthropometry and Pose from a Single Uncalibrated Image / C. Barron, I. A. Kakadiaris // Computer Vision and Image Understanding . – 2001. – No. 81. – P. 269–284.
115. Dempster W. T. Properties of Body Segments Based on Size and Weight/ W. T. Dempster, G. R. L. Gaughran // American Journal of Anatomy. – 1967. – No.120/ – P.33-54.
116. Белова А.Н. Нейрореабилитация: руководство для врачей / А.Н. Белова. – М.: Антидор. – 2000. – 568 с.
117. Медицинская реабилитация: Руководство для врачей / Под ред. В.А. Епифанова. – М. : МЕДпресс-информ, 2008. – 352 с.

118. Checka N. AVIMA: Automatic Video-Based Motion Analysis / N. Checka // NASA Human Research Program Investigators' Workshop 2014 . – P. 3109.

## ПРИЛОЖЕНИЕ А

## Ключевые фрагменты программного кода

## А.1 Вычитание фона

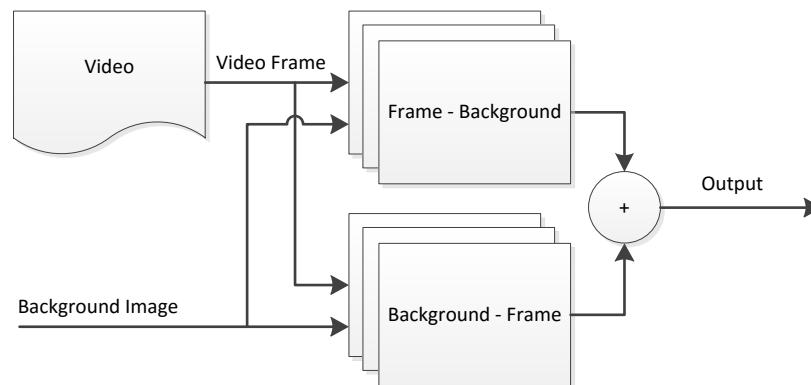


Рисунок А.1 – Логическая диаграмма процесса вычитания фона

Процесс выполняется с помощью функции `subtractVideoImage()` класса

**Rehabilitation:**

```

int Rehabilitation::subtractVideoImage(string inputVideoPath, string
inputImagePath, bool show)
{
    mBgImagePath = inputImagePath;
    mVideoPath = inputVideoPath;
    mSubtractedVideoPath = inputVideoPath + "_subtracted.avi";

    //read input video
    VideoCapture inputVideo(inputVideoPath);
    if (!inputVideo.isOpened())
    {
        cout << "Error: fail to read input video file" << endl;
        waitKey();
        return -1;
    }

    //read input background image
    Mat backgroundImage = imread(inputImagePath);

    if (show) {
        namedWindow("Background Image", CV_WINDOW_AUTOSIZE);
        imshow("Background Image", backgroundImage);

        namedWindow("Input Video", CV_WINDOW_AUTOSIZE);
        namedWindow("Output Video", CV_WINDOW_AUTOSIZE);
    }
}

```

```

}

VideoWriter writer(
    mSubtractedVideoPath,
    inputVideo.get(CV_CAP_PROP_FOURCC),
    inputVideo.get(CV_CAP_PROP_FPS),
    Size((int)inputVideo.get(CV_CAP_PROP_FRAME_WIDTH),
(int)inputVideo.get(CV_CAP_PROP_FRAME_HEIGHT)),
    true // is color
);
if (!writer.isOpened())
{
    cout << "Error: fail to write the video" << endl;
    waitKey();
    return -2;
}

Mat frame, sub1, sub2;
while (1)
{
    inputVideo >> frame;
    if (frame.empty()) break;

    if (show) imshow("Input Video", frame);

    subtract(backgroundImage, frame, sub1);
    subtract(frame, backgroundImage, sub2);
    add(sub1, sub2, frame);

    if (show) imshow("Output Video", frame);

    char c = cvWaitKey(15);
    if (c == 27) break; //if ESC key pressed, then break

    writer << frame;
}

return 0;
}

```

## А.2 Пороговая бинаризация

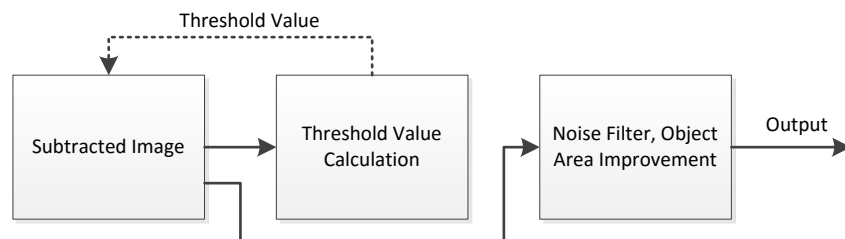


Рисунок А.2 – Логическая диаграмма процесса пороговой бинаризации

Это процесс выполняется с помощью функции `defineObjectArea()` класса

### Rehabilitation:

```
int Rehabilitation::defineObjectArea(string inputVideoPath, bool show)
{
    if (inputVideoPath != "") mSubtractedVideoPath = inputVideoPath; //if
NOT empty inputVideoPath, that means need to process with new input

    mObjectAreaVideoPath = mSubtractedVideoPath + "_objectarea.avi"; //

    //read input video
    VideoCapture inputVideo(mSubtractedVideoPath);
    if (!inputVideo.isOpened())
    {
        cout << "Error: fail to read input video file" << endl;
        waitKey();
        return -1;
    }

    //setup output video / writer
    VideoWriter outputVideo(
        mObjectAreaVideoPath,
        CV_FOURCC('M', 'J', 'P', 'G'), //inputVideo.get(CV_CAP_PROP_FOURCC),
        25.0, //inputVideo.get(CV_CAP_PROP_FPS),
        Size((int)inputVideo.get(CV_CAP_PROP_FRAME_WIDTH),
(int)inputVideo.get(CV_CAP_PROP_FRAME_HEIGHT)), //640x640
        false // is NOT color
    );
    if (!outputVideo.isOpened())
    {
        cout << "Error: fail to write the video" << endl;
        waitKey();
        return -2;
    }

    if (show) {
        namedWindow("Input Video", CV_WINDOW_AUTOSIZE);
        namedWindow("Ouput Video", CV_WINDOW_AUTOSIZE);
    }

    Mat frame;
    Mat grey_frame, blur_frame, bin_frame, object_frame;
    Mat eroded_frame, dilated_frame;
    Mat skeket_frame;

    Common common;

    int i = 0;
    while (1)
    {
        inputVideo >> frame;
```

```

        if (frame.empty()) break;
        if (show) imshow("Input Video", frame);

        cvtColor(frame, grey_frame, CV_RGB2GRAY); //convert to grey
        blur(grey_frame, blur_frame, Size(7,7), Point(-1,-1)); //smooth
grey frame
        bthreshold_value = threshold_value_calculation(blur_frame);
        threshold(blur_frame, bin_frame, 35, 255, CV_THRESH_BINARY);
//convert to binary

        //delete shadow (suppose that our subtracted video contains only
object area (not shadow)
        float x1 = 258.0, y1 = 428.0, x2 = 322.0, y2 = 438.0;
        for (int i = 0; i < bin_frame.rows; i++) //y
        {
            for (int j = 0; j < bin_frame.cols; j++) //x
            {
                if (i>442 || (j>344&&i>425) || (j>256 && (y2-y1)/(x2-
x1)*(j-x1)+y1 < i))
                {
                    bin_frame.at<uchar>(i, j) = 0;
                }
            }
        }

        common.getObjectArea(bin_frame, object_frame);

        erode(bin_frame, eroded_frame, cv::Mat(5,5,CV_8UC1), cv::Point(-
1,-1), 2, 0, morphologyDefaultBorderValue()); //delete noise
        dilate(eroded_frame, dilated_frame, cv::Mat(5,5,CV_8UC1),
cv::Point(-1,-1), 2, 0, morphologyDefaultBorderValue()); //

        if (show) imshow("Ouput Video", grey_frame);

        char c = cvWaitKey(11);
        if (c == 27) break;
        outputVideo << bin_frame;

        i++;
    }

    return 0;
}

```

Здесь используется функция `threshold_value_calculation()` для вычитания значения порога и функция `getObjectArea()` класса `Common` для фильтрации (кроме этих функций используются функции `erode()` и `dilate()` для устранения помех).

**Программные коды этих функций:**

```

const int gfHEIGHT = 480;
const int gfWIDTH = 640;

```



```

const int gfMAX_AREA_NUMBER = 307200;
const int gfSTEP = 640; //to speed up program

//mark the area's index (from 0 to infinity)
int gfWhiteAreaIndex = 0;    //white area index = 0, 1, 2, 3,...

//[x][y][0] = 0/1: not-check-yet/check pixel(x,y)
//[x][y][1]: index of white/black area that pixel(x,y) belongs to
int gfCheckedList[gfHEIGHT][gfWIDTH][2];

//array contains number of pixels of an area, init value = {0}
int gfPixelQuantity[gfMAX_AREA_NUMBER] = {0}; //int pixelQuantity[] = {0};

cv::Mat gfSrc;
int Common::getObjectArea(Mat src, Mat& dst) {
    int maxQuantity = 0;
    int maxWhiteAreaIndex = 0;

    gfWhiteAreaIndex = 0;
    //gfBlackAreaIndex = -1;
    for (int i=0; i<gfHEIGHT; i++) { //rows
        for (int j=0; j<gfWIDTH; j++) { //cols
            gfCheckedList[i][j][0] = 0;           //mark all pixel as
unchecked
            //gfCheckedList[i][j][1] = -1;       //no need
        }
    }

    gfSrc = src;//.clone();

    for (int i=0;i<src.rows;i++)
    {
        for (int j=0;j<src.cols;j++)
        {
            if (gfCheckedList[i][j][0] == 0) { //if point is not
checked yet
                gfCheckedList[i][j][0] = 1; //then mark pixel(i,j)
as checked

                if (gfSrc.data[gfSTEP * i + j] == 255) { //if it's
white point
                    gfCheckedList[i][j][1] = gfWhiteAreaIndex;
                    //then store the index of new white area (white area Index goes from 0,
1, 2,...)
                    gfPixelQuantity[gfWhiteAreaIndex] = 1;
                    //at first, quantity of are with gfWhiteAreaIndex = 1
                    gfCheckNextWhitePointOf(i,j);
                    //then find all neighbour white points
                    if (maxQuantity <
gfPixelQuantity[gfWhiteAreaIndex]) { //save the largest white area index
and it's white point quantity
                        maxQuantity =
gfPixelQuantity[gfWhiteAreaIndex]; //.. quantity
                        maxWhiteAreaIndex = gfWhiteAreaIndex;
                    }
                }
            }
        }
    }
    //.. index
}

```

```

        }
        gfWhiteAreaIndex++;
    } /*else {
    } */
    }
}

for (int i=0;i<src.rows;i++)
{
    for (int j=0;j<src.cols;j++)
    {
        if (gfCheckedList[i][j][1]!=maxWhiteAreaIndex) {
            gfSrc.data[gfSTEP * i + j] = 0;
        }
    }
}

dst = gfSrc.clone();
return 0;
}

```

**Функция getObjectArea() в свою очередь вызывает рекурсивную функцию gfCheckNextWhitePointOf():**

```

void gfCheckNextWhitePointOf(int I, int J) //recursive function to define
white point in same area
{
    if (J-1>=0 && gfCheckedList[I][J-1][0]==0) {
        gfCheckedList[I][J-1][0] = 1;
        if (gfSrc.data[gfSTEP*I + J-1]==255) { //if
(gfSrc.data[gfSrc.step*I + (J-1)*gfSrc.channels()]==255) {
            gfCheckedList[I][J-1][1] = gfWhiteAreaIndex;
            gfPixelQuantity[gfWhiteAreaIndex]++;
            gfCheckNextWhitePointOf(I,J-1);
        }
    }
    if (I+1<gfSrc.rows && J-1>=0 && gfCheckedList[I+1][J-1][0]==0) {
        gfCheckedList[I+1][J-1][0] = 1;
        if (gfSrc.data[gfSTEP*(I+1) + J-1]==255) {
            gfCheckedList[I+1][J-1][1] = gfWhiteAreaIndex;
            gfPixelQuantity[gfWhiteAreaIndex]++;
            gfCheckNextWhitePointOf(I+1,J-1);
        }
    }
    if (I+1<gfSrc.rows && gfCheckedList[I+1][J][0]==0) {
        gfCheckedList[I+1][J][0] = 1;
        if (gfSrc.data[gfSTEP*(I+1) + J]==255) {
            gfCheckedList[I+1][J][1] = gfWhiteAreaIndex;
            gfPixelQuantity[gfWhiteAreaIndex]++;
            gfCheckNextWhitePointOf(I+1,J);
        }
    }
}

```

```

if (I+1<gfSrc.rows && J+1<gfSrc.cols && gfCheckedList[I+1][J+1][0]==0) {
    gfCheckedList[I+1][J+1][0] = 1;
    if (gfSrc.data[gfSTEP*(I+1) + J+1]==255) {
        gfCheckedList[I+1][J+1][1] = gfWhiteAreaIndex;
        gfPixelQuantity[gfWhiteAreaIndex]++;
        gfCheckNextWhitePointOf(I+1,J+1);
    }
}
if (J+1<gfSrc.cols && gfCheckedList[I][J+1][0]==0) {
    gfCheckedList[I][J+1][0] = 1;
    if (gfSrc.data[gfSTEP*I + J+1]==255) {
        gfCheckedList[I][J+1][1] = gfWhiteAreaIndex;
        gfPixelQuantity[gfWhiteAreaIndex]++;
        gfCheckNextWhitePointOf(I,J+1);
    }
}
if (I-1>=0 && J+1<gfSrc.cols && gfCheckedList[I-1][J+1][0]==0) {
    gfCheckedList[I-1][J+1][0] = 1;
    if (gfSrc.data[gfSTEP*(I-1) + J+1]==255) {
        gfCheckedList[I-1][J+1][1] = gfWhiteAreaIndex;
        gfPixelQuantity[gfWhiteAreaIndex]++;
        gfCheckNextWhitePointOf(I-1,J+1);
    }
}
if (I-1>=0 && gfCheckedList[I-1][J][0]==0) {
    gfCheckedList[I-1][J][0] = 1;
    if (gfSrc.data[gfSTEP*(I-1) + J]==255) {
        gfCheckedList[I-1][J][1] = gfWhiteAreaIndex;
        gfPixelQuantity[gfWhiteAreaIndex]++;
        gfCheckNextWhitePointOf(I-1,J);
    }
}
if (I-1>=0 && J-1>=0 && gfCheckedList[I-1][J-1][0]==0) {
    gfCheckedList[I-1][J-1][0] = 1;
    if (gfSrc.data[gfSTEP*(I-1) + J-1]==255) {
        gfCheckedList[I-1][J-1][1] = gfWhiteAreaIndex;
        gfPixelQuantity[gfWhiteAreaIndex]++;
        gfCheckNextWhitePointOf(I-1,J-1);
    }
}
}
}

```

### A.3 Скелетизация бинарного изображения тела

Были реализованы пять различных алгоритмов, здесь приведен программный код для алгоритма скелетизации Жанг-Суена.

Алгоритм:

1. **While** points are deleted **do**
2.     **For all** pixels  $p(i,j)$  **do**

3.                   **if** (a)  $2 \leq B(P_1) \leq 6$   
                           (b)  $A(P_1) = 1$   
                           (c) Apply one of the following:  
                               1.  $P_2 \times P_4 \times P_6 = 0$  in odd iterations  
                               2.  $P_2 \times P_4 \times P_8 = 0$  in even iterations  
                           (d) Apply one of the following:  
                               1.  $P_4 \times P_6 \times P_8 = 0$  in odd iterations  
                               2.  $P_2 \times P_6 \times P_8 = 0$  in even iterations  
                           **then**  
 4.                   Delete pixel  $p(i,j)$   
 5.                   **end if**  
 6.                   **end for**  
 7.                   **end while**

Здесь  $A(P_1)$  – число переходов из 0 в 1 по часовой стрелке от точки  $P_0$ ,  
 $B(P_1)$  – количество ненулевых соседей  $P_1$ .

Процесс скелетизации выполняется с помощью функции `defineObjectSkelet()` класса `Rehabilitation` и она использует функцию `thinning()`, коды которой приведены ниже:

```
/**
 * Function for thinning the given binary image
 *
 * @param im Binary image with range = 0-255
 */
void thinning(cv::Mat& im)
{
    im /= 255;

    cv::Mat prev = cv::Mat::zeros(im.size(), CV_8UC1);
    cv::Mat diff;

    do {
        thinningIteration(im, 0);
        thinningIteration(im, 1);
        cv::absdiff(im, prev, diff);
        im.copyTo(prev);
    }
    while (cv::countNonZero(diff) > 0);

    im *= 255;
```

```

}
/**
 * @param im    Binary image with range = 0-1
 * @param iter  0=even, 1=odd
 */
void thinningIteration(cv::Mat& im, int iter)
{
    cv::Mat marker = cv::Mat::zeros(im.size(), CV_8UC1);

    for (int i = 1; i < im.rows-1; i++)
    {
        for (int j = 1; j < im.cols-1; j++)
        {
            uchar p2 = im.at<uchar>(i-1, j);
            uchar p3 = im.at<uchar>(i-1, j+1);
            uchar p4 = im.at<uchar>(i, j+1);
            uchar p5 = im.at<uchar>(i+1, j+1);
            uchar p6 = im.at<uchar>(i+1, j);
            uchar p7 = im.at<uchar>(i+1, j-1);
            uchar p8 = im.at<uchar>(i, j-1);
            uchar p9 = im.at<uchar>(i-1, j-1);

            int A = (p2 == 0 && p3 == 1) + (p3 == 0 && p4 == 1) +
                (p4 == 0 && p5 == 1) + (p5 == 0 && p6 == 1) +
                (p6 == 0 && p7 == 1) + (p7 == 0 && p8 == 1) +
                (p8 == 0 && p9 == 1) + (p9 == 0 && p2 == 1);
            int B = p2 + p3 + p4 + p5 + p6 + p7 + p8 + p9;
            int m1 = iter == 0 ? (p2 * p4 * p6) : (p2 * p4 * p8);
            int m2 = iter == 0 ? (p4 * p6 * p8) : (p2 * p6 * p8);

            if (A == 1 && (B >= 2 && B <= 6) && m1 == 0 && m2 == 0)
                marker.at<uchar>(i,j) = 1;
        }
    }

    im &= ~marker;
}

```

#### А.4 Выделение характерных точек модели

Это процесс выполняется с помощью функции `defineSpecialPoints()` класса `Rehabilitation`. Алгоритм работы этой функции показан на рис. А.3. Программный код:

```

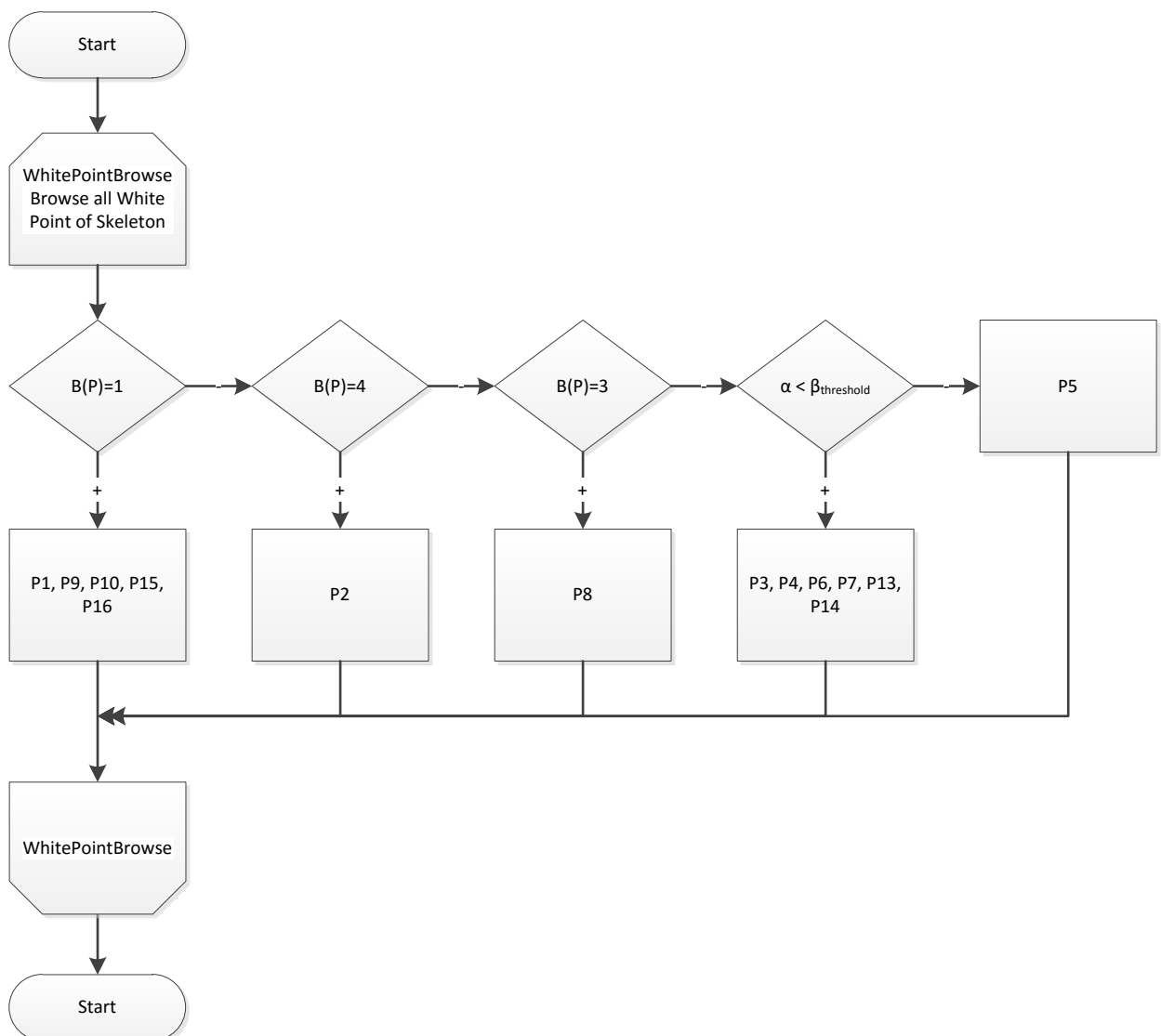
...
for (int i = 0; i < bin_frame.rows; i++) //y
{
    for (int j = 0; j < bin_frame.cols; j++) //x
    {
        if (bin_frame.at<uchar>(i, j) == 255)
        {
            if (B(i,j)==1) {
                p1 = Point(i,j);
                p9 = Point(i,j);
            }
        }
    }
}

```

```

        p10 = Point(i,j);
        p15 = Point(i,j);
        p16 = Point(i,j);
    } else if ((B(i,j)==4) {
        p2 = Point(i,j);
    } else if ((B(i,j)==3) {
        p8 = Point(i,j);
    } else if (alpha(i,j)<beta_threshold) {
        p3 = Point(i,j);
        p4 = Point(i,j);
        p6 = Point(i,j);
        p7 = Point(i,j);
        p13 = Point(i,j);
        p14 = Point(i,j);
    } else {
        p5 = Point(i,j);
    };
}
} ...
}

```



## Рисунок А.3 – Логическая схема алгоритма выделения ХТ

### А.5 Вычисление тангенциальной скорости и тангенциального ускорения

```

int Rehabilitation::kinematic_calculation(int XT_number, int XT_index [], int
XT_position [], int col_number, int row_number)
{
    int x1, y1, x2, y2, x3, y3; //positions of 1 point for 3 continuous
frames
    Point P1, P2, P3;
    Point2d P0, M12, M23;
    double A1, B1, C1, A2, B2, C2;
    double R, a1, a2, b1, b2, fi1, fi2, w1, w2, e, v, a; //ai, bi - vector
phap tuyen
    double Tao = 1/25.0; //(s) because 25 frames per second

    for (int i = 1; i< row_number-1; i++) {
        for (int j = 0; j< XT_number; j++) { //duyet tung diem XT
            P1.x = XT_position[(i-1)*col_number + j*2];
            P1.y = XT_position[(i-1)*col_number + j*2 +1];
            P2.x = XT_position[ i *col_number + j*2];
            P2.y = XT_position[ i *col_number + j*2 +1];
            P3.x = XT_position[(i+1)*col_number + j*2];
            P3.y = XT_position[(i+1)*col_number + j*2 +1];

            M12.x = (P1.x + P2.x) / 2.0;
            M12.y = (P1.y + P2.y) / 2.0;
            M23.x = (P2.x + P3.x) / 2.0;
            M23.y = (P2.y + P3.y) / 2.0;

            A1 = P2.x-P1.x; B1 = P2.y - P1.y; C1 = (P2.x-P1.x)*M12.x +
(P2.y-P1.y)*M12.y;
            A2 = P3.x-P2.x; B2 = P3.y - P1.y; C2 = (P3.x-P2.x)*M23.x +
(P3.y-P2.y)*M23.y;

            if (A1*B2-A2*B1 != 0) {
                P0.x = (C1*B2-C2*B1)/(A1*B2-A2*B1);
                P0.y = (A1*C2-A2*C1)/(A1*B2-A2*B1);

                R = sqrt((P0.x-P1.x)*(P0.x-P1.x) + (P0.y-P1.y)*(P0.y-
P1.y));

                a1 = P1.y - P0.y;
                b1 = - (P1.x - P0.x);
                a2 = P2.y - P0.y;
                b2 = - (P2.x - P0.x);
                fi1 = acos( abs(a1*a2 + b1*b2) / ( sqrt(a1*a1 + b1*b1)
* sqrt(a2*a2 + b2*b2) ) );

                a1 = P2.y - P0.y;

```

```

        b1 = - (P2.x - P0.x);
        a2 = P3.y - P0.y;
        b2 = - (P3.x - P0.x);
        fi2 = acos( abs(a1*a2 + b1*b2) / ( sqrt(a1*a1 + b1*b1)
* sqrt(a2*a2 + b2*b2) ) );

        w1 = fi1 / Tao;
        w2 = fi2 / Tao;

        e = ( w2 - w1 ) / Tao;

        v = w1*R;
        a = e * R;

    } else {
        P0.x = DBL_MAX;
        P0.y = DBL_MAX;

        double s1 = sqrt( (P2.x-P1.x)*(P2.x-P1.x) + (P2.y -
P1.y)*(P2.y-P1.y) );
        double s2 = sqrt( (P3.x-P2.x)*(P3.x-P2.x) + (P3.y -
P2.y)*(P3.y-P2.y) );
        v = s1 / Tao;
        a = (s2-s1) / (Tao*Tao);

    }
    }
    cout << endl;
}

return 0;
}

```

## A.6 Регистрация обучающего и повторенного движений веб-камерой и отправка данных на сервер через Socket веб-браузера

Коды написаны на разных языках для отображения веб-контента и работы с веб-Socket протоколом.

### HTML, PHP коды для отображения веб-контента:

```

<?php
/**
 * Template Name: Create Exercise
 */
get_header(); ?>

<div id="primary" class="content-area">
    <main id="main" class="site-main" role="main">

```



```

<article id="post-54" class="post-54 page type-page status-
publish hentry">
    <header class="entry-header">
        <h1 class="entry-title">Process</h1>
    </header><!-- .entry-header -->
    <div class="entry-content">
        <div>
            <ul class="nav nav-tabs">
                <li class="active"><a
id="create_instruction_tab" href="#create_instruction" data-
toggle="tab">Instruction</a></li>
                <li><a id="create_video_tab"
href="#create_video" data-toggle="tab">Capture Video</a></li>
            </ul>
            <!-- Tab panes -->
            <div class="tab-content">
<div id="create_instruction" class="tab-pane active">
    <br/>

    <!-- New Post Form -->

    <div id="postbox">

    <form id="new_post" name="new_post" method="post" action="">

    <p><label for="title">Title</label><br />

    <input type="text" id="title" value="" tabindex="1" size="20"
name="title" />

    </p>

    <p><label for="description">Description</label><br />

    <textarea id="description" tabindex="3" name="description" cols="50"
rows="6"></textarea>

    </p>

    <p><?php
wp_dropdown_categories(array(
    'hide_empty' => 0,
    'hierarchical' => 1,
    'tab_index' => 4,
    'taxonomy' => 'category',
    'child_of' => 6 //"Exercise" category
));
?></p>

    <p><label for="post_tags">Tags</label>

    <input type="text" value="" tabindex="5" size="16" name="post_tags"
id="post_tags" /></p>

```

```

    <p align="right"><input type="submit" value="Publish" tabindex="6"
id="submit" name="submit" /></p>

    <input type="hidden" name="post_type" id="post_type" value="post" />

    <input type="hidden" name="action" value="post" />

    <?php wp_nonce_field( 'new-post' ); ?>

</form>

</div>

<!--// New Post Form -->
</div>
<div id="create_video" class="tab-pane">
    <br/><center>
        <button id="show_camera_button" class="btn btn-primary"
type="button">Show camera</button> <button id="start_video_process_button"
class="btn btn-primary" type="button">Start Process</button> <button
id="stop_video_process_button" class="btn btn-primary" type="button">Stop
process</button>
    </center>
    <br/>
    <center>
        <video id="camera" autoplay="autoplay" width="640"></video>
        <canvas id="canvas" width="640" height="480"></canvas>
    </center>
</div>

        </div>
    </div>
</div><!-- .entry-content -->

        <footer class="entry-footer"><span class="edit-
link"><a class="post-edit-link" href="http://192.168.0.64/wp-
admin/post.php?post=54&action=edit">Edit</a></span></footer><!-- .entry-
footer -->

    </article><!-- #post-## -->

    <?php // If comments are open or we have at least one
comment, load up the comment template.
    if ( comments_open() || get_comments_number() ) :
        comments_template();
    endif; ?>

    </main><!-- .site-main -->
</div><!-- .content-area -->

<?php get_footer(); ?>

```

**Javascript коды для отправки данных на сервер через веб-socket протокол:**

```
function dataURIToBlob(dataURI) {
```

```

    // convert base64/URLEncoded data component to raw binary data held in a
string
    var byteString;
    if (dataURI.split(',')[0].indexOf('base64') >= 0)
        byteString = atob(dataURI.split(',')[1]);
    else
        byteString = unescape(dataURI.split(',')[1]);

    // separate out the mime component
    var mimeString = dataURI.split(',')[0].split(':')[1].split(';')[0];

    // write the bytes of the string to a typed array
    var ia = new Uint8Array(byteString.length);
    for (var i = 0; i < byteString.length; i++) {
        ia[i] = byteString.charCodeAt(i);
    }

    return new Blob([ia], {type:mimeString});
}

// Get a reference to the video element on the page.
var vid = document.getElementById('camera');

//canvas processing
var canvas = document.getElementById('canvas');
var ctx = canvas.getContext('2d');

//isProcessing
var timer;
var ws = null;

var i;

function deleteWs() {
    if (null!=ws) {
        ws.close();
        ws.onopen = null;
        ws.onerror = null;
        ws.onclose = null;
        ws.onmessage = null;
        ws = null;
    }
}

function showCameraBtnClickFunction( jQuery ) {
    //show recomend_box
    //$("#recomend_box").html("Recomendation will go here...");

    // Normalize the various vendor prefixed versions of getUserMedia.
    navigator.getUserMedia = (navigator.getUserMedia ||
        navigator.webkitGetUserMedia ||
        navigator.mozGetUserMedia ||
        navigator.msGetUserMedia);

    // Check that the browser supports getUserMedia.
    // If it doesn't show an alert, otherwise continue.

```

```

if (navigator.getUserMedia) {
    // Request the camera.
    navigator.getUserMedia(
        // Constraints
        {
            video: true
        },
        // Success Callback
        function(localMediaStream) {
            // Create an object URL for the video stream and use
this
            // to set the video source.
            vid.src =
window.URL.createObjectURL(localMediaStream);
        },
        // Error Callback
        function(err) {
            // Log the error to the console.
            console.log('The following error occurred when trying to
use getUserMedia: ' + err);
        }
    );
} else {
    alert('Sorry, your browser does not support getUserMedia');
}

function sendData() {
    console.log(i++);
    //send data through this socket connection
    if (null!= ws && ws.readyState == WebSocket.OPEN) { //Connection is
ready for sending data
        console.log("sent....")

        //write frame into canvas (for grabbing data later)
        ctx.drawImage(vid, 0, 0, 640, 480);

        //grab frame data from canvas //binary base64
        var data = canvas.toDataURL('image/jpeg'); //empty - png by
default, 'image/bmp'

        newblob = dataURIToBlob(data);

        //send data to server
        if (ws.bufferedAmount == 0) {
            //ws.send("a");
            ws.send(data); console.log(data);
        } else {
            console.log("Connection speed is slow!!!");
        }
    }
}
}

```

```

function startVideoProcessFunction( jQuery ) {
    //declare websocket
    ws = new WebSocket("ws://192.168.0.64:7777");
    ws.onopen = function(evt) {
        console.log("Connection open...");

        i = 0;
        console.log("start.....");

        sendData();

        console.log("end.....");
    };
    ws.onmessage = function(evt) {
        console.log("Received message: " + evt.data);
    };
    ws.onclose = function(evt) {
        deleteWs();
        console.log("Connection closed.");
    };
    ws.onerror = function(evt) {
        deleteWs();
        console.log("Connection broken! Error: " + evt.data);
    }
}

function stopVideoProcessFunction( jQuery ) {
    //hide recomend_box
    //$("#recomend_box").empty();

    clearInterval(timer);
    deleteWs();
    console.log("timer is cleared.....");
}

$(document).ready(function(){
    //just show camera, nothing happen
    $("#show_camera_button").click( showCameraBtnClickFunction );

    //start sending data to server to process. After recording, trainer
    should choose the right period
    $("#start_video_process_button").click( startVideoProcessFunction );

    //stop sending data to server, then trainer can choose the right period
    or start over again
    $("#stop_video_process_button").click( stopVideoProcessFunction );
});

```

## ПРИЛОЖЕНИЕ Б

Акты внедрения результатов диссертационной работы



VIETLOGIC INFORMATION & COMMUNICATION  
TECHNOLOGY COMPANY LIMITED  
Address: 57A, 23 Tran Phu street, Hanoi, Vietnam  
Tel./Fax: +84 437 522 534  
Email: [contact@vietlogic.com](mailto:contact@vietlogic.com)  
Website: [www.vietlogic.com](http://www.vietlogic.com)

APPROVED by

Vice-president of Vietlogic Information &  
Communication Technology Co. Ltd



Le Thi Hoa

May 20<sup>th</sup>, 2015

### ACT OF IMPLEMENTATION

of Nguyen Huy Cuong's PhD thesis results

This act confirms that the Decision Support System for Remote Motion Rehabilitation, developed by Nguyen Huy Cuong as the result of his PhD study at Odessa National Politechnic University, was successfully implemented in our company "VIETLOGIC".

The system developed by Nguyen Huy Cuong was successfully approbated and used in the company's technical seminars and conferences with the participation of doctor-expert. The technology is based on quantitative comparison of movement kinematic parameters getting from webcams in the form of a video stream.

Despite the shortcomings of the system, which can be solved in the future, by using the system's algorithm, we can develop many different products in different areas for serving people, such as remote learning, distant sport training, computer gaming etc.

At present, our company is cooperating with the author on the development of a commercial product, based on idea and technology proposed by Nguyen Huy Cuong in his thesis.

Head of Technical Department

Le Van Hung

УТВЕРЖДАЮ

Вице-Президент Vietlogic Информационные &  
Коммуникационные Технологии ООО



Ле Тхи Хоа

« 20 » мая 2015

### АКТ ВНЕДРЕНИЯ

Результатов кандидатской диссертационной работы

**Нгуен Гуи Кионг**

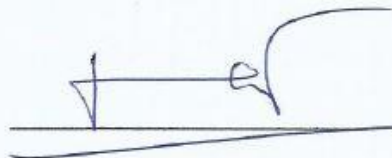
Настоящим актом внедрения подтверждается, что система поддержки принятия решений при дистанционной двигательной реабилитации, созданная Нгуен Гуи Кионгом в результате работы над кандидатской диссертацией в Одесском Национальном Политехническом Университете, была успешно внедрена в нашей компании «VIETLOGIC».

Система, разработанная Нгуен Гуи Кионгом, успешно апробирована и рассмотрена на технических семинарах и конференциях нашей компании с участием врача-эксперта. Технология системы основана на количественном сравнении кинематических параметров движения, поступающих с веб-камер в виде видеопотока.

Несмотря на отдельные недостатки системы, которые можно решить в будущем, при применении этой системы можно развивать различные продукты в сферах обслуживания человека, например: удаленное обучение, дистанционная спортивная тренировка, компьютерные игры и пр.

В настоящее время, наша компания взаимодействует с автором по вопросу разработки коммерческого продукта, в основу которого взяты идея и технология, предложенные в диссертации Нгуен Гуи Кионгом.

Начальник технического департамента



Ле Ван Хунг



«ЗАТВЕРДЖУЮ»

Проректор з науково-педагогічної та виховної роботи Одеського національного політехнічного університету

«» Нестеренко С.А.  
2015 р.



АКТ

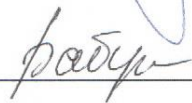
про впровадження в навчальний процес результатів дисертаційної роботи за темою «Моделі і методи комп'ютерної підтримки прийняття рішень в автоматизованих системах рухової реабілітації» на здобуття наукового ступеня кандидата технічних наук Нгуена Гуї Кіонга

Комісія у складі професора кафедри інформаційних систем д.т.н. Крилова В.М., заступника завідувача кафедри інформаційних систем доцента к.т.н. Бабілонги О. Ю., заступника завідувача кафедри інформаційних систем доцента к.т.н. Ніколенко А. О. склали цей акт про те, що у Одеському національному політехнічному університеті, при викладанні дисциплін «Теорія прийняття рішень» та «Інтелектуальний аналіз даних», а також при виконанні курсових і дипломних робіт для студентів напряму підготовки 6.050101 «Комп'ютерні науки» та спеціальності 7.05010101 «Інформаційні управляючі системи та технології», використовувались наступні оригінальні матеріали, які отримано у дисертаційній роботі Нгуена Г.К. : моделі руху тіла людини у відеопотоці, метод кількісної оцінки подібності рухів людини у відеопотоці, програмні засоби для аналізу рухів у відеопотоці.


Професор кафедри ІС  
професор, д. т. н.

  
В.М. Крилов

Заступник завідувача кафедри ІС  
доцент, к. т. н.

  
О. Ю. Бабілонга

Заступник завідувача кафедри ІС  
доцент, к. т. н.

  
А. О. Ніколенко