

**Министерство образования и науки Российской Федерации**  
федеральное государственное автономное образовательное учреждение  
высшего образования  
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

---

Школа информационных технологий и робототехники  
Направление подготовки 27.04.04 «Управление в технических системах»  
Отделение школы (НОЦ) автоматизации и робототехники

**МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ**

Тема работы
<b>Стратегия ведения игры роботом с применением системы технического зрения</b>

УДК 621.865.8-2:793

Студент

Группа	ФИО	Подпись	Дата
8АМ61	Липатов Дмитрий Сергеевич		

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Руководитель ВКР	Замятин С.В.	Доцент, к.т.н		
Руководитель ООП	Пушкарев М.И.	Доцент, к.т.н		

**КОНСУЛЬТАНТЫ:**

По разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОСГН	Баннова К.А.	К.Э.Н		

По разделу «Социальная ответственность»

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ассистент ОКИД	Авдеева И.И.			

**ДОПУСТИТЬ К ЗАЩИТЕ:**

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Руководитель ОАР	Леонов С.В.	К.Т.Н		

Томск – 2018 г.

**Запланированные результаты обучения**  
по направлению 27.04.04 «Управление в технических системах»

<b>Код результатов</b>	<b>Результаты обучения (выпускник должен быть готов)</b> <b>Профессиональные и общепрофессиональные компетенции</b>
P1	Применять глубокие естественнонаучные и математические знания для решения научных и инженерных задач в области анализа, синтеза, проектирования, производства и эксплуатации средств автоматизации и систем управления техническими объектами.
P2	Уметь обрабатывать, анализировать и обобщать научно-техническую информацию, передовой отечественный и зарубежный опыт в области теории, проектирования, производства и эксплуатации средств автоматизации и систем управления техническими объектами
P3	Ставить и решать инновационные задачи инженерного анализа, связанные с разработкой технических систем управления с использованием аналитических методов и сложных моделей.
P4	Выполнять инновационные инженерные проекты по разработке программно-аппаратных средств автоматизированных систем различного назначения с использованием современных методов проектирования, систем автоматизированного проектирования, передового опыта разработки конкурентно способных изделий.
P5	Планировать и проводить теоретические и экспериментальные исследования в области проектирования аппаратных и программных средств автоматизированных систем с использованием новейших достижений науки и техники, передового отечественного и зарубежного опыта. Критически оценивать полученные данные и делать выводы.
P6	Осуществлять авторское сопровождение процессов проектирования, внедрения и эксплуатации программно-аппаратных средств автоматизированных систем различного назначения.
P7	Владеть иностранным языком на уровне, позволяющем работать в интернациональной профессиональной среде с пониманием культурных, языковых и социально-экономических различий партнеров
P8	Осуществлять коммуникации в профессиональной среде и в обществе в целом, активно владеть иностранным языком, разрабатывать документацию, презентовать и защищать результаты инновационной инженерной деятельности, в том числе на иностранном языке.
P9	Эффективно работать индивидуально и в качестве члена и руководителя группы, в том числе междисциплинарной и международной, при решении инновационных инженерных задач.
P10	Демонстрировать личную ответственность и ответственность за работу возглавляемого коллектива, приверженность и готовность следовать профессиональной этике и нормам ведения инновационной инженерной деятельности. Демонстрировать глубокие знания правовых, социальных, экологических и культурных аспектов инновационной инженерной деятельности.
P11	Демонстрировать способность к самостоятельному обучению, непрерывному самосовершенствованию в инженерной деятельности, способность к педагогической деятельности.

**Министерство образования и науки Российской Федерации**  
федеральное государственное автономное образовательное учреждение  
высшего образования  
**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

---

Школа информационных технологий и робототехники  
Отделение школы (НОЦ) автоматизации и робототехники  
Направление подготовки 27.04.04 «Управление в технических системах»

УТВЕРЖДАЮ:  
Руководитель ООП  
\_\_\_\_\_ Пушкарёв М.И.  
(Подпись) (Дата)

**ЗАДАНИЕ**  
**на выполнение выпускной квалификационной работы**

В форме:

Магистерской диссертации
--------------------------

(бакалаврской работы, дипломного проекта/работы, магистерской диссертации)

Студенту:

Группа	ФИО
8АМ61	Липатову Дмитрию Сергеевичу

Тема работы:

Стратегия ведения игры роботом с применением системы технического зрения	
Утверждена приказом директора (дата, номер)	N 2181/с от 28.03.2018 г.

Срок сдачи студентом выполненной работы:	16.06. 2018 г.
--	----------------

**ТЕХНИЧЕСКОЕ ЗАДАНИЕ:**

<p><b>Исходные данные к работе</b> <i>(наименование объекта исследования или проектирования; производительность или нагрузка; режим работы (непрерывный, периодический, циклический и т. д.); вид сырья или материал изделия; требования к продукту, изделию или процессу; особые требования к особенностям функционирования (эксплуатации) объекта или изделия в плане безопасности эксплуатации, влияния на окружающую среду, энергозатратам; экономический анализ и т. д.).</i></p>	Объектом исследования является алгоритмы выигрышной стратегии ведения игры и технического зрения, применяемые для игры роботом в пасьянс «Паук»
--	---

<p><b>Перечень подлежащих исследованию, проектированию и разработке вопросов</b></p> <p><i>(аналитический обзор по литературным источникам с целью выяснения достижений мировой науки техники в рассматриваемой области; постановка задачи исследования, проектирования, конструирования; содержание процедуры исследования, проектирования, конструирования; обсуждение результатов выполненной работы; наименование дополнительных разделов, подлежащих разработке; заключение по работе).</i></p>	<p>Обзор методов и выбор оптимального решения задачи стратегии игры. Обзор и оценка методов распознавания карт и выбор оптимального. Разработка программного обеспечения робота. Разработка эмулятор клавиатуры. Оценка эффективности алгоритма.</p>
<p><b>Перечень графического материала</b></p> <p><i>(с точным указанием обязательных чертежей)</i></p>	<p>Блок-схема алгоритма стратегии игры Блок-схема алгоритма распознавания карт</p>

<p><b>Консультанты по разделам выпускной квалификационной работы</b></p> <p><i>(с указанием разделов)</i></p>	
<p><b>Раздел</b></p>	<p><b>Консультант</b></p>
<p>Финансовый менеджмент, ресурсоэффективность и ресурсосбережение</p>	<p>Баннова Кристина Алексеевна</p>
<p>Социальная ответственность</p>	<p>Авдеева Ирина Ивановна</p>
<p>Раздел на иностранном языке</p>	<p>Шепетовский Денис Владимирович</p>
<p><b>Названия разделов, которые должны быть написаны на русском и иностранном языках:</b></p>	
<p>Введение (Introduction)</p>	
<p>Обзор литературы и научных источников (Review of literature and scientific articles)</p>	
<p>Выбор методов и технических решений (Selection of methods and technical solutions)</p>	

<p><b>Дата выдачи задания на выполнение выпускной квалификационной работы по линейному графику</b></p>	<p>01.03.18</p>
--	-----------------

**Задание выдал руководитель:**

<p>Должность</p>	<p>ФИО</p>	<p>Ученая степень, звание</p>	<p>Подпись</p>	<p>Дата</p>
<p>Доцент ОАИР</p>	<p>Замятин С.В.</p>	<p>к.т.н</p>		

**Задание принял к исполнению студент:**

<p>Группа</p>	<p>ФИО</p>	<p>Подпись</p>	<p>Дата</p>
<p>8АМ61</p>	<p>Липатов Дмитрий Сергеевич</p>		

**ЗАДАНИЕ ДЛЯ РАЗДЕЛА  
«ФИНАНСОВЫЙ МЕНЕДЖМЕНТ, РЕСУРСОЭФФЕКТИВНОСТЬ И  
РЕСУРСОСБЕРЕЖЕНИЕ»**

Студенту:

<b>Группа</b> 8АМ61	<b>ФИО</b> Липатову Дмитрию Сергеевичу
------------------------	---

<b>Инженерная школа</b> Уровень образования	<b>ИТиР</b> Магистратура	<b>Отделение</b> Направление/специальность	<b>АиР</b> Управление в технических системах
--	-----------------------------	---	---

**Исходные данные к разделу «Финансовый менеджмент, ресурсоэффективность и ресурсосбережение»:**

1. Стоимость ресурсов научного исследования (НИ): материально-технических, энергетических, финансовых, информационных и человеческих	Затраты на выполнение НИР включают в себя затраты на сырье, материалы, комплектующие изделия, специальное оборудование для научных (экспериментальных) работ, основную и дополнительную заработную платы исполнителей, отчисления на социальные нужды, накладные расходы
2. Нормы и нормативы расходования ресурсов	
3. Используемая система налогообложения, ставки налогов, отчислений, дисконтирования и кредитования	НИР выполнялась в соответствии со стандартной системой налогообложения, отчислений, кредитования

**Перечень вопросов, подлежащих исследованию, проектированию и разработке:**

1. Оценка коммерческого и инновационного потенциала НТИ	Определение потенциальных потребителей результатов исследования и анализ конкурентных технических решений с позиции ресурсоэффективности и ресурсосбережения, оценка готовности проекта к коммерциализации. Проведения FAST анализа и SWOT-анализ. Диаграмма Исикавы.
2. Разработка устава научно-технического проекта	Информация о заинтересованных сторонах проекта, цели и ожидаемые результаты НИР, трудозатраты и функции исполнителей проекта
3. Планирование процесса управления НТИ: структура и график проведения, бюджет, риски и организация закупок	Составление перечня этапов и работ по выполнению НИР, составление калькуляции по отдельным статьям затрат всех видов необходимых ресурсов. Бюджетирование и организация структуры предприятия.
4. Определение ресурсной, финансовой, экономической эффективности	Определение эффективности и перспектив научного исследования.

**Перечень графического материала (с точным указанием обязательных чертежей):**

<ol style="list-style-type: none"> <li>1. «Портрет» потребителя результатов НТИ</li> <li>2. Сегментирование рынка</li> <li>3. Оценка конкурентоспособности технических решений</li> <li>4. Диаграмма FAST</li> <li>5. Матрица SWOT</li> <li>6. График проведения и бюджет НТИ</li> <li>7. Оценка ресурсной, финансовой и экономической эффективности НТИ</li> <li>8. Потенциальные риски</li> </ol>
---

<b>Дата выдачи задания для раздела по линейному графику</b>	
---	--

**Задание выдал консультант:**

<b>Должность</b> Доцент ОСГН	<b>ФИО</b> Баннова К.А.	<b>Ученая степень, звание</b> К.Э.Н.	<b>Подпись</b>	<b>Дата</b> 01.03.2018
---------------------------------	----------------------------	---	----------------	---------------------------

**Задание принял к исполнению студент:**

<b>Группа</b> 8АМ61	<b>ФИО</b> Липатов Дмитрий Сергеевич	<b>Подпись</b>	<b>Дата</b> 01.03.2018
------------------------	---	----------------	---------------------------

## ЗАДАНИЕ ДЛЯ РАЗДЕЛА «СОЦИАЛЬНАЯ ОТВЕТСТВЕННОСТЬ»

Студенту:

Группа	ФИО
8АМ61	Липатову Дмитрию Сергеевичу

Инженерная школа	ИТиР	Отделение	АиР
Уровень образования	Магистратура	Направление/специальность	27.04.04 Управление в технических системах

### Исходные данные к разделу «Социальная ответственность»:

1. Характеристика объекта исследования (вещество, материал, прибор, алгоритм, методика, рабочая зона) и области применения	Целью магистерской диссертации является разработка стратегии ведение игры и создание на основе робота с применением технического зрения. Область применения – образовательная и научная деятельность. Рабочее помещение - офисное место (стол, стул, персональный компьютер)
--	--

### Перечень вопросов, подлежащих исследованию, проектированию и разработке:

<p><b>1. Производственная безопасность</b></p> <p>1.1. Анализ выявленных вредных факторов при разработке и эксплуатации проектируемого решения.</p> <p>1.2. Анализ выявленных опасных факторов при разработке и эксплуатации проектируемого решения.</p> <p>1.3. Рекомендации по минимизации влияний на работника.</p>	<p>1.1. В качестве вредных факторов согласно ГОСТ 12.0.003-74 выделены: показатели микроклимата и освещения, шум, вибрация, электромагнитное излучение, психофизические факторы (нервно психологические, статические, умственные, монотонность).</p> <p>1.2. В качестве опасных факторов выделены: опасность поражения электрическим током, короткое замыкание, статическое электричество.</p> <p>1.3. Приведены рекомендации по улучшению освещения на рабочем месте согласно СанПиН 2.2.4.3359-16</p>
<p><b>2. Экологическая безопасность:</b></p> <p>2.1. Анализ воздействия на окружающую среду</p> <p>2.2. Рекомендации по минимизации влияния на окружающую среду.</p>	<p>2.1. Деятельность по разработке ПО не связана с производством, поэтому влияние на окружающую среду минимально.</p> <p>2.2. Рассмотрена утилизация комплектующих ПК, люминесцентных ламп.</p>
<p><b>3. Защита в чрезвычайных ситуациях:</b></p> <p>3.1. Перечень возможных ЧС на объекте.</p> <p>3.2. Меры по предотвращению и ликвидации ЧС и их последствий</p>	<p>3.1. Вероятные ЧС возникающие на рабочем месте пожары и взрывы.</p> <p>3.2. Мероприятия по предотвращению наиболее типичной ЧС – пожара, согласно нормативным документам:</p> <ul style="list-style-type: none"> <li>- НПБ 105-03;</li> <li>- ППБ 01-03.</li> </ul>
<p><b>4. Правовые и организационные вопросы обеспечения безопасности:</b></p> <ul style="list-style-type: none"> <li>- специальные (характерные для проектируемой рабочей зоны) правовые нормы трудового законодательства;</li> <li>- организационные мероприятия при компоновке рабочей зоны</li> </ul>	<p>4.1 Охрана труда для операторов и пользователей ПК. Требования безопасности во время работы. Соответствие рабочего места по требованиям и правила и нормы поведения на производстве согласно НТД (ГОСТ 12.2.032-78)</p> <p>4.2 Организационные мероприятия при компоновке рабочей зоны. Соблюдение "Трудового кодекса Российской Федерации" от 30.12.2001 N 197-ФЗ (ред. от 05.02.2018)</p>

Дата выдачи задания для раздела по линейному графику	01.03.18
--	----------

### Задание выдал консультант:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Ассистент	Авдеева Ирина Ивановна			01.03.18

### Задание принял к исполнению студент:

Группа	ФИО	Подпись	Дата
8АМ61	Липатов Дмитрий Сергеевич		01.03.18

## РЕФЕРАТ

Выпускная квалификационная работа 133 с., 30 рис., 39 табл., 50 источников, 10 прил.

Ключевые слова: \_\_\_\_\_ ПАСЬЯНС, NP-ПОЛНАЯ ЗАДАЧА, ТЕХНИЧЕСКОЕ ЗРЕНИЕ, РАСПОЗНОВАНИЕ КАРТ, КОНТУРНЫЙ АНАЛИЗ, КЛЮЧЕВЫЕ ТОЧКИ, KAZE, НID-УСТРОЙСТВО.

Объектом исследования является задача по разработке стратегии ведения игры “Пасьянс паук”. Цель работы – разработка стратегии ведения игры и на её основе создание робота с применением технического зрения. В процессе исследования проводились поиски методов для стратегии ведения игры и распознавания карт на игровом поле.

В результате исследования разработаны алгоритмы ведения игры “Пасьянс паук” и распознавания карт на игровом поле, на основе которых создан робот на языке программирования C++.

Основные конструктивные, технологические и технико-эксплуатационные характеристики: робот состоит из камеры (logitech c270) ЭВМ (ноутбук) и эмулятор устройства ввода (Arduino Pro Micro 2шт.). При разработке использовалось ПО: ОС Lubuntu 16.04, IDE Code Blocks 13.12 и Arduino IDE, Библиотека OpenCV 3.2.

Степень внедрения: разработан и спроектирован прототип.

Область применения: тестирование программного обеспечения, обучающие курсы по алгоритмам распознавания и методам машинного обучения. Экономическая эффективность при разработке дает использование свободно распространяемого программного обеспечения.

В будущем планируется портирование на портативные устройства (raspberry pi, Android устройства) и оптимизация ПО.

# **ОБОЗНАЧЕНИЯ, СОКРАЩЕНИЯ, НОРМАТИВНЫЕ ССЫЛКИ**

## **Нормативные ссылки**

ГОСТ 19.701-90 Единая система программной документации схемы алгоритмов, программ, данных и систем. Обозначения условные и правила выполнения.

ГОСТ 2.702-2011 Единая система конструкторской документации (ЕСКД). Правила выполнения электрических схем

ГОСТ 12.0.003-74 Система стандартов безопасности труда (ССБТ). Опасные и вредные производственные факторы.

ГОСТ 12.1.003-83 Система стандартов безопасности труда (ССБТ). Шум. Общие требования безопасности.

ГОСТ 12.1.030-81 Система стандартов безопасности труда (ССБТ). Электробезопасность. Защитное заземление. Зануление.

ГОСТ 12.1.004-91 Система стандартов безопасности труда (ССБТ). Пожарная безопасность. Общие требования.

ГОСТ 12.2.032-78 Система стандартов безопасности труда (ССБТ). Рабочее место при выполнении работ сидя. Общие эргономические требования.

ГОСТ 12.2.033-78 Система стандартов безопасности труда (ССБТ). Рабочее место при выполнении работ стоя. Общие эргономические требования.

## **Обозначения и сокращения**

IDE – Integrated Development Environment

NP – non-deterministic polynomial

CV – Computer Vision

OpenCV – Open Computer Vision

GPU – graphics processing unit

CUDA – Compute Unified Device Architecture

SIFT – Scale-invariant feature transform

HID – Human interface device

## Оглавление

Введение.....	14
1. Обзор литературы и научных источников.....	16
1.1 Обзор исследований о пасьянс «Паук».....	16
1.1.1 История возникновения пасьянса «Паук».....	16
1.1.2 Правила игры пасьянса «Паук».....	16
1.1.3 Исследование игры пасьянса «Паук».....	17
1.2 Техническое зрение.....	18
1.2.1 История компьютерного зрения.....	20
1.2.2 Литературные источники по компьютерному зрению.....	21
1.2.3 Интернет источники по компьютерному зрению.....	23
1.2.4 Материалы от вузов по компьютерному зрению.....	23
1.2.5 Исследования распознавания игровых карт.....	24
2. Выбор методов и технических решений.....	25
2.1 Выбор методов реализации выигрышной стратегии.....	25
2.1.1 Постановка задачи.....	25
2.1.2 Описание методов решения задачи.....	26
2.1.3 Сравнение методов и выбор оптимального.....	29
2.2 Выбор методов распознавания на изображения.....	31
2.2.1 Постановка задачи распознавания на изображения.....	31
2.2.2 Описание методов распознавания на изображения.....	32
2.2.3 Сравнение методов распознавания на изображения.....	35
2.3 Выбор технологий эмуляции устройства ввода.....	36
3. Реализация алгоритма и проектирование робота.....	38
3.1 Выбор программного обеспечения для реализации алгоритмов.....	38
3.2 Реализация алгоритма стратегии ведения игры.....	40
3.2.1 Настройка среды разработки.....	40
3.2.2 Разработка библиотек классов.....	42

3.2.3	Реализация алгоритма.....	43
3.3	Реализация алгоритма распознавания карт на изображении.....	44
3.3.1	Установка библиотеки OpenCV.....	44
3.3.2	Выбор камеры для проекта .....	46
3.3.3	Обработка изображений и контурный анализ .....	47
3.3.4	Распознавание карт по ключевым точкам и реализация алгоритма... ..	51
3.4	Разработка эмулятор клавиатуры .....	52
3.4.1	Выбор аппаратной и программной платформы .....	52
3.4.2	Реализация аппаратной составляющей.....	53
3.4.2	Реализация программной составляющей .....	55
3.5	Тестирование и оценка результатов.....	56
4	Финансовый менеджмент, ресурсоэффективность и ресурсосбережение.....	58
4.1	Предпроектный анализ .....	58
4.1.1	Потенциальные потребители результатов исследования .....	58
4.1.2	Анализ конкурентных технических решений с позиции ресурсоэффективности и ресурсосбережения.....	58
4.1.3	FAST-анализ .....	60
4.1.4	Диаграмма Исикава.....	63
4.1.5	SWOT-анализ.....	64
4.1.6	Оценка готовности проекта к коммерциализации.....	65
4.2	Инициация проекта .....	67
4.3	Планирование управления научно-техническим проектом.....	68
4.3.1	Иерархическая структура работ проекта .....	68
4.3.2	Контрольные события проекта .....	69
4.3.3	План проекта.....	69
4.3.4	Бюджет научного исследования .....	71
4.3.4.1	Расчет затрат на сырье, материалы, покупные изделия и полуфабрикаты .....	71
4.3.4.2	Расчет затрат на специальное оборудование для научных работ .....	72

4.3.4.3	Основная заработная плата исполнителей темы .....	73
4.3.4.4	Отчисления во внебюджетные фонды (страховые отчисления)..	73
4.3.4.5	Накладные расходы .....	74
4.3.4.6	Формирование бюджета затрат научно-исследовательского проекта.....	75
4.3.5	Организационная структура проекта .....	75
4.4	Определение ресурсной, финансовой, бюджетной, социальной и экономической эффективности исследования .....	76
4.4.1.	Оценка научно-технического уровня НИР.....	76
5.	Социальная ответственность .....	79
	Введение.....	79
5.1	Производственная безопасность .....	79
5.1.1	Анализ выявленных вредных факторов при разработке и эксплуатации проектируемого решения.....	79
5.1.1.1	Освещение.....	80
5.1.1.2	Вибрация .....	81
5.1.1.3	Микроклимат .....	82
5.1.1.4	Шум .....	83
5.1.1.5	Электромагнитные излучения .....	83
5.1.1.6.	Психофизиологические факторы .....	84
5.1.2.	Анализ выявленных опасных факторов при разработке и эксплуатации проектируемого решения.....	84
5.1.2.1	Электрический ток .....	85
5.1.2.2	Короткое замыкание .....	86
5.1.2.3	Статическое электричество.....	86

5.1.3. Рекомендации по минимизации влияний на работника на примере освещения .....	87
5.2 Экологическая безопасность.....	89
5.2.1 Утилизация комплектующих ПК .....	90
5.2.2 Утилизация люминесцентных ламп.....	90
5.3 Защита в чрезвычайных ситуациях .....	91
5.3.1 Перечень возможных ЧС на объекте .....	91
5.3.2 Меры по предотвращению и ликвидации ЧС и их последствий .....	92
5.4 Правовые и организационные вопросы обеспечения безопасности .....	93
5.4.1. Специальные правовые нормы трудового законодательства.....	93
5.4.2. Организационные мероприятия при компоновке рабочей зоны .....	94
Заключение .....	95
Заключение .....	96
Список использованных источников .....	97
Приложение А .....	102
Приложение Б.....	116
Приложение В.....	119
Приложение Г .....	122
Приложение Д.....	123
Приложение Е.....	124
Приложение Ж.....	126
Приложение З .....	127
Приложение И .....	131
Приложение К .....	133

## Введение

Современные роботы проникли в очень многие сферы человеческой жизни. И во многом они превосходят или находятся наравне с человеком. Так в игровой сфере роботы отметились недавними победами в Го и Dota 2[1].

В данной работе рассматривается вариант разработки алгоритма и реализация на его основе робота, который играет пасьянс «Паук».

Было проведено тестирование алгоритма, сравнение количества выигранных партий и затраченного времени на партию при разных количествах мастей со статистикой игры людей в пасьянс.

Далее был спроектирован и создан робот, который имитирует человека с точки зрения компьютера и игры. Для этого была использована технология технического зрения, которая позволяет производить обнаружение, отслеживание и классификацию объектов[2], как бы имитируя зрение человека. Для устройства ввода был создан эмулятор, который распознается компьютером как клавиатура и позволяет эмулировать нажатия клавиш или их сочетаний.

**Целью данной работы** является разработка стратегии ведения игры и на её основе создание робота с применением технического зрения.

Для решения поставленной цели необходимо выполнить следующие задачи:

- Исследование методов реализации выигрышной стратегии и выбор оптимального метода;
- Исследование методов распознавания на изображения;
- Разработка и реализация алгоритма выигрышной стратегии ведения игры;
- Разработка и реализация алгоритма распознавания карт на изображении;
- Проектирование и интеграция устройств ввода-вывода в робота;
- Тестирование и оценка эффективности алгоритма.

Выполнение данной цели решает проблемы теории управления, робототехники, компьютерного зрения:

Из теории управления в работе решаются проблемы:

- определения оптимального хода;
- стратегии ведения игры.

Из робототехники в работе решаются проблемы:

- конструкции робота;
- сбора и передачи данных.

Из компьютерного зрения в работе решаются проблемы:

- распознавания игрового поля;
- распознавания игровых карт;
- определения положение карт.

**Объектом исследования** в данной работе является алгоритмы выигрышной стратегии ведения игры и технического зрения, применяемые для игры роботом в пасьянс «Паук».

**Предметом исследования** является задача проектирования и разработки робота, реализующего алгоритмы выигрышной стратегии ведения игры и технического зрения.

**Научной новизной** данной работы является использование алгоритмов выигрышной стратегии ведения игры и распознавания карт на изображении.

**Реализация работы:** в ходе выполнения работы был разработан прототип.

## **1. Обзор литературы и научных источников**

В данной главе приведены работы, касающиеся тем, затронутых в диссертации.

### **1.1 Обзор исследований о пасьянс «Паук»**

В данном подразделе рассматривается история и правила игры пасьянс «Паук» и научные статьи, которые рассматривают игру в качестве объекта исследования.

#### **1.1.1 История возникновения пасьянса «Паук»**

Такой пасьянс, как «Паук», известен практически всем благодаря присутствию в наборе Aisleriot, мобильных приложениях и программном обеспечении Windows. Но до электронной эры этой игры, которая началась еще в начале девяностых, была ее карточная история.

Большинство любителей изучения игр считают, что родиной всех пасьянсов стала знаменитая Бастилия во французском Париже. Общеизвестными эти раскладки карт стали в 16 веке. Правила пасьянсов начали устанавливаться только во времена Наполеона Бонапарта, когда эти раскладки получили еще большее распространение. В российской империи первой книгой о пасьянсах руководствовались «Собранием карточных раскладок Гранд-пасьянсов». Она была выпущена в 1826. Еще одно известное издание – 1870 года от английской леди Кадоган, где был описан и пасьянс «Паук»[3].

#### **1.1.2 Правила игры пасьянса «Паук»**

Для пасьянса требуется 2 колоды по 52 карты. Сначала 54 карты разделяют на 10 столбцов, остальные 50 делятся на 5 кучек по 10 карт. В каждом столбце все верхние карты открыты. Разрешается переносить карты между столбцами. Карты можно перемещать по данным правилам:

- на пустой столбец можно переместить любую верхнюю карту;
- на верхнюю карту любого столбца разрешено переместить следующую по старшинству карту, независимо масти;
- последовательность карт одной масти, разложенных по утешению значения, можно переместить как одну карту.

Если верхняя карта столбца закрыта, то она открывается автоматически. Если собирается последовательность из по порядку собранных карт одной масти, то эта последовательность сбрасывается. Разрешается в любой момент времени взять одну из кучек карт и разложить по одной карте в каждый столбец. Партия выиграна если все карт сброшены[4].

Существует три уровня сложности этого пасьянса, которые отличаются количеством мастей. Первый уровень (One Suit) самый простой, в нем игрок раскладывает колоду из карт масти пики (каждая карта повторяется 8 раз). Второй уровень сложности (Two Suits) предлагает играть двумя мастями - пики и червы (каждая карта повторяется 4 раза). На третьем уровне сложности (Four Suits) игроку предстоит раскладывать карты всех четырех мастей (колода содержит по 2 одинаковой карты каждой масти) [5].

### **1.1.3 Исследование игры пасьянса «Паук»**

Научных статей на русском языке рассматриваемые игру не было найдено. Однако на иностранных языках они есть. Так в статье Марка Вайссера «How Many Games of Spider Solitaire are Winnable?»[6] рассчитывается математически процент игр, в которых теоретически возможно выиграть. Там он равняется 99.9%. Также существует сайт «Winnable Spider Solitaire Games» [7], созданный Алексом Робинсоном. В нем автор собрал результаты 32000 рандомных партий в пасьянс «Паук». Процент выигрышных партий равен 99.991%.

Ещё одна тема, которую рассматривают исследователи, - это какого типа является задача создания выигрышной стратегии игры пасьянс «Паук». Так в

статье «Spider Solitaire is NP-Complete»[8] автор Jesse Stern приводит доказательство того, что выигрыш пасьянса «Паук» является NP-полной задачей. В дальнейшем в статье «Spider Solitaire is NP-Compleet»[9] автор Кеннет Верстрет подтверждает это доказательство.

Если рассматривать другие карточные игры, то многие из них решены при помощи машинного обучения. Так в статье «Solitaire: Man Versus Machine»[10] авторы используют метод развертывания для игры в пасьянс «Косынка». Стратегия, которая была получена, использует повторные развертывания. Она выигрывает в среднем в два раза больше игр, чем человек. Также, используя методы машинного обучения, были разработаны стратегии для игр Червы и Шафкопф [11-12]. Эти игры уже рассчитаны на нескольких человек, и разработанные стратегии позволяют добиться высокого процента побед на соперниками-людьми.

Для сравнения разработанного алгоритма и человека необходимо найти статистику. Теоретический расчет по проценту выигрыша был взят из книги «Spider Solitaire Winning Strategies»[13] Steve Brown. В ней автор предложил стратегию, при которой процент выигрыша при 4 мастях равняется 40% при 2 80% при 1 99%. Также была взята статистика с сайта[14] где, люди непосредственно играют в пасьянс «Паук».

## **1.2 Техническое зрение**

Техническое зрение (Компьютерное зрение) — теория и технология создания машин, которые могут производить обнаружение, отслеживание и классификацию объектов.

Как научная дисциплина, компьютерное зрение относится к теории и технологии создания искусственных систем, которые получают информацию из изображений.

Как технологическая дисциплина, компьютерное зрение стремится применить теории и модели компьютерного зрения к созданию систем компьютерного зрения.

Подразделы компьютерного зрения включают воспроизведение действий, обнаружение событий, слежение, распознавание образов, восстановление изображений и некоторые другие[2].

Специализация систем технического зрения сильно зависит от области применения, аппаратной платформы и требований по производительности. Некоторые системы являются автономными и решают специфические проблемы детектирования и измерения, тогда как другие системы составляют подсистемы более крупных систем, которые уже могут содержать подсистемы контроля механических манипуляторов (роботы), информационные базы данных (поиск похожих изображений), интерфейсы человек-машина и (компьютерные игры) т.д. Однако, существуют функции, типичные для многих систем компьютерного зрения.

**Получение изображений:** цифровые изображения получаются от одного или нескольких датчиков изображения, которые помимо различных типов светочувствительных камер включают датчики расстояния, радары, ультразвуковые камеры и т. д.

**Предварительная обработка:** перед тем, как методы компьютерного зрения могут быть применены к видеоданным, чтобы извлечь определённую долю информации, необходимо обработать видеоданные, для того, чтобы они удовлетворяли некоторым условиям, в зависимости от используемого метода.

**Выделение деталей:** детали изображения различного уровня сложности выделяются из видеоданных.

**Детектирование/Сегментация:** на определённом этапе обработки принимается решение о том, какие точки или участки изображения являются важными для дальнейшей обработки.

**Высокоуровневая обработка:** на этом шаге входные данные обычно представляют небольшой набор данных, например, набор точек или участок изображения, в котором предположительно находится определённый объект.[2]

### 1.2.1 История компьютерного зрения

Компьютерное зрение имеет небольшую по меркам науки историю развития. Как самостоятельная дисциплина, компьютерное зрение зародилось в начале 1950-х годов. В 1951 Джон фон Нейман предложил анализировать микроснимки при помощи компьютеров путём сравнения яркости в соседних областях снимков.[15].

В 1960-е начались исследования в области распознавания машинописного и рукописного текста, а также в области классификации хромосом и клеток в изображениях, полученных с микроскопа. Первый успех в области технического зрения можно связать с разработкой психолога Корнельской лаборатории авиации Фрэнка Розенблатта — перцептроном. Перцептрон был впервые смоделирован на универсальной ЭВМ IBM-740 в 1958 году. Аппаратный вариант перцептрона — Mark I Perceptron был изготовлен в 1960 году и предназначался для распознавания зрительных образов. В 1963-м году появилась диссертация Робертса, который предложил простейший детектор контуров и предложил первые методы распознавания трёхмерных объектов[15].

Наличие чрезвычайно сложной взаимосвязи между свойствами трёхмерных объектов мира и их двумерными изображениями было осознано в начале 1970-х годов. В результате в 1970-х годах британский нейробиолог и психолог Марр стал основоположником научной программы, которая основывалась на изучении зрительной системы человека с целью её формализации и реализации в виде алгоритмов.

Связанные с этими этапами уровни обработки называются соответственно: обработка нижнего уровня, среднего уровня, высокого уровня[16].

В 1980-х получила популярность другая программа исследований, которая заключалась в поиске новых сложных математических методов для решения задач компьютерного зрения. В то время как применение ряда математических методов позволило решить сложные задачи компьютерного зрения, во многих случаях такой подход позволил лишь найти применение математическим методам, а не решить задачи зрения [17].

Девяностые годы XX в. с появлением нового поколения датчиков двумерных цифровых информационных полей смогло существенным образом снизить стоимость, что позволяло, значительно расширить область применения. Рост быстродействия микропроцессоров, снижение цен на камеры и скорости передачи данных при передаче видео позволили реализовать действующие в режиме реального времени алгоритмы машинного зрения на стандартных компьютерах [18].

В конце 1990-х годов и в течение первого десятилетия XXI века в компьютерном зрении произошел качественный скачок сразу в нескольких направлениях. В этом периоде трудно выявить какие-либо новые парадигмы, охватывающие всю дисциплину целиком. Скорее, скачок вызван резким ростом интереса к компьютерному зрению и, как следствие, большим энтузиазмом в переносе методов из других дисциплин в компьютерное зрение. Прежде всего, результатом этого прорыва стал прогресс в методах описания изображений [18].

### **1.2.2 Литературные источники по компьютерному зрению**

Одной из первых книг, раскрывающая технические проблемы решения задач компьютерного зрения, является труд У. Прэтта «Цифровая обработка изображений» [20,21]. Книга посвящена вопросам математического представления непрерывных и дискретных изображений. Подробно рассмотрены двумерные преобразования, в том числе преобразования Фурье, Адамара и Карунена-Лоэва. Далее, одним из известных авторов является Т.С.

Хуанг, который в 1981 г. изложил основы теории и применения новых эффективных в вычислительном отношении алгоритмов цифровой обработки изображений в книге «Быстрые алгоритмы в цифровой обработке изображения»[22].

Из современных изданий по данной тематике выделяются такие книги, как «Компьютерное зрение. Современный подход»[23] Д. А. Форсайта и «Цифровая обработка изображений»[19] Р. Гонсалеса, изданные в 2004 и 2005 годах. Эти книги описывают базовые понятия и методологию компьютерной обработки изображений, дают основы для дальнейшего изучения области.

Также можно отметить книгу «Компьютерное зрение»[24] Л. Шапиро, Дж. Стокман. В данной книге теоретические аспекты обработки зрительных данных рассматриваются с привлечением большого количества примеров из практических задач. Наряду с классическими темами, в книге рассматриваются базы данных изображений и системы виртуальной и дополненной реальности. Приведены примеры приложений в промышленности, медицине, землепользовании, мультимедиа и компьютерной графике.

Из еще не переведенных можно выделить «Computer Vision: Algorithms and Applications»[25], Richard Szeliski, «Computer Vision: Models, Learning, and Inference» Simon J.D. Prince и «Programming Computer Vision with Python»[26]. Jan Erik Solem. В первой книге автор исследует различные методы, наиболее часто используемые для анализа и интерпретации изображений. Он также описывает сложные алгоритмы, которые успешно используются в реальном мире, как для специализированных приложений, так и для задач потребительского уровня. Во второй подробно описываются передовые методы. В третьей объясняется теория компьютерного зрения на примере программ с пояснениями.

### **1.2.3 Интернет источники по компьютерному зрению**

Помимо литературных источников немало информации можно узнать на сайтах разработчиков библиотек содержащие алгоритмы компьютерного зрения. Наиболее популярной библиотекой является Open Computer Vision (OpenCV). На сайте [opencv.org](http://opencv.org) разработчика можно найти документацию, книги (Learning OpenCV, Mastering OpenCV) и примеры реализованных программ [28]. На сайте [vlfeat.org](http://vlfeat.org) представлены алгоритмы компьютерного зрения на чистом C, есть интерфейсы для MathLab. Сайт [www.simplecv.org](http://www.simplecv.org) содержит библиотеку на c/c++, построенную поверх OpenCV. Также есть готовое пособие для начинающих – «Practical Computer Vision with SimpleCV»[29]. Основная цель проекта – предоставить упрощенный интерфейс ко всем алгоритмам.

Также существуют написанные на C++ библиотеки ViSP с алгоритмами компьютерного зрения (преимущественно в области отслеживания-трекинга и наблюдения) [30] и SHARK – Machine learning library с алгоритмами машинного обучения, выгодно отличается от альтернатив наличием больше нигде не реализованных алгоритмов[31]. Сайт OpenVIDIA : Parallel GPU Computer Vision содержит алгоритмы компьютерного зрения на GPU (CUDA) [32].

Наряду с OpenCV, для начала изучения и использования алгоритмов компьютерного зрения можно применить MATLAB (toolbox + sample) [33].

### **1.2.4 Материалы от вузов по компьютерному зрению**

Наряду с сайтами разработчиков многие университеты выкладывают в свободный доступ материалы по курсу компьютерное зрение. Так на сайте университета штата Пенсильвания можно найти лекции Роберта Коллинза по дисциплине введение в компьютерное зрение [34]. В Техасском университете в Остине были подготовлены семинары в области компьютерного зрения. В них обследовали и обсуждали текущие концепции, относящиеся к объекту

распознавания, автоматического аннотирования изображений и сцен понимание. Целью семинаров было понимание современных подходов к некоторым важным проблемам и определение возможных направления будущих исследований [35]. В Массачусетском технологическом институте есть курс, разработанный преподавателем Антонио Торральба, охватывающий современные подходы к распознаванию объектов и связь компьютерного зрения с другими дисциплинами. Цель этого курса - углубленное изложение методов компьютерного зрения для распознавания объектов, сцен, материалов, действий [36].

### **1.2.5 Исследования распознавания игровых карт**

Статьи о распознавании игровых карт встречаются довольно часто в англоязычных научных публикациях. Часто это связано с работой систем мониторинга и анализа игр в казино. Так в статьях «Computer Vision for Card Games» [37] и «Machine Vision in Casino Game Monitoring» [38] авторы рассматривают распознавание карт, расположенных на реальном игральном столе, и используют разные методы. Так в первой статье авторы для распознавания используют нейросеть, а во второй контурный анализ и поиск по ключевым точкам.

## 2. Выбор методов и технических решений

В данном разделе представлены обзор и сравнение методов реализации выигрышной стратегии и методов распознавания на изображения. После проведения сравнений выбраны оптимальные методы. Также рассмотрены и выбраны методы и технологии эмуляции устройства ввода.

### 2.1 Выбор методов реализации выигрышной стратегии

#### 2.1.1 Постановка задачи

Как было рассмотрено в разделе 1.1.3 задача нахождения выигрышной стратегии в игре пасьянс «Паук» является NP-полной задачей. В теории алгоритмов NP-полная задача — это задача из класса NP, к которой можно свести любую другую задачу из класса NP. NP-полные задачи образуют подмножество «самых сложных» задач в классе NP, и если для какой-то из них будет найден «быстрый» алгоритм решения, то и любая другая задача из класса NP может быть решена так же «быстро». [39]

Взаимоотношение между классами P, NP, NP-complete (NP-полными задачами), NP-hard (NP-трудными задачами) в случае, если  $P \neq NP$  и если  $P = NP$  представлено на рисунке 2.1.

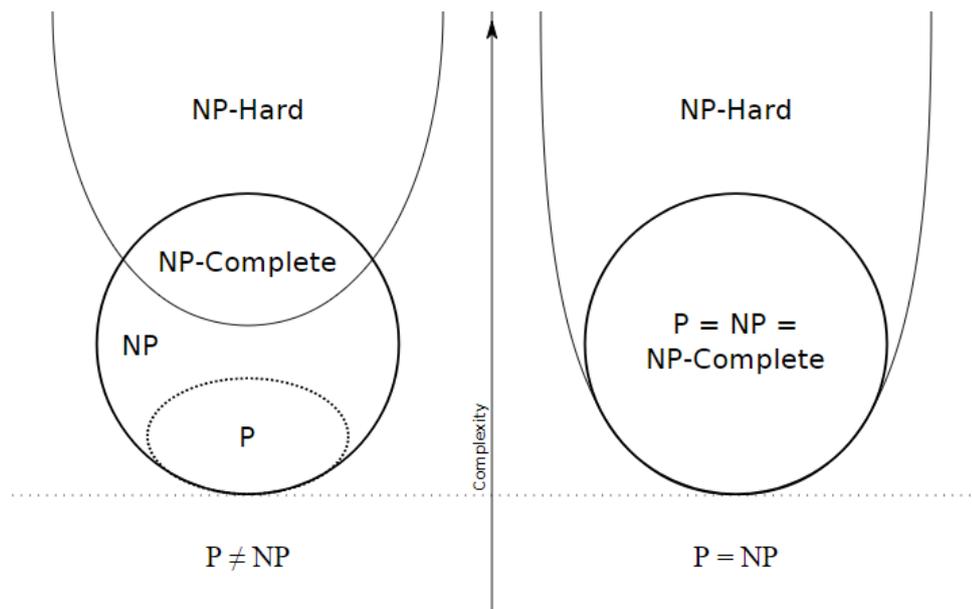


Рисунок 2.1 — Взаимоотношение между классами

Исходя из правил игры, можно составить алфавит. Составим матрицу  $C_n = 2(\{\spadesuit, \clubsuit, \heartsuit, \diamondsuit\}) \times \{1, \dots, n\}$  где  $n$  номер поля и является целым числом  $n \in \mathbb{N}$ , а элементами являются карты. Каждая карта описывается двумя переменными  $(s, v)$  где  $s$  - масть, а  $v$  - значение. Карты бывают черными, если их масть  $\spadesuit$  или  $\clubsuit$ , и красными, если  $\heartsuit$  или  $\diamondsuit$ .  $n$ - $w$  таблица — это набор непустых последовательностей размером не более  $w$  и больше  $C_n$ . Также каждая карта появляется в двух последовательностях в таблице. Последняя карта последовательности называется верхней картой. Подпоследовательность, полученная удалением верхней карты, называется скрытой частью последовательности. Карта принадлежит последовательности, если карта с имеет значение на один меньше, чем значение верхней карты.

Пасьянс паук — это четырехмерный упорядоченный набор фиксированной длины  $(n, w, d, T)$ , где

- $n \in \mathbb{N}$  число полей;
- $w \in \mathbb{N}$  ширина таблица;
- $d \in \mathbb{N}$  размер поля
- $T$  это начальная  $n$ - $w$  таблица.

### 2.1.2 Описание методов решения задачи

Существуют две общие категории подходов к проблеме отыскания приемлемого алгоритма решения NP-полной задачи. Классификация методов представлена на рисунке 2.2.

К первой группе относятся подходы, в которых делается попытка максимального сокращения объема перебора, хотя при этом и признается неизбежность экспоненциального времени работы.

Подходы, относящиеся ко второй категории, применимы исключительно к оптимизационным задачам и включают прием, который можно назвать “снижение требований”. Он заключается в отказе от поиска оптимального решения и в нахождении вместо этого хорошего решения за приемлемое время.

Алгоритмы, основанные на этом приеме, обычно называются эвристическими или приближенными, поскольку они используют различные разумные соображения без строгих обоснований. [40]

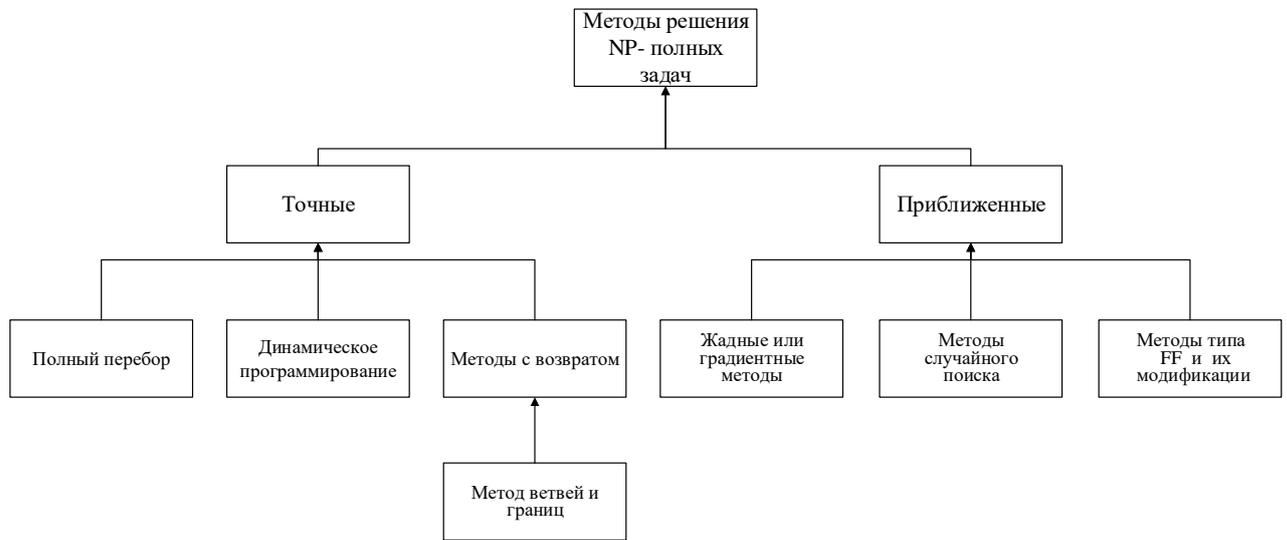


Рисунок 2.2 — Классификация методов решения NP-полной

Рассмотрим методы решения более подробно:

- Полный перебор.

Метод решения математических задач, который заключается в переборе все возможных вариантов. Пример работы метода приставлен на рисунке 2.3.[41]

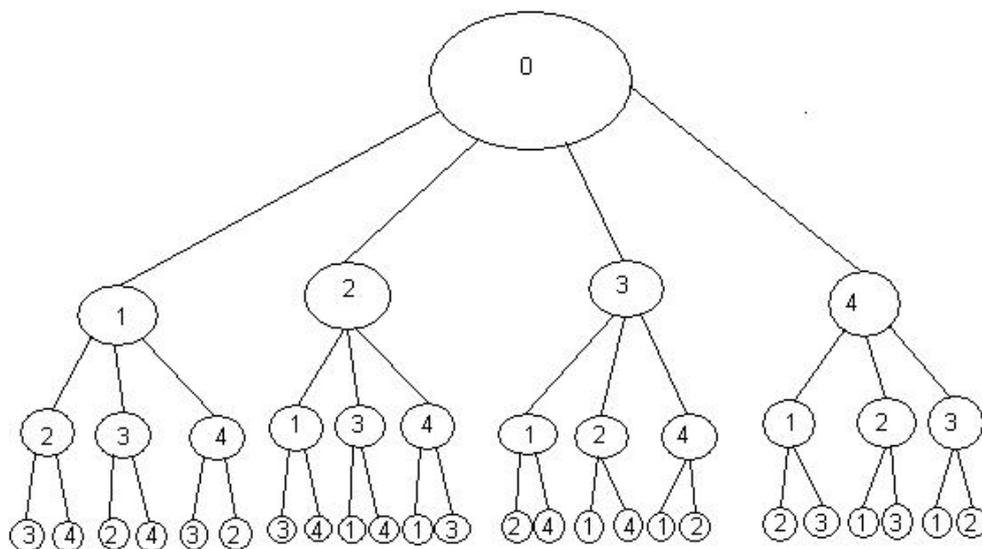


Рисунок 2.3 - Дерево перебора

- Динамическое программирование.

Динамическим программированием называют процесс пошагового решения задачи, когда на каждом шаге выбирается одно решение из множества допустимых решений, причем такое, которое оптимизирует заданную целевую функцию или функцию критерия.

Однако, возможности применения ограничиваются, в частности, следующими факторами: алгоритм на базе динамического программирования не всегда является эффективным алгоритмом, хотя при этом может быть существенно лучше метода полного перебора; задача может быть устроена так, что нельзя найти "разумный" способ сведения ее к ряду подзадач.[41]

– Метод ветвей и границ

Метод ветвей и границ - это вариация полного перебора, с исключениями заведомо не оптимальных решений. Решение состоит в построении "частичных решений", представленных в виде дерева поиска, и применении мощных методов построения оценок, позволяющих опознать бесперспективные частичные решения, в результате чего от дерева поиска на одном шаге отсекается целая ветвь. Пример работы метода приведен на рисунке 2.4.[41]

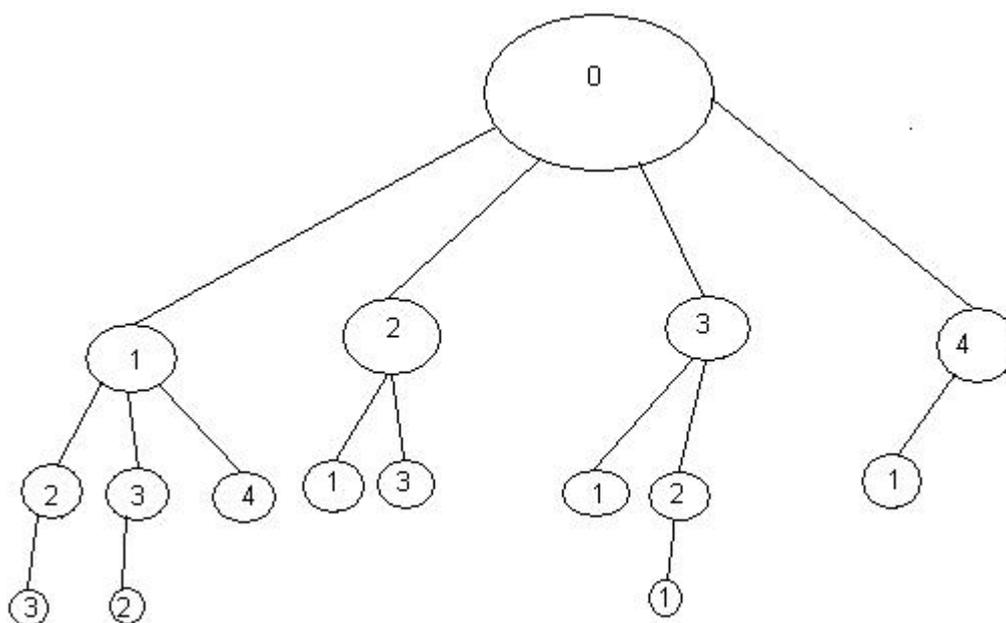


Рисунок 2.4 - Дерево перебора с осечёнными ветвями

– "Жадные" или градиентные алгоритмы

Применение градиентного алгоритма к задаче выбора дает точное решение. Однако в силу своей простоты он применяется для построения приближенных или эвристических алгоритмов. На каждой отдельной стадии "жадный" алгоритм выбирает тот вариант, который является локально оптимальным в том или ином смысле. [41]

– Методы случайного поиска

Методы из этой группы основаны на том, что генерируется случайная последовательность, затем проверяется, является ли она решением. Из множества найденных решений выбирается наилучшее. Критерием прекращения поиска может служить или время поиска, или количество итераций.

Дальнейшим развитием идеи жадных алгоритмов является разработка их вероятностных аналогов, так называемых GRASP-алгоритмов.[40]

– Методы типа FF и их модификации

Данные методы основываются на выборе первого, подходящего по какому-либо критерию. Трудоемкость алгоритма  $O(n)$ . Эти методы используются для нахождения первоначального решения, которое затем улучшается другим методом.[38]

### 2.1.3 Сравнение методов и выбор оптимального

Сравним описанные методы и сведем в таблицу в таблицу 2.1.

Таблица 2.1 – Сравнительный анализ методов

Метод	Тип алгоритма	Сложность	Плюсы	Минусы
Полный перебор	Точный	$O(N!)$	Простота реализации; точное решение	входные данные не велики; временная сложность
Метод динамического программирования	Точный	$O(W*N)$	Независимость от вида исходных данных; точное решение	большой объем вычислительной работы
Метод ветвей и границ	Точный	$O(N!)$	Возможно сокращение времени; простота реализации	работает как полный перебор

Метод	Тип алгоритма	Сложность	Плюсы	Минусы
Жадный алгоритм	Приближенный	$O(N \cdot \log(N))$	Высокая скорость; может работать с большими значениями N; простота реализации	Решение неточное
Методы случайного поиска	Приближенный		Высокая скорость; может работать с большими значениями N; независимость от вида исходных данных	не гарантирует нахождение оптимального решения

В результате сравнения был выбран метод ветвей и границ, который сокращает время, простотой в реализации, однако, недостатком метода является то, что он работает как полный перебор.

Задачу получения выигрышной стратегии игры пасьянс «Паук» необходимо разбить на несколько подзадач такие как:

– **Ограниченный перебор и оценка каждой последовательности;**

Для данной подзадачи необходимо разработать систему ограничений перебора, критерии оценивания последовательностей;

– **Выбор оптимальной последовательности.**

В данной подзадаче надо разработать критерии, по которым последовательность будет считаться оптимальной.

– **Оценка оптимальной последовательности относительно пройденных этапов;**

В этой подзадаче необходимо разработать критерии сравнения оптимальной последовательности с пройденными этапами и в результате выбрать действия для этапа.

– **Выбор критериев оценки окончания и результатов работы.**

В данной подзадаче необходимо разработать критерий, по которому будет проводиться оценка успешности окончания работы.

В результате можно составить обобщенную блок-схему алгоритма, которая представлена на рисунке 2.5.

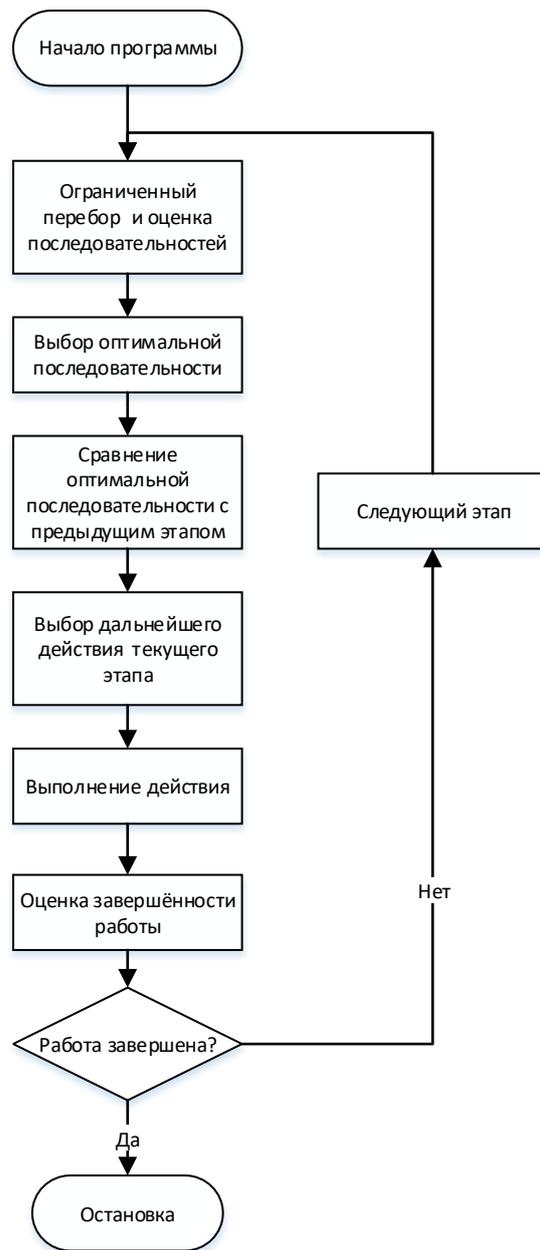


Рисунок 2.5 - Обобщенная блок-схема алгоритма

## 2.2 Выбор методов распознавания на изображения

### 2.2.1 Постановка задачи распознавания на изображения

Для того чтобы разработанный робот мог играть в пасьянс «Паук» по приведенному выше алгоритму ему необходимо получать данные о картах на экране с помощью камеры. Для этого необходимо разработать алгоритм распознавания игры карт на изображении. Алгоритм должен распознавать карты и учитывать их расположения.

Также алгоритм должен учитывать следующее:

- Две одинаковые карты;
- Пустые столбцы;
- Собранные колоды.

### **2.2.2 Описание методов распознавания на изображения**

В данном разделе рассмотрим три существующих метода распознавания объектов на изображении: контурный анализ, поиск шаблона (более известный, как *template matching*) и сопоставление по ключевым точкам (*feature detection, description & matching*).

Контурный анализ — это один из важных и очень полезных методов описания, хранения, распознавания, сравнения и поиска графических образов/объектов. Контур — это внешние очертания (обвод) предмета/объекта.

При проведении контурного анализа:

- полагается, что контур содержит достаточную информацию о форме объекта;
- внутренние точки объекта во внимание не принимаются.

Вышеприведённые положения, разумеется, накладывают существенные ограничения на область применения контурного анализа, которые, в основном, связаны с проблемами выделения контура на изображениях:

- из-за одинаковой яркости с фоном объект может не иметь чёткой границы, или может быть зашумлён помехами, что приводит к невозможности выделения контура;
- перекрытие объектов или их группировка приводят к тому, что контур выделяется неправильно и не соответствует границе объекта.

Однако, переход к рассмотрению только контуров объектов позволяет уйти от пространства изображения – к пространству контуров, что существенно снижает сложность алгоритмов и вычислений.[42]

Чтобы оперировать полученным контуром, его необходимо как-то представить (закодировать). Например, указывать вершины отрезков, составляющих контур. Другой известный способ кодирования контура – это цепной код Фримена (Рисунок 2.6).[43]

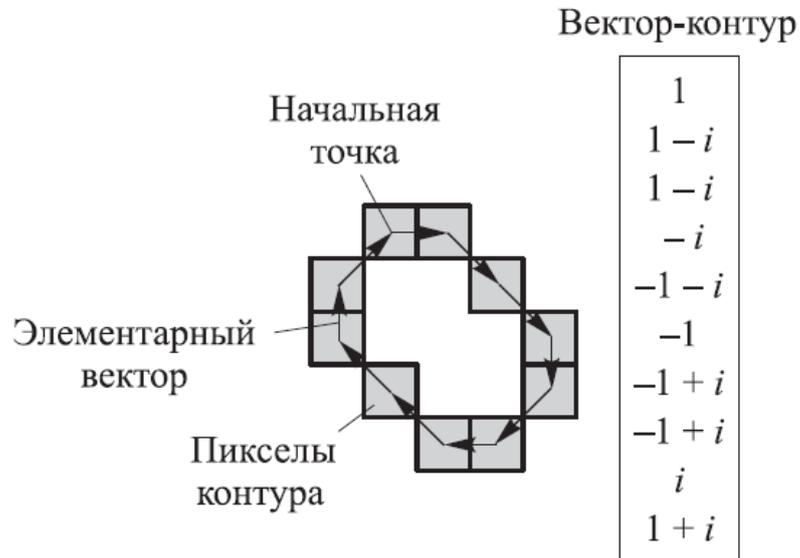


Рисунок 2.6 - Кодирование контура

Метод поиск шаблона применяется для поиска участков изображений, которые наиболее схожи с некоторым заданным изображением. Таким образом, входными параметрами метода являются:

- изображение, на котором мы будем искать шаблон;
- изображение объекта, который мы хотим найти на тестируемой картинке; размер шаблона должен быть меньше размера проверяемого изображения.

Цель работы алгоритма — найти на тестируемой картинке область, которая лучше всего совпадает с шаблоном. Пример приведен на рисунке 2.7.



Рисунок 2.7 – Пример распознавание по шаблону

Поиск шаблона производится путем последовательного перемещения его на один пиксель за раз по тестируемому изображению, и оценкой схожести каждой новой области с шаблоном. По результатам проверки выбирается та область, которая имеет наивысший коэффициент совпадения. По сути — это процент совпадения области картинки и шаблона.[42]

Однако стоит понимать, что `template matching` не позволяет с уверенностью сказать был ли найден исходный объект, поскольку это вероятностная характеристика, зависящая от масштаба, углов обзора, поворотов картинки и наличия физических помех.

Концепция `feature detection` в компьютерном зрении относится к методам, которые нацелены на вычисление абстракций изображения и выделения на нем ключевых особенностей. Данные особенности затем используются для сравнения двух изображений с целью выявления у них общих составляющих.

Рассмотрим алгоритмы, которые для своей работы используют ключевые точки (`feature points`) изображения. Под ключевыми точками понимаются некоторые участки картинки, которые являются отличительными для данного изображения.

Подобные точки каждый алгоритм определяет по-своему. Для нахождения ключевых точек на изображениях и последующего их сравнения используются три составляющие:

- Детектор (`feature detector`) — осуществляет поиск ключевых точек на изображении.
- Дескриптор (`descriptor extractor`) — производит описание найденных ключевых точек, оценивая их позиции через описание окружающих областей.
- Матчер (`matcher`) — осуществляет построение соответствий между двумя наборами точек изображений.[42]

### 2.2.3 Сравнение методов распознавания на изображения

Сравним описанные методы и сведем в таблицу в таблицу 2.1.

Таблица 2.1 – Сравнительный анализ методов распознавания

Метод	Плюсы	Минусы
Контурный анализ	Простота и быстрдействие;	Слабая устойчивость к помехам;
Поиск шаблона	Простота	Ложные срабатывания там, где нет объекта; не позволяет с уверенностью сказать был ли найден объект; зависит от масштаба, углов обзора, поворотов картинки и наличия физических помех;
Сопоставление по ключевым точкам	Устойчивость к помехам, трансформациям и физическим помехам; высокая скорость работы;	Объект должен обладать достаточно большим числом уникальных ключевых точек

Из рассмотренных методов единственным неподходящим методом является поиск по шаблону из-за ложных срабатываний там, где нет объекта. Методы контурного анализа и сопоставление по ключевым точкам были выбраны для распознавания.

С помощью контурного анализа будут найдены карты на изображении и определено положение, а методом сопоставления по ключевым точкам уже определены масть и имя. Однако перед этим необходимо подобрать фильтры, которые уберут шумы и выделяют контуры карт. В завершении алгоритма надо определить по расположению карты, к каким столбцам на игровом поле они относятся. В результате составлена обобщенная блок-схема алгоритма распознавания карт, которая представлена на рисунке 2.8.



Рисунок 2.8 - Блок-схема алгоритма распознавания

### 2.3 Выбор технологий эмуляции устройства ввода

При наличии алгоритмов для игры в пасьянс «Паук» и распознавания карт остается разработать управляющее устройство, с помощью которого робот перемещает карты. Управляющим устройством у компьютера являются устройства ввода. Чтобы робот управлял компьютером, ему необходимо эмулировать устройства ввода. Наиболее распространённые устройства ввода являются клавиатура и мышь. Из этих двух устройств эмуляция клавиатуры имеет ряд преимуществ перед мышью:

- при большом количестве карт столбец сужается и смещает карты, так что возможность точного выбора карты с помощью мыши осложняется;
- при использовании камеры необходимо соотносить координаты с полученного изображения и экраном.

Еще одним вопросом является то, каким видом эмуляции пользоваться: программной или аппаратной. Из них была выбрана программная эмуляция, которая позволяет создать одно устройство, подходящее для всех ОС.

Эмулировать необходимо клавиатуру со спецификации USB HID. USB HID (human interface device) class — класс устройств USB для взаимодействия с человеком. Этот класс включает в себя такие устройства, как клавиатура, мышь, игровой контроллер. Одно из преимуществ USB HID — это обилие драйверов устройств, доступных в большинстве современных операционных систем. USB HID класс и его базовые функции описаны в USB-IF документации, без какой-либо привязки к конкретному программному обеспечению.[44]

### 3. Реализация алгоритма и проектирование работа

#### 3.1 Выбор программного обеспечения для реализации алгоритмов

Выбор программного обеспечения необходимо начать с выбора операционной системы, на которой будет происходить разработка проекта. На выбор из популярных ОС есть Microsoft Windows, Linux, macOS. Последняя сразу отпадает, так как для её работы необходимы специально разрешенные устройства.

Для сравнения сведем плюсы и минусы Windows и Linux в таблицу 3.1.

Таблица 3.1 – Сравнение ОС Windows и Linux

	Плюсы	Минусы
Windows	Совместимость с оборудованием и сторонним ПО, поддержка пользователей, функциональность системы.	Вирусы, скорость работы системы, цена.
Linux	Цена системы, разнообразие версий, вирусов практически нет.	Сложность освоения ОС, совместимость с оборудованием.

Исходя из сравнений, была выбрана операционная система Linux. Наиболее популярным дистрибутивом Linux является Ubuntu. Однако был выбран не оригинальный дистрибутив Ubuntu, а производный Lubuntu. Основным преимуществом его является легковесность, энергоэффективность и малое потребление ресурсов.

Системные требования данного дистрибутива рекомендуют использовать ЦП минимум Pentium 4, Pentium M, AMD K8 и SVGA совместимый видеоадаптер. Оперативной памяти не менее 512 Мб (1 Гб рекомендуется), а также 6 Gb на жестком диске. Аудиокарта подойдет любая совместимая. Из периферии необходимы монитор с разрешением 1024x768, клавиатура, мышь, DVD-ROM или USB, доступ к интернету.[45]

Среду разработки необходимо выбирать исходя из языка программирования, на котором будет написано ПО. Проводя обзор литературы, было замечено, что большинство библиотек по компьютерному зрению

поддерживают язык программирования C++. Отсюда и выбор C++ языком программирования, на котором разрабатывается проект.

Средами разработки, которые есть на Linux и дают возможность разработки проектов на C++, являются: Code::Blocks, Netbeans, Eclipse, QtCreator. Сведем в таблицу 3.2 плюсы и минусы выбранных сред.

Таблица 3.2 – Сравнение сред разработки

	Плюсы	Минусы
Code::Blocks	Поддержка большинства компиляторов, интегрированный список задач, дебаггер, GUI и многое другое. бесплатность; простота интерфейса; возможность выбора и подгрузки различных компиляторов; множество шаблонов приложения для разных библиотек (wxWidgets, Ogre3D, Irrlicht, Opengl, SDL и т.д.) кроссплатформенность	Редкость официальных релизов; нет подсветки ошибок в редакторе; полное отсутствие рефакторинга;
Eclipse	Бесплатность, плагины (невероятно много), рефакторинг, автоподстановка любых идентификаторов, встроенная поддержка Doxygen, возможность отладки через GDB, кроссплатформенность	Высокий уровень вхождения.
NetBeans	Бесплатность; плагины; рефакторинг; автоподстановки методов, классов; возможность отладки, включая удалённую; кроссплатформенность.	Недостаточная поддержка разных видов сборки; требует установку и подключение нескольких компиляторов.
QtCreator	Бесплатность; плагины; рефакторинг; возможность отладки; кроссплатформенность.	Требует установку; большой вес приложений.

Исходя из сравнения, была выбрана среда разработки Code::Blocks. Code::Blocks — это среда разработки с открытым исходным кодом. Используя архитектуру плагинов, наподобие Eclipse, тут все возможности предоставляют плагины. Code::Blocks также ориентирован на C и C++, что делает его идеальным кандидатом для работы с этим языком.

В данном разделе было выбраны: операционная система Lubuntu, язык программирования C++ и среда разработки Code::Blocks.

## 3.2 Реализация алгоритма стратегии ведения игры

### 3.2.1 Настройка среды разработки

Для разработки программного обеспечения необходимо создать проект. Это происходит в несколько этапов. На первом этапе надо запустить Code::Blocks и появился главный экран, который изображен на рисунке 3.1.

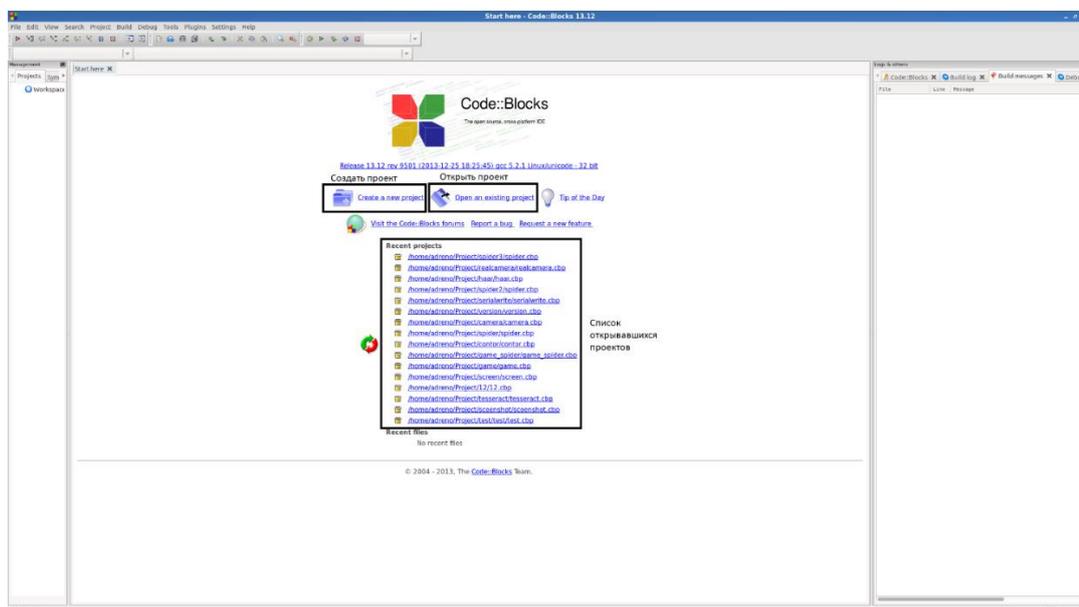


Рисунок 3.1 - Общий вид среды разработки Code::Blocks

Далее надо нажать на кнопку «Create a new project» проявится диалоговое окно выбора шаблона, изображенное на рисунке 3.2. В нем выбираем «OpenCV project» и нажимаем «Go».

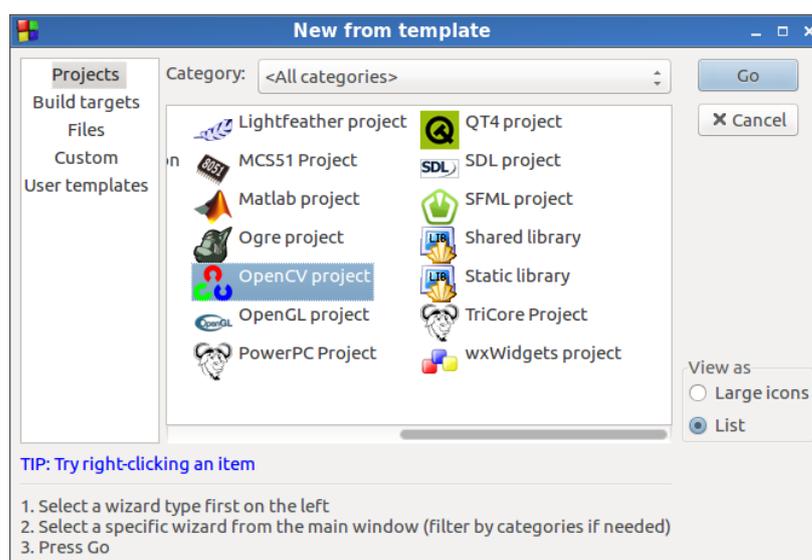


Рисунок 3.2 - Диалоговое окно выбора шаблона

Далее необходимо выбрать язык программирования в окне, изображённом на рисунке 3.3 и нажать «Next».

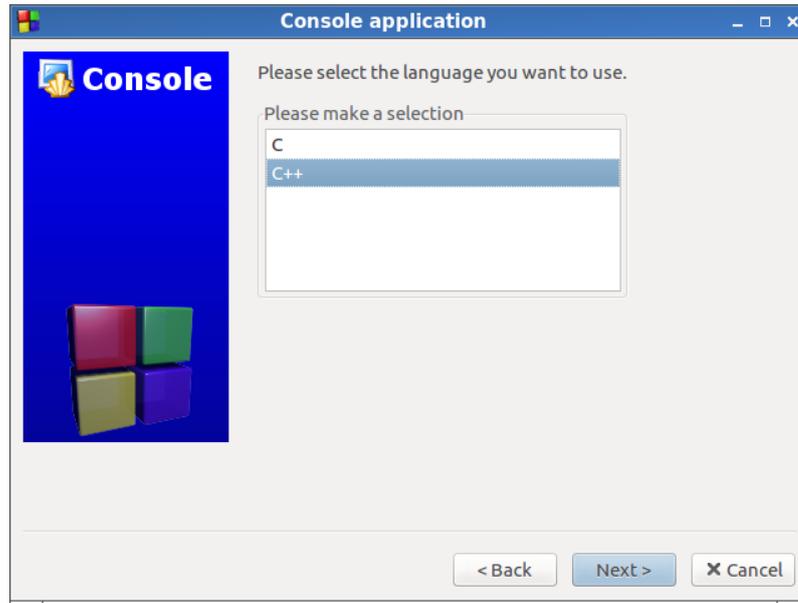


Рисунок 3.3 - Окно выбора языка программирования

Следующим этапом будет выбора названия и места расположения проекта в окне, изображённом на рисунке 3.4

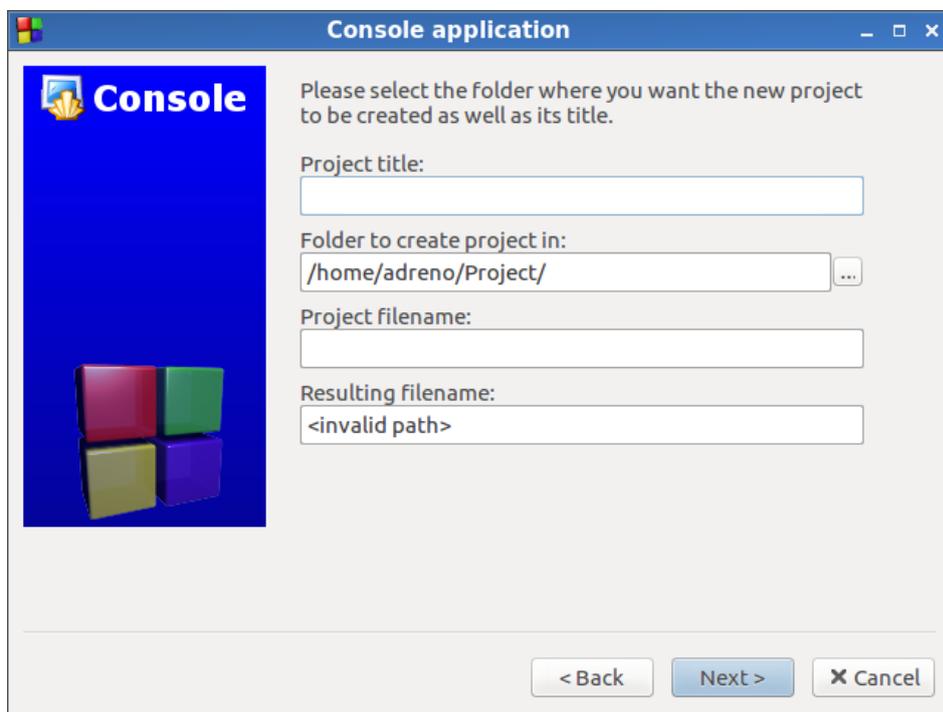


Рисунок 3.4 - Окно выбора названия и расположения проекта

И в результате будет создан проект и откроется окно, изображенное на рисунке 3.5, в котором и происходит разработка программы.

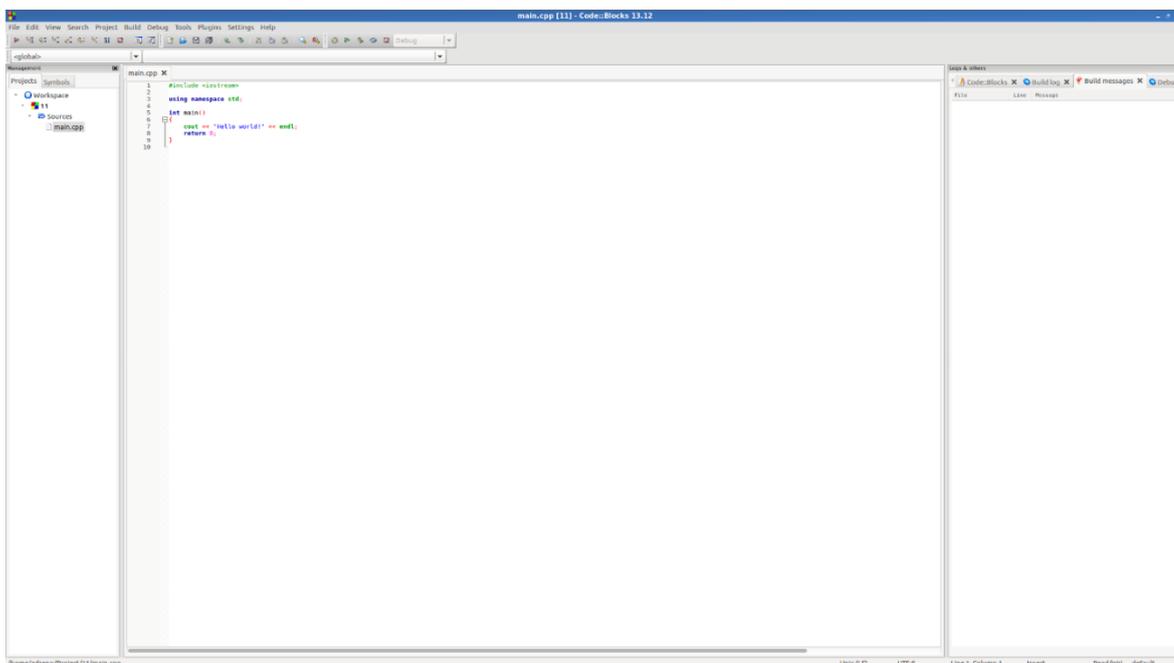


Рисунок 3.5 – Рабочее окно проекта

### 3.2.2 Разработка библиотек классов

Реализацию алгоритма начнём с создания библиотек классов описывающие объекты, использованные в игре пасьянс «Паук». Первый класс будет описывать карты, он называется Cards. Интерфейс класса представлен в приложении Г, а в проекте заголовочным файлом «Cards.h». Реализация класса описана в исполняемом файле «Cards.cpp»

Следующим будет класс, описывающий перенос карты, с названием Steps. Интерфейс класса представлен в приложении Д, а в проекте заголовочным файлом «Steps.h». Реализация класса описана в исполняемом файле «Steps.cpp»

Основным классом, описывающим игровое поле, называется GameTable. Интерфейс класса представлен в приложении Е, а в проекте заголовочным файлом «GameTable.h». Реализация класса описана в исполняемом файле «GameTable.cpp»

Последним классом программы будет SeriesSteps. В нем описывается последовательность шагов. Интерфейс класса представлен в приложении Ж, а в

проекте заголовочным файлом «SeriesSteps.h». Реализация класса описана в исполняемом файле «SeriesSteps.cpp»

### 3.2.3 Реализация алгоритма

После реализации классов необходимо решить раннее поставленные задачи. Во-первых, введем систему оценок последовательностей шагов:

- перенос на карту одинаковой масти рассчитывается по формуле:

$$R = 3 * (S_1 + S_2) \quad (3.1)$$

где  $R$  – оценка переноса;  $S_1$  – размер последовательности карт, которая переносится;  $S_2$  – размер последовательности карт, на который перенесется.

- перенос карты масти пики на карту масти червы и наоборот: 2;
- перенос карты масти крести на карту масти бубны и наоборот: 2;
- перенос последовательности карт на пустой столбец рассчитывается по формуле:

$$R = 2 * S \quad (3.2)$$

где  $R$  – оценка переноса;  $S$  – размер последовательности карт, которая переносится;

- остальные: 1.

Также разработан перебор, реализованный рекурсией, который ограничен:

- Размером последовательности шагов: 5;
- Рейтингом последовательности шагов: 200;
- Числом найденных последовательностей шагов: 25000;

Ограничения подбирались экспериментальным путем. Также введена система целей, выполнение которых увеличивает оценку последовательности:

- Открытие скрытой карты: 200;
- Открытие скрытых карт на всем столбце: 1000;
- Сбор последовательности, которую можно сбросить: 5000.

Цели рассчитывались исходя из максимально допустимой оценки. Так как максимальная допустимая оценка для последовательности размера 5 будет равна 195, тогда выполнение открытия скрытой карты будет 200. Затем вычисляются остальные.

Элемент массива последовательностей шагов будет оптимальным, если выполняется одно из условий:

- последовательность шагов выполнила одну из целей;
- средний размер последовательностей карт на поле увеличился. Расчет среднего размера последовательностей карт производится по формуле:

$$Sr = \frac{\sum \frac{S_j}{h_i}}{w} \quad (3.3)$$

где  $Sr$  – оценка переноса;  $w$  – число непустых столбцов;  $h_i$  – размер  $i$ -ого столбца;  $S_j$  – размер  $j$ -ой последовательности карт.

Партия будет считаться оконченной, если не будет найдена оптимальная последовательность шагов и все стопки сброшены. Если стопки не все сброшены, то одна сбрасывается, и игра продолжается. Выигрыш или проигрыш зависит от количества собранных последовательностей карт. Если собрано 8, то партия выиграна, остальные результаты - проигрыш. Блок-схема разработанного алгоритма представлена в приложении Б.

### **3.3 Реализация алгоритма распознавания карт на изображении**

#### **3.3.1 Установка библиотеки OpenCV**

Для начала разработки необходимо установить библиотеку OpenCV. Это делается в несколько этапов [46]:

1. Обновить Lubuntu. Для этого в консоли написать команды:

```
$ sudo apt-get update
```

```
$ sudo apt-get upgrade
```

2. Установить дополнительные библиотеки командами:

```
$ sudo apt-get install git libgtk2.0-dev pkg-config libavcodec-dev libavformat-dev libswscale-dev
```

```
$ sudo apt-get install python3.5-dev python3-numpy libtbb2 libtbb-dev
```

```
$ sudo apt-get install libjpeg-dev libpng-dev libtiff5-dev libjasper-dev libdc1394-22-dev libeigen3-dev libtheora-dev libvorbis-dev libxvidcore-dev libx264-dev sphinx-common libtbb-dev yasm libfaac-dev libopencore-amrnb-dev libopencore-amrwb-dev libopenexr-dev libgstreamer-plugins-base1.0-dev libavutil-dev libavfilter-dev libavresample-dev
```

3. Чтобы скачать OpenCV, надо использовать команды:

```
$ sudo -s
```

```
$ cd /opt
```

```
/opt$ git clone https://github.com/Itseez/opencv.git
```

```
/opt$ git clone https://github.com/Itseez/opencv_contrib.git
```

4. Установить CMake можно при помощи команды:

```
$ sudo apt-get install cmake
```

5. Собрать и установить OpenCV при помощи CMake, нужны команды:

```
/opt$ cd opencv
```

```
/opt/opencv$ mkdir release
```

```
/opt/opencv$ cd release
```

```
/opt/opencv/release$ cmake -D BUILD_TIFF=ON -D WITH_CUDA=OFF -D  
ENABLE_AVX=OFF -D WITH_OPENGL=OFF -D WITH_OPENCL=OFF -D  
WITH_IPP=OFF -D WITH_TBB=ON -D BUILD_TBB=ON -D  
WITH_EIGEN=OFF -D WITH_V4L=OFF -D WITH_VTK=OFF -D  
BUILD_TESTS=OFF -D BUILD_PERF_TESTS=OFF -D  
CMAKE_BUILD_TYPE=RELEASE -D CMAKE_INSTALL_PREFIX=/usr/local -  
D OPENCV_EXTRA_MODULES_PATH=/opt/opencv_contrib/modules  
/opt/opencv/
```

```
/opt/opencv/release$ make -j4
```

```
/opt/opencv/release$ make install
```

```
/opt/opencv/release$ ldconfig
```

```
/opt/opencv/release$ exit
```

```
/opt/opencv/release$ cd ~
```

После проведения всех операций в среде разработки Code::Blocks можно реализовывать программы распознавания.

### 3.3.2 Выбор камеры для проекта

Камера выбиралась исходя из соотношения цена качества, однако, также необходимо задать определенные технические характеристики, которые точно должны быть:

- Поддержка Linux;
- Тип подключения USB;
- Минимальное разрешение 1280x720.

Соберем все найденные камеры в таблицу 3.3 и сравним их характеристики. Также проведем сравнение качества картинки по найденным в интернете фото и видео и проставим оценки по пятибалльной системе каждой камере.

Таблица 3.3 – Сравнение характеристик камер

	Dexр H-200	Genius Facecam 1000X	Logitech C270	HP HD 2300
Расширение	640x480, 320x240, 1280x720	640x480, 1280x720	1280x720	1280x720
Оценка изображения	1	2	4	4
Цена	790	999	1 199	1 499

Как можно видеть по таблице 3.3, оптимальным вариантом будет выбрать камеру Logitech C270. Изображение камеры представлено на рисунке 3.6.



Рисунок 3.6 – Изображение камеры Logitech C270

### 3.3.3 Обработка изображений и контурный анализ

Обработка изображений является неотъемлемой частью технического зрения. Так на рисунке 3.7 представлено изображение, полученное с камеры. Можно увидеть цифровой шум, который ухудшает точность распознавания. Чтобы качество распознавания повысилось, необходимо провести ряд преобразований.

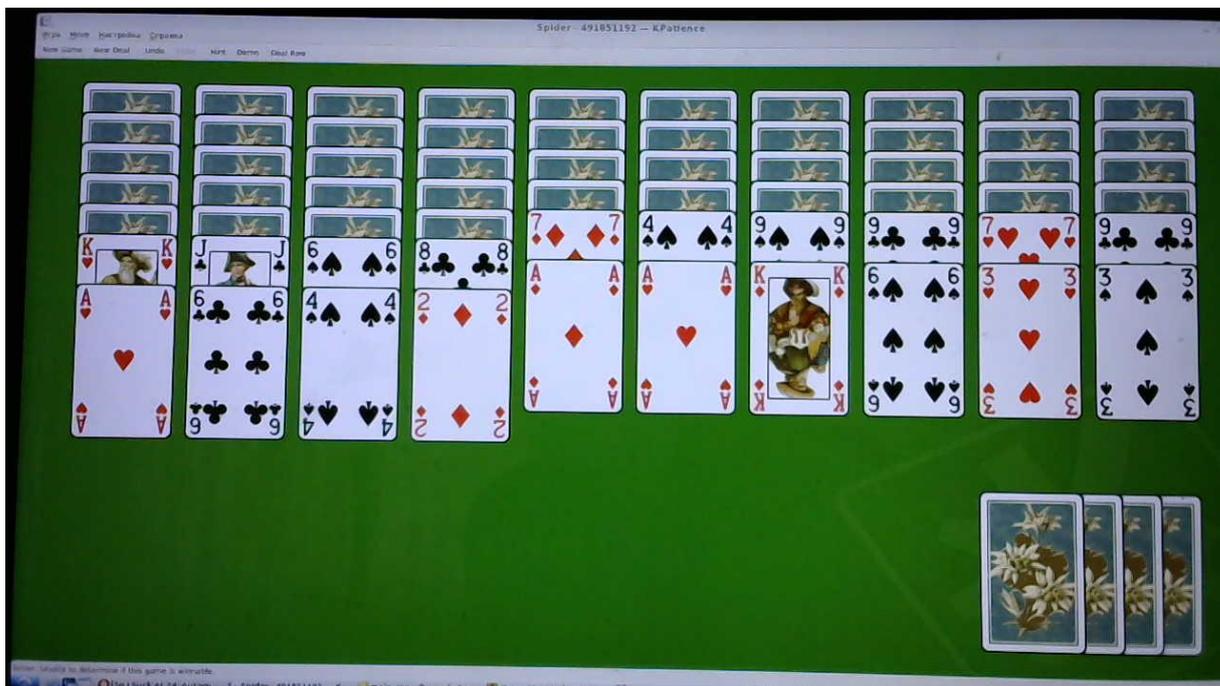


Рисунок 3.7 – Необработанное изображение с камеры

Первое, что надо сделать, это устранить шумы. Для этого применим сглаживающий фильтр. Наиболее часто используемым методом является Гауссово сглаживание (размытие). Гауссовский фильтр с ядром описывается уравнением:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (3.4)$$

где  $x$  - расстояние от центра по горизонтальной оси,  $y$  - расстояние от центра по вертикальной оси,  $\sigma$  - среднеквадратичное отклонение распределения гаусса.[45]

Размер ядра стандартного отклонения распределения Гаусса по  $x$  и  $y$  направлению были подобраны экспериментальным путем. В результате обработки Гауссовым сглаживанием (параметры  $x=1$ ,  $y=1$ ,  $\sigma=2000$ ) получено изображение, показанное на рисунке 3.8.

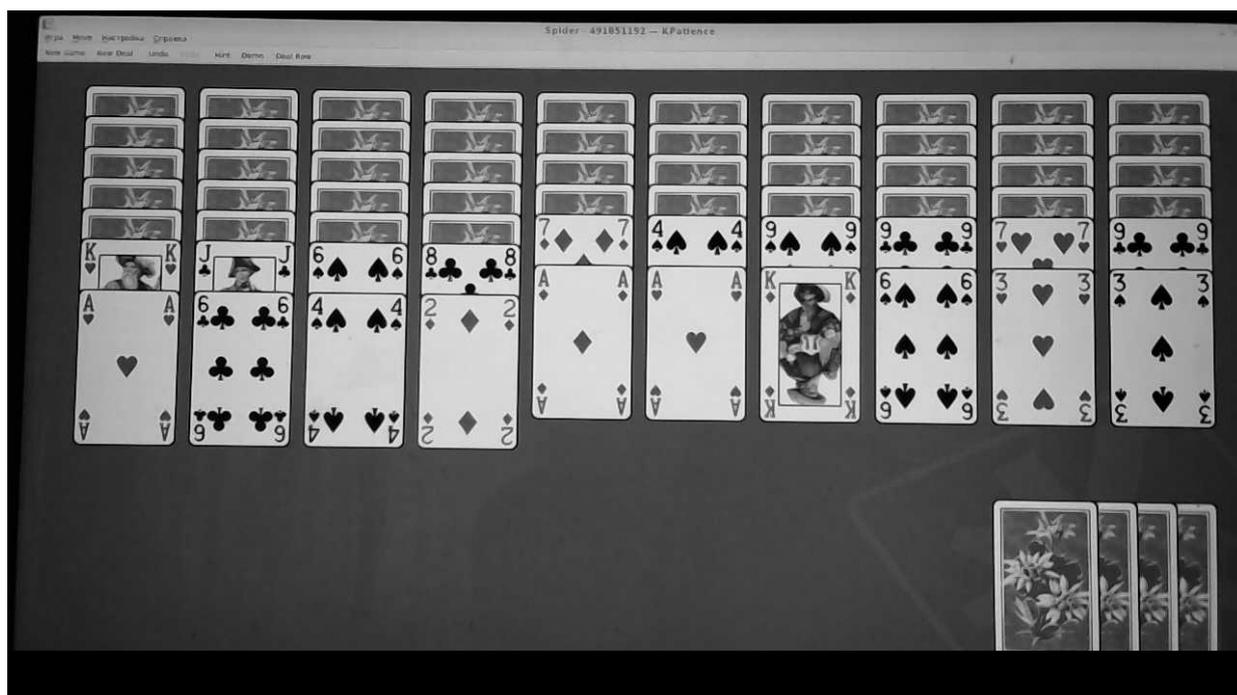


Рисунок 3.8 – Изображение, обработанное Гауссовым размытием

Следующим действием будет пороговое преобразование изображения. Фильтр выделяет объекты из окружающего фона, когда яркости пикселей объектов и фона сосредоточены вблизи двух преобладающих значений. Обычно фильтр используется для получения двухградационного (бинарного,

т.е. черно-белого) изображения из полутонового изображения или для удаления шума.

Выделение объектов происходит путем определения значения порога, разделяющего области распределения яркостей. Если значение яркости определенного пикселя больше порогового значения, то это точка принадлежит объекту, если меньше — фону [46]. Использованный тип порогового преобразования CV\_THRESH\_BINARY\_INV рассчитывается по формуле:

$$dst(x, y) = \begin{cases} 0 & \text{if } src(x, y) > threshold \\ \max\_value & \text{otherwise} \end{cases} \quad (3.5)$$

где *src* — исходный массив(изображение), *dst* — целевой массив, *threshold*=120 — пороговая величина, *max\_value*=255 — максимальное значение.[48]

В результате было получено изображение (рисунок 3.9), на котором можно точно найти контуры карт.

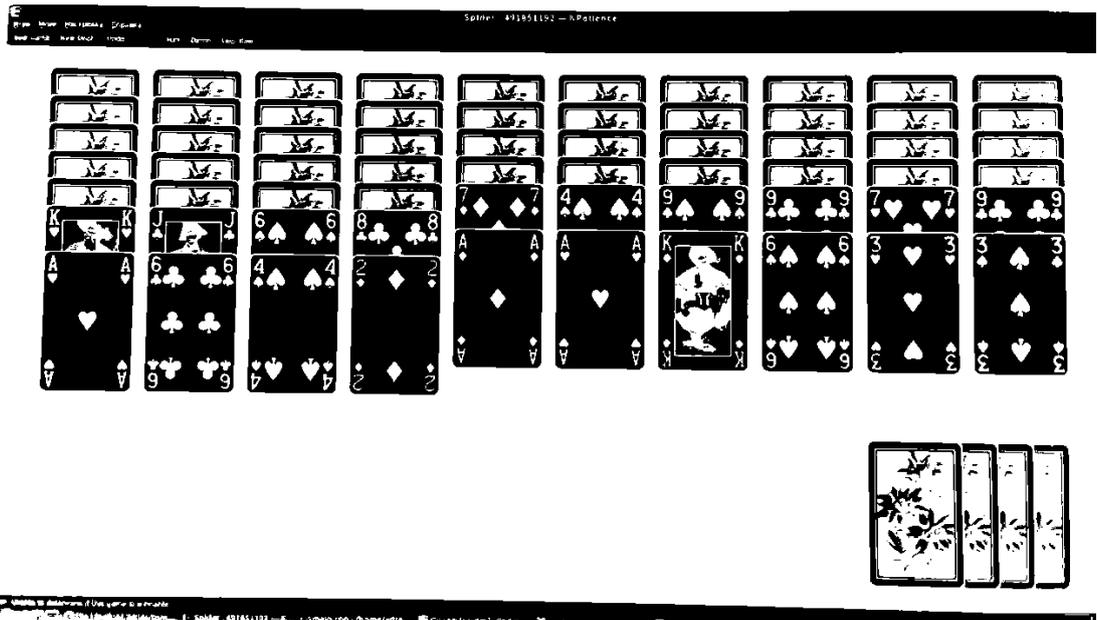


Рисунок 3.9 – Изображение, полученное пороговым преобразованием

В результате контурного анализа был получен массив контуров. Затем массив был отсортирован по площади. Далее были найдены контуры карт при помощи вычисления площади и размера каждого контура. Найденные на изображении контуры показаны на рисунке 3.10.

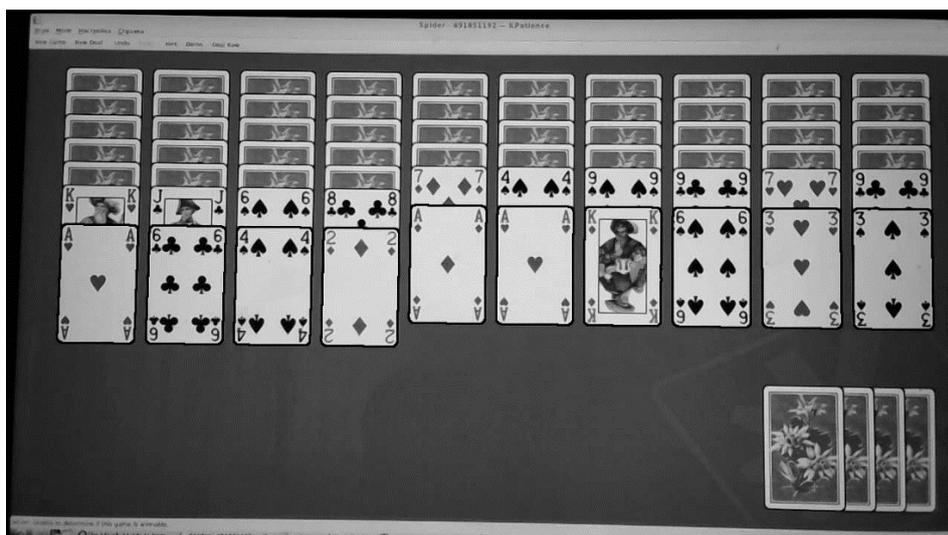


Рисунок 3.10 – Найденные контуры карт

Полученные изображения карт необходимо спроецировать на плоскость и масштабировать до размера шаблона. Для этого надо:

1. Найти прямоугольник максимального размера, который может вписаться в заданный замкнутый контур;
2. Вычислить матрицу преобразование, которая позволит преобразовать в необходимое изображение;
3. Преобразовать в необходимое изображение.

Пример полученного и исходного изображений представлены на рисунке 3.11.

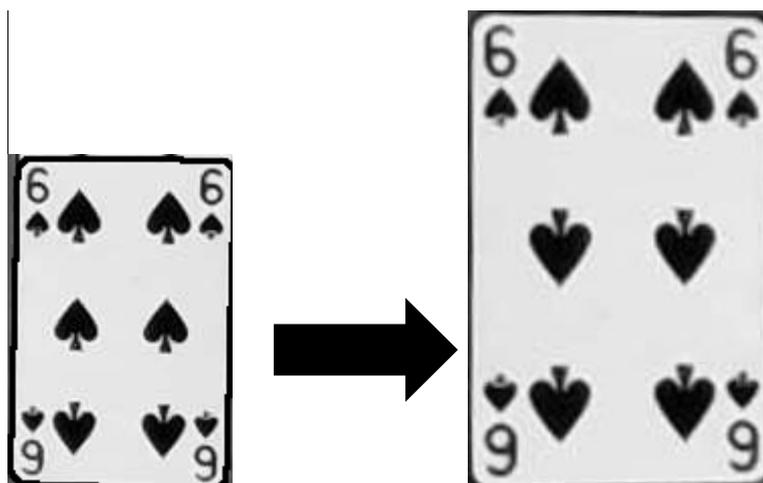


Рисунок 3.11 – Масштабирование и проецирование изображения

В заключении контурного анализа полученный массив был отсортирован по расположению, и удалены карты, которые ниже других карт.

### 3.3.4 Распознавание карт по ключевым точкам и реализация алгоритма

Получение изображения сравниваются с шаблонами карт по ключевым точкам. В разработанном алгоритме использовались:

- Детектор KAZE;
- Дескриптор матчер BruteForce.

KAZE детектор - новый метод обнаружения и описания 2D-функций, который полностью работает в нелинейном масштабе. Предыдущие методы, такие как SIFT или SURF, находят функции в гауссовском масштабе. Однако гауссовское размытие не учитывает естественные границы объектов и сглаживает в одинаковой степени детали и шум при разработке исходного изображения через масштабное пространство. [49]

BruteForce - это простой метод. Он принимает дескриптор одного объекта в первом наборе и сопоставляет со всеми другими объектами во втором наборе, используя некоторые вычисления расстояния. И ближайший возвращается.

Примера работы представлен на рисунке 3.12.

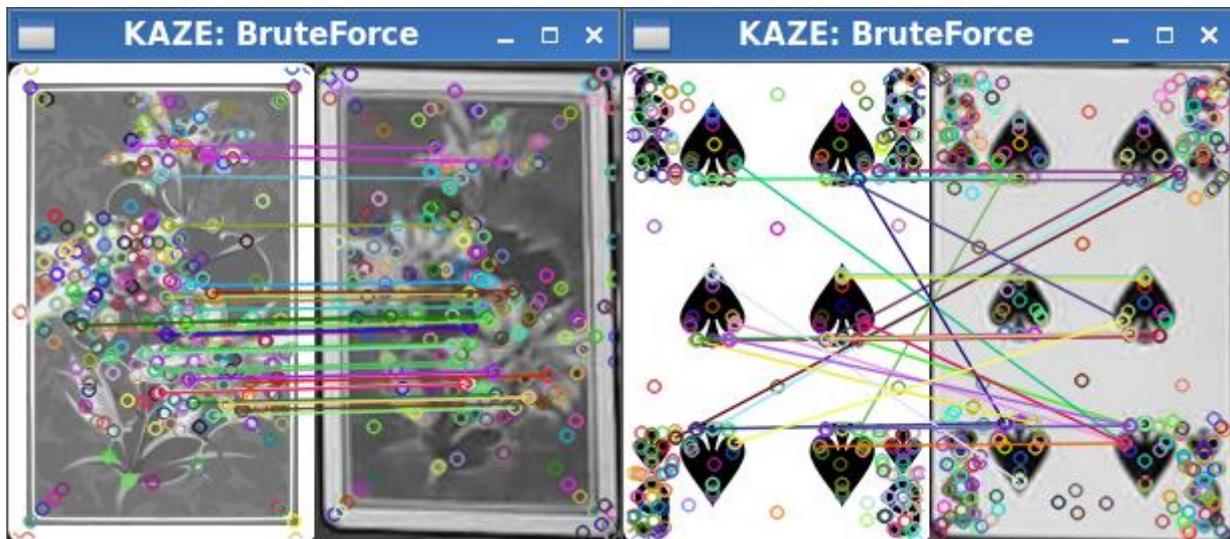


Рисунок 3.12 – Распознавание карт по ключевым точкам

В результате был разработан алгоритм распознавания карт на изображении. Блок-схема алгоритма представлена в приложении В.

### 3.4 Разработка эмулятор клавиатуры

#### 3.4.1 Выбор аппаратной и программной платформы

Выбор аппаратной платформы с поддержкой USB HID был ограничен семейством Arduino, так как существуют готовые библиотеки, которые позволяют в короткие сроки разработать эмулятор устройств ввода-вывода. Однако, выберем и сравним платы которые поддерживают USB HID в таблице 3.4.

Таблица 3.4 – Сравнение плат Arduino

	Due	Leonardo	Arduino Pro Micro	GENUINO ZERO
МК	Atmel SAM3X8E ARM Cortex-M3	Atmega32u4	Atmega32u4	ATSAMD21G18, ARM Cortex M0+
Число UART	3	1	1	2
Цена	1290	1300	200	2820

В результате сравнения была выбрана плата Arduino Pro Micro из-за низкой цены. Характеристики платы [48]:

- Микроконтроллер: ATmega32u4
- Предельное напряжение питания: 5-20 В
- Рекомендуемое напряжение питания: 7-12 В
- Цифровых вводов/выводов: 18
- ШИМ: 7 цифровых пинов могут быть использованы как выводы ШИМ (5 в Pro версии)
- Аналоговые выводы: 4
- Максимальная сила тока: 40 mAh с одного вывода и 500 mAh со всех выводов.
- Flash память: 32 КБ
- SRAM: 2,5 КБ
- EEPROM: 1 КБ
- Тактовая частота: 16 МГц

Изображения платы представлено на рисунке 3.12

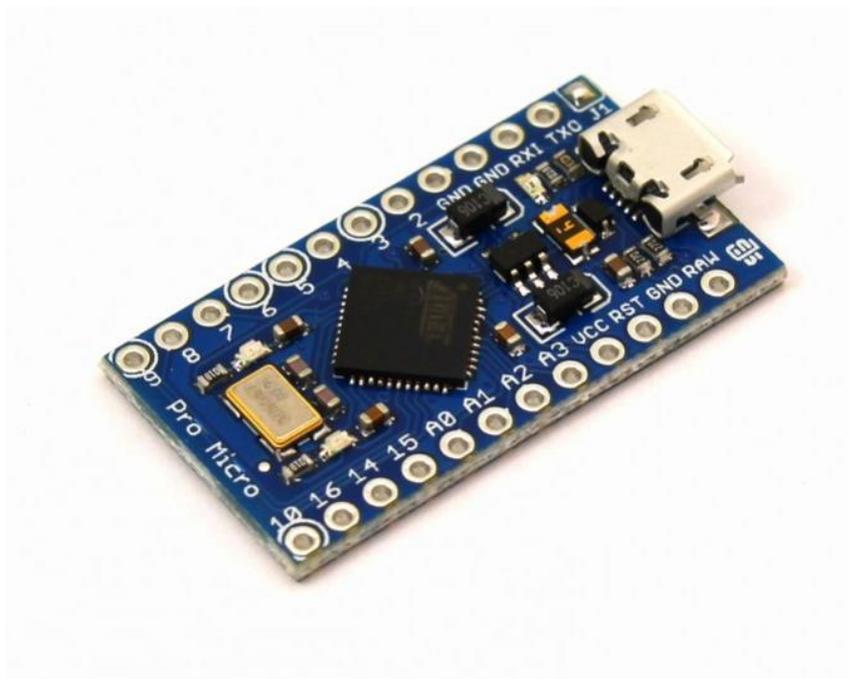


Рисунок 3.12 –Изображение платы Arduino Pro Micro

### 3.4.2 Реализация аппаратной составляющей

Для аппаратной реализации надо определить интерфейсы эмулятора клавиатуры. Так как на у микроконтроллера один UART-интерфейс, то понадобится две платы Arduino Pro Micro. Для связи между ними будет использован интерфейс i2c, так как выводы для связи по интерфейсу SPI не подключены к цифровым пинам.

В результате была разработана структурная схема устройства, изображенная на рисунке 3.13.

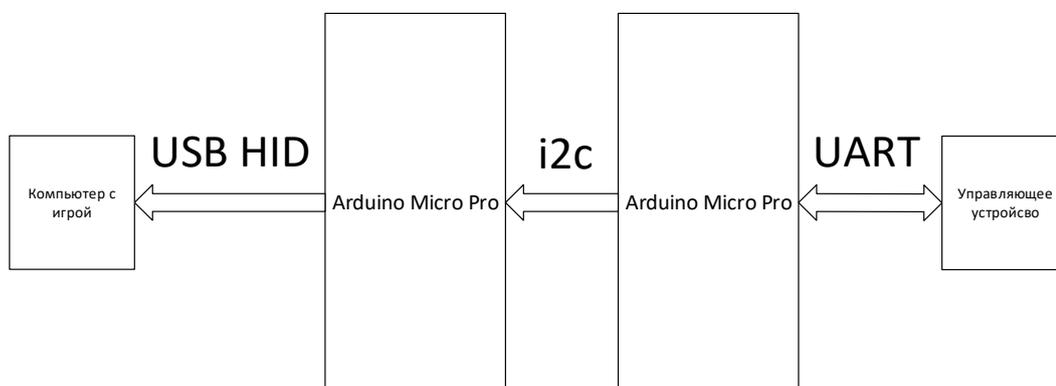


Рисунок 3.13 - Структурная схема эмулятора клавиатуры

Для соединения по интерфейсу i2c необходимо использовать 2 пина у каждой платы. Это пины 2 и 3. Также соединены платы по GND. Для удобства использования платы размещены на макетной плате. В результате получено устройство, изображенное на рисунках 3.14 и 3.15.



Рисунок 3.14 – Прототип эмулятора клавиатуры (вид сверху)

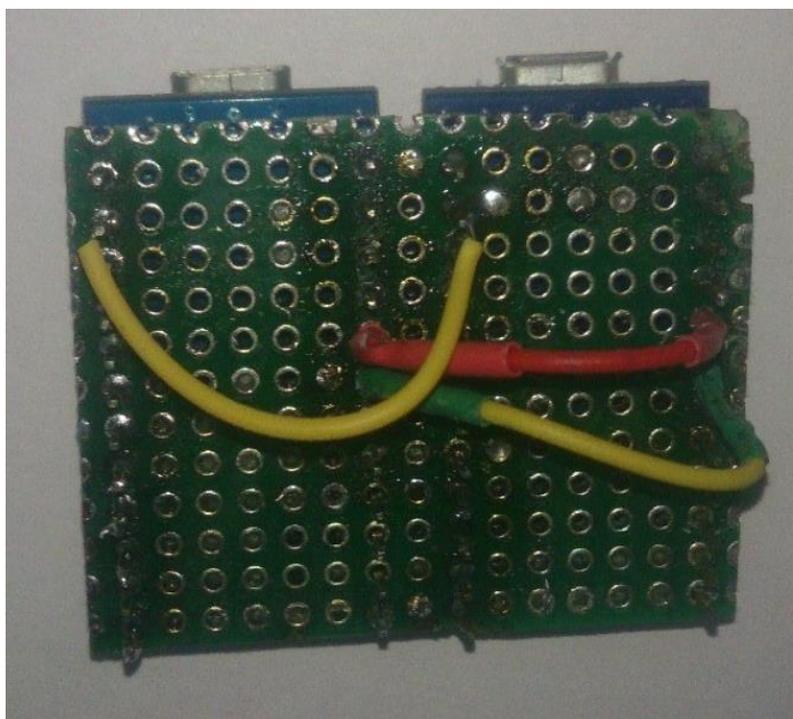
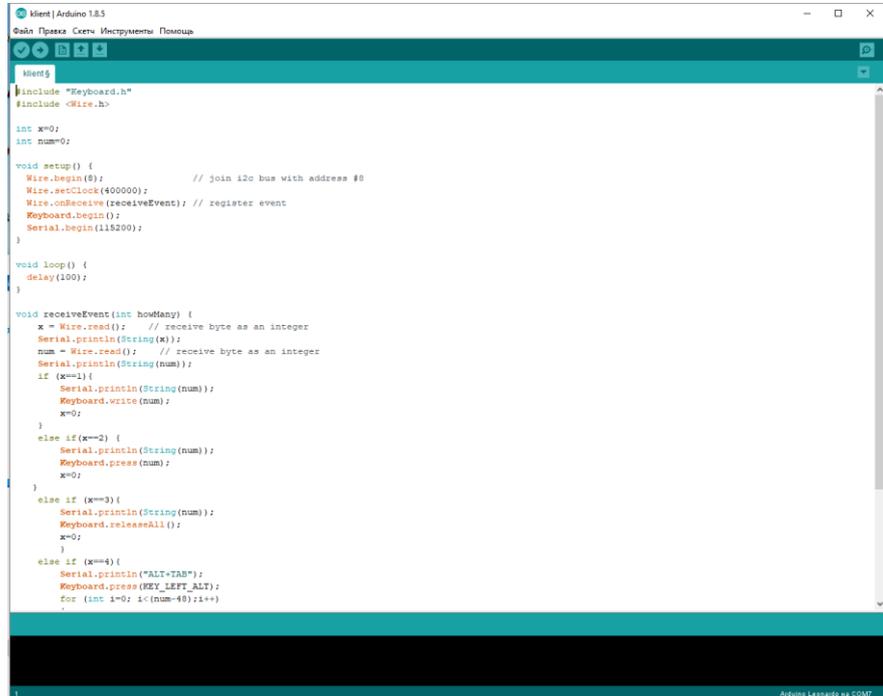


Рисунок 3.15 – Прототип эмулятора клавиатуры (вид снизу)

### 3.4.2 Реализация программной составляющей

Программная часть эмулятора клавиатуры разрабатывалась на языке C++ в среде разработки Arduino IDE. Интерфейс среды разработки представлен на рисунке 3.16.



```
Client | Arduino 1.8.5
Файл Плата Скетч Инструменты Помощь

#include "Keyboard.h"
#include <Wire.h>

int x=0;
int num=0;

void setup() {
  Wire.begin(8); // join i2c bus with address #8
  Wire.setClock(400000);
  Wire.onReceive(receiveEvent); // register event
  Keyboard.begin();
  Serial.begin(115200);
}

void loop() {
  delay(100);
}

void receiveEvent(int howMany) {
  x = Wire.read(); // receive byte as an integer
  Serial.println(String(x));
  num = Wire.read(); // receive byte as an integer
  Serial.println(String(num));
  if (x==1){
    Serial.println(String(num));
    Keyboard.write(num);
    x=0;
  }
  else if(x==2) {
    Serial.println(String(num));
    Keyboard.press(num);
    x=0;
  }
  else if (x==3){
    Serial.println(String(num));
    Keyboard.releaseAll();
    x=0;
  }
  else if (x==4){
    Serial.println("ALT+TAB");
    Keyboard.press(KEY_LEFT_ALT);
    for (int i=0; i<(num-48);i++)
  }
```

Рисунок 3.16 – Интерфейс Arduino IDE

Для создания проекта надо выбрать в меню «файл» → «новый», затем в меню «инструменты» в подразделе «платы» выбрать нужную плату. Если планы нет в списке там же в подразделе «платы» выбрать «менеджер плат» и загрузить нужную.

Программа передающей платы выполняет следующие функции:

1. Принятие сообщения управляющего устройства;
2. Преобразование сообщение в команду с типом нажатия и преобразования символа в номер кнопки, согласно таблице ASCII;
3. Передача команды по i2c.

Код программы подставлен в приложении 3.

Программа выполняющей платы выполняет следующие функции:

1. Принятие команды от передающей платы;
2. Выполнение команды.

Код программы подставлен в приложении И.

Также разработан клиент, передающий команды от управляющего устройства, который включает функции:

1. Нажатие кнопки
2. Зажатие кнопки
3. Сброс кнопок
4. Перенос карты по полю

### 3.5 Тестирование и оценка результатов

После проведенных работ все разработанные программы и устройства были объединены в робота. Структурная схема робота приведена на рисунке 3.17. Блок-схема программы робота представлена в приложении К.

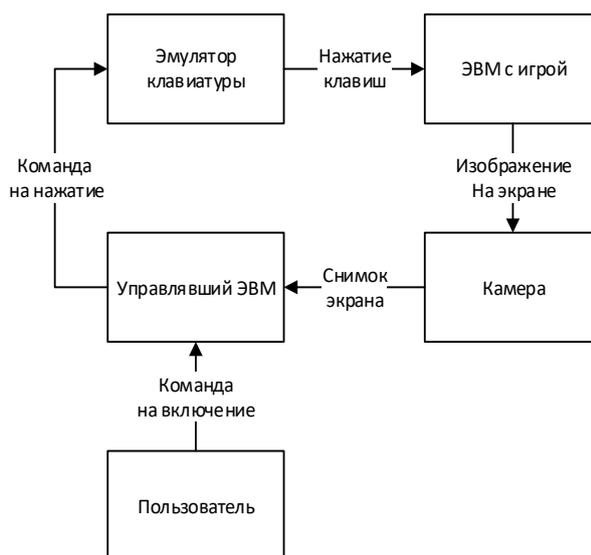


Рисунок 3.17 – Структурная схема робота

Проведем тестирование робота. Для этого робот сыграет по 10 партий при разном количестве мастей в игру, и сведем результаты в таблицу 3.5.

Таблица 3.5 – Результаты игры роботом

№ партии	Число собранных колод	Затраченное время, с.	Число шагов	Число шагов программы
Число мастей 1				
1	8	1427,602	181	73
2	8	1207,991	148	62
3	8	1289,840	146	58
4	8	1145,753	129	51

№ партии	Число собранных колод	Затраченное время, с.	Число шагов	Число шагов программы
5	8	1178,654	132	55
6	8	1356,332	171	68
7	4	754,421	64	36
8	8	1201,804	134	53
9	8	1394,781	175	79
10	8	1234,447	157	62
Итого ср. зн. выигр.	<b>7,6</b>	<b>1219,1625</b>	<b>143,7</b>	<b>59,7</b>
Число мастей 2				
1	1	651,173	95	35
2	0	581,453	77	36
3	8	2216,341	184	66
4	8	2442,521	194	84
5	3	952,124	96	62
6	6	2163,523	179	80
7	8	2086,178	170	77
8	8	2403,067	189	78
9	2	707,910	76	55
10	5	1448,357	125	51
Итого ср. зн.	<b>4,9</b>	<b>1565,2647</b>	<b>138,5</b>	<b>62,4</b>
Число мастей 4				
1	0	1951,565	140	56
2	2	2356,432	168	76
3	1	1847,325	136	63
4	0	1574,523	104	58
5	8	4768,443	479	177
6	2	2878,188	175	85
7	1	2054,656	158	69
8	6	4396,754	403	138
9	1	3171,032	284	102
10	3	3171,032	284	102
Итого ср. зн.	<b>2,4</b>	<b>2816,995</b>	<b>233,1</b>	<b>92,6</b>

Сведем результаты со статистикой людей[13,14] в таблицу 3.6

Таблица 3.6 – Сравнение статистик робота и людей

	Человек, процент выигранных игр (число партий, число выигранных партий)	Стратегия, предложенная Стивом Брауном	Робот, процент выигранных игр
1 масть	98,74% (7542, 7446)	99%	90%
2 масть	49,75% (12196, 6067)	80%	40%
4 масть	28,88% (7437, 2148)	40%	10%

По полученным результатам можно видеть, что по проценту выигранных партий алгоритм уступает людям и стратегии, предложенной Стивом Брауном.

## 4 Финансовый менеджмент, ресурсоэффективность и ресурсосбережение

### 4.1 Предпроектный анализ

#### 4.1.1 Потенциальные потребители результатов исследования

Для анализа потребителей результатов исследования рассмотрен целевой рынок и проведено его сегментирование.

Сегментирование рынка услуг по разработке роботов для образования можно по следующим критериям: виды общеобразовательной организации, область обучения (рисунок 4.1).

		Область обучения		
		ТАУ	Техническое зрение	Машинное обучение
Вид обще- образов- ательн	ВУЗ			
	СУЗ			
	СОШ			

Рисунок 4.1 - Карта сегментирования рынка услуг по разработке роботов:



В приведенной карте сегментирования показано, что ниши на рынке услуг по разработке роботов для образования, где уровень конкуренции низок, являются области обучения техническому зрению и машинному обучению.

#### 4.1.2 Анализ конкурентных технических решений с позиции ресурсоэффективности и ресурсосбережения

Детальный анализ конкурирующих разработок, существующих на рынке, проводится систематически, поскольку рынки пребывают в постоянном движении.

Целесообразно проводить данный анализ с помощью оценочной карты (Таблица 4.1). Для этого оберем разработки:

1. LEGO MINDSTORMS EV3
2. EduTrainer Universal

Критерии для сравнения и оценки ресурсоэффективности и ресурсосбережения, приведенные в таблице 4.1, подбираются, исходя из выбранных объектов сравнения с учетом их технических и экономических особенностей разработки, создания и эксплуатации.

Позиция разработки и конкурентов оценивается по каждому показателю экспертным путем по пятибалльной шкале, где 1 – наиболее слабая позиция, а 5 – наиболее сильная. Веса показателей, определяемые экспертным путем, в сумме должны составлять 1.

Анализ конкурентных технических решений определяется по формуле:

$$K = \sum B_i \cdot \text{Б}_i. \quad (4.1)$$

где K – конкурентоспособность научной разработки или конкурента;  $B_i$  – вес показателя (в долях единицы);  $\text{Б}_i$  – балл i-го показателя.

Таблица 4.1 - Оценочная карта для сравнения конкурентных технических решений

Критерии оценки	Вес критерия	Баллы			Конкурентоспособность		
		Б <sub>ф</sub>	Б <sub>к1</sub>	Б <sub>к2</sub>	К <sub>ф</sub>	К <sub>к1</sub>	К <sub>к2</sub>
1	2	3	4	5	6	7	8
<b>Технические критерии оценки ресурсоэффективности</b>							
1. Повышение производительности труда пользователя	0,1	3	3	3	0,3	0,3	0,3
2. Удобство в эксплуатации (соответствует требованиям потребителей)	0,1	2	4	4	0,2	0,4	0,4
3. Помехоустойчивость	0,01	3	3	4	0,03	0,03	0,04
4. Энергоэкономичность	0,01	4	3	2	0,04	0,03	0,02
5. Надежность	0,1	3	3	5	0,3	0,3	0,5
6. Уровень шума	0,01	5	5	5	0,05	0,05	0,05
7. Безопасность	0,03	5	5	5	0,15	0,15	0,15
8. Потребность в ресурсах памяти	0,02	2	5	3	0,04	0,1	0,06
9. Функциональная мощность (предоставляемые возможности)	0,05	2	5	1	0,1	0,25	0,05
10. Простота эксплуатации	0,1	5	3	2	0,5	0,3	0,2
11. Качество интеллектуального	0,02	4	4	4	0,08	0,08	0,08

Критерии оценки	Вес критерия	Баллы			Конкурентоспособность		
		Б <sub>ф</sub>	Б <sub>к1</sub>	Б <sub>к2</sub>	К <sub>ф</sub>	К <sub>к1</sub>	К <sub>к2</sub>
интерфейса							
12. Возможность подключения в сеть ЭВМ	0,01	5	5	5	0,05	0,05	0,05
<b>Экономические критерии оценки эффективности</b>							
1. Конкурентоспособность продукта	0,1	3	3	3	0,3	0,3	0,3
2. Уровень проникновения на рынок	0,01	1	5	5	0,01	0,05	0,05
3. Цена	0,1	5	2	1	0,5	0,2	0,1
4. Предполагаемый срок эксплуатации	0,1	4	3	4	0,4	0,3	0,4
5. Послепродажное обслуживание	0,1	3	3	3	0,3	0,3	0,3
6. Финансирование научной разработки	0,01	1	3		0,01	0,03	0
7. Срок выхода на рынок	0,01	4	5	5	0,04	0,05	0,05
8. Наличие сертификации разработки	0,01	1	5	5	0,01	0,05	0,05
Итого	1	65	77	69	3,41	3,32	3,15

Уязвимость позиции конкурентов обусловлена ценой, простотой эксплуатации и энергоэкономичностью и это дает возможно занять свою нишу и увеличить определенную долю рынка. Основные конкурентные преимущества разработки цена и простота эксплуатации.

#### **4.1.3 FAST-анализ**

FAST-анализ выступает как синоним функционально-стоимостного анализа. Проведение FAST-анализа предполагает шесть стадий:

##### **1. Выбор объекта FAST-анализа**

В рамках магистерской диссертации в качестве объекта FAST-анализа выступает объект исследования. Объектом исследования в данной является алгоритмы выигрышной стратегии ведение игры и технического зрения, применяемые для игры роботом в пасьянс «Паук».

##### **2. Описание главной, основных и вспомогательных функций, выполняемых объектом**

В рамках данной стадии FAST-анализа объект проанализирован с позиции функционального устройства. Так, при анализе выделены и описаны следующие функции объекта:

- 1) Главная функция
- 2) Основная функция
- 3) Вспомогательная функция

Всю информацию, полученную при выполнении данной стадии, представим в табличной форме (таблица 4.2).

Таблица 4.2 - Классификация функций, выполняемых объектом исследования

Наименование детали	Количество деталей на узел	Выполняемая функция	Ранг функции		
			Главная	Основная	Вспомогательная
Поле	1	Размещение карт		X	
Столбец карт	10	Размещение и перенос карт по правилам	X		
Стопка карт	5	Добавление новых карт		X	
Клавиатура	1	Управление переносом карт			X
Монитор	1	Отображение игрового поля			X

### 3. Определение значимости выполняемых функций объектом

На первом этапе построим матрицу смежности функции (таблица 4.3).

Таблица 4.3 - Матрица смежности

	Размещение карт	Размещение и перенос карт по правилам	Добавление новых карт	Управление переносом карт	Отображение игрового поля
Размещение карт	=	<	<	>	>
Размещение и перенос карт по правилам	>	=	>	>	>
Добавление новых карт	>	<	=	>	>
Управление переносом карт	<	<	<	=	=
Отображение игрового поля	<	<	<	=	=

Второй этап связан с преобразованием матрицы смежности в матрицу количественных соотношений функций (таблица 4.4).

Таблица 4.4 - Матрица количественных соотношений функций

	Размещение карт	Размещение и перенос карт по правилам	Добавление новых карт	Управление переносом карт	Отображение игрового поля	ИТОГО
Размещение карт	1	0,5	0,5	1,5	1,5	5
Размещение и перенос карт по правилам	1,5	1	1,5	1,5	1,5	7
Добавление новых карт	1,5	0,5	1	1,5	1,5	5
Управление переносом карт	0,5	0,5	0,5	1	1	3,5
Отображение игрового поля	0,5	0,5	0,5	1	1	3,5

=24

В рамках третьего этапа определим значимость функций путем деления балла, полученного по каждой функции, на общую сумму баллов по всем функциям. Так, для функции размещение карт относительная значимость равна  $5/24 = 0,21$ ; для функции размещение и перенос карт по правилам –  $7/24 = 0,29$ ; для функции добавление новых карт –  $0,21$ ; для функции управление переносом карт –  $0,145$  и для функции отображение игрового поля –  $0,145$ .

#### 4. Анализ стоимости функций, выполняемых объектом исследования

Задача данной стадии заключается в том, что с помощью специальных методов оценить уровень затрат на выполнение каждой функции. Сделать это возможно с помощью применения нормативного метода. Расчет стоимости функций приведен в таблице 4.5.

Таблица 4.5 - Определение стоимости функций, выполняемых объектом исследования

Наименование детали	Количество деталей на узел	Выполняемая функция	Трудоемкость детали, нормо-ч	Стоимость материала, руб.	Заработная плата, руб.	Себестоимость, руб.
Поле	1	Размещение карт	24	200	-	-
Столбец карт	10	Размещение и перенос карт по правилам	24	200	-	-
Стопка карт	5	Добавление новых карт	24	200	-	-

Наименование детали	Количество деталей на узел	Выполняемая функция	Трудоемкость детали, нормо-ч	Стоимость материала, руб.	Заработная плата, руб.	Себестоимость, руб.
Клавиатура	1	Управление переносом карт	24	-	-	1000
Монитор	1	Отображение игрового поля	24	-	-	5000

## 5. Построение функционально-стоимостной диаграммы объекта и ее анализ

Информация об объекте исследования, собранная в рамках предыдущих стадий, на данном этапе обобщается в виде функционально-стоимостной диаграммы (ФСД) (рисунок 4.2).

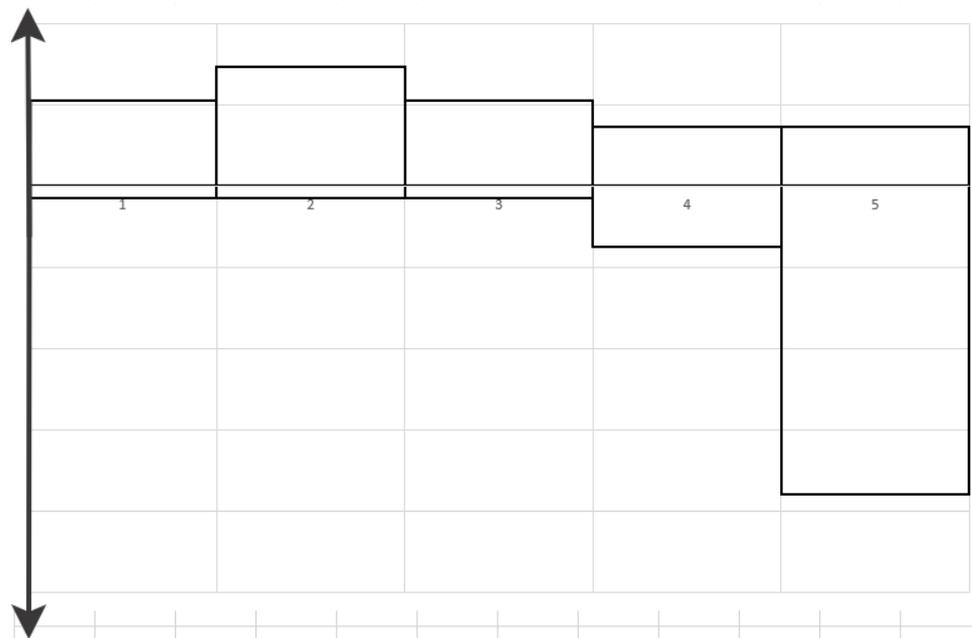


Рисунок 4.2 - Функционально-стоимостная диаграмма

### 4.1.4 Диаграмма Исикава

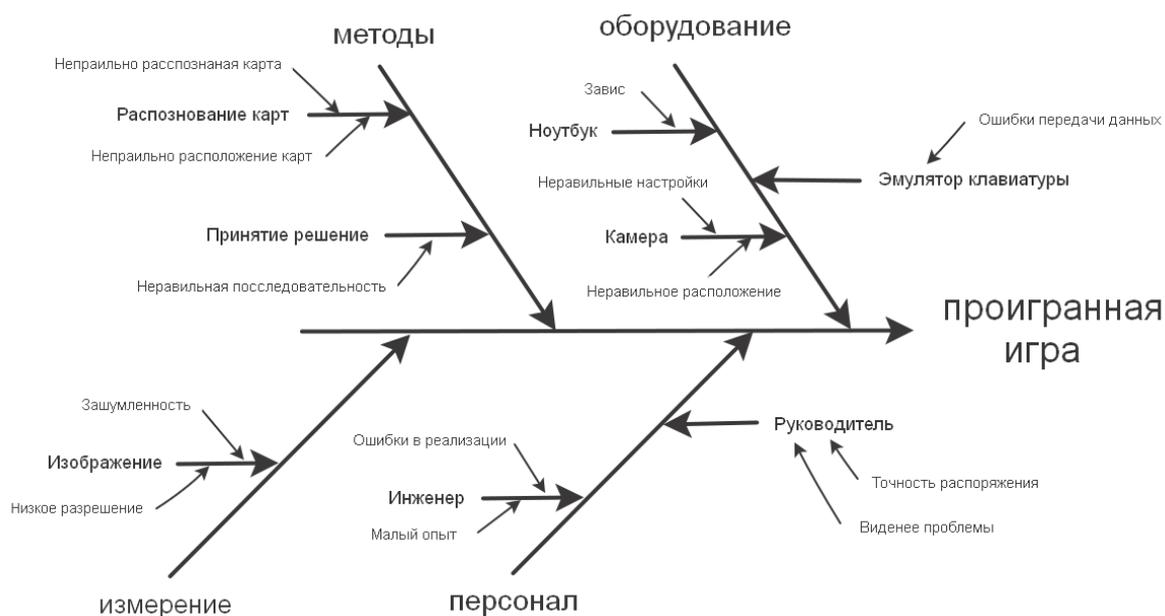
Диаграмма причины-следствия Исикавы (Cause-and-Effect-Diagram) - это графический метод анализа и формирования причинно-следственных связей, инструментальное средство для систематического определения причин проблемы и последующего графического представления.

Область применения диаграммы:

- Выявление причин возникновения проблемы;
- Анализ и структурирование процессов на предприятии;

– Оценка причинно-следственных связей.

Диаграмма Исикавы представлена на рисунке 4.3



Рисунке 4.3 - Общий вид диаграммы Исикавы

#### 4.1.5 SWOT-анализ

SWOT – Strengths (сильные стороны), Weaknesses (слабые стороны), Opportunities (возможности) и Threats (угрозы) – представляет собой комплексный анализ научно-исследовательского проекта. SWOT-анализ применяют для исследования внешней и внутренней среды проекта. Он проводится в несколько этапов.

Первый этап заключается в описании сильных и слабых сторон проекта, в выявлении возможностей и угроз для реализации проекта, которые проявились или могут появиться в его внешней среде.

Второй этап состоит в выявлении соответствия сильных и слабых сторон научно-исследовательского проекта внешним условиям окружающей среды.

В рамках третьего этапа составлена итоговая матрица SWOT-анализа, которая приводится в магистерской диссертации (таблица 4.5).

Таблица 4.5 - Матрица SWOT

	<p><b>Сильные стороны научно-исследовательского проекта:</b> С1. Удобство в эксплуатации (соответствует требованиям потребителей). С2. Функциональная мощность (предоставляемые возможности). С3. Конкурентоспособность продукта. С4. Срок выхода на рынок. С5. Высоко квалифицированный научный труд.</p>	<p><b>Слабые стороны научно-исследовательского проекта:</b> Сл1. Отсутствие у потенциальных потребителей квалифицированных кадров по работе с научной разработкой Сл2. Отсутствие инжиниринговой компании, способной построить производство под ключ Сл3. Большой срок поставок материалов и комплектующий, используемые при проведении научного исследования</p>
<p><b>Возможности:</b> В1. Использование инновационной инфраструктуры-туры ТПУ В2. Использование инфраструктуры ОЭЗ ТВТ Томск В3. Появление дополнительного спроса на новый продукт В4. Снижение таможенных пошлин на сырье и материалы, используемые при научных исследованиях В5. Повышение стоимости конкурентных разработок</p>	<p>Использование инновационной структуры ТПУ позволит повысить конкурентоспособность ПО и ускорить выход на рынок. Так же использование развитой международной инфраструктуры поможет ускорить выход ПО на рынок. Возможно появление дополнительного спроса на новый продукт благодаря использованию высоко квалифицированного научного труда. Благодаря снижению таможенных пошлин на платы возможно повышение конкурентоспособности ПО</p>	<p>Появление дополнительного спроса на новый продукт может привести к отсутствию у потенциальных потребителей квалифицированных кадров. Снижение таможенных пошлин на сырье и материалы, используемые при научных исследованиях может привести к увеличению срока поставки плат, используемых для проведения научного исследования.</p>
<p><b>Угрозы:</b> У1. Отсутствие спроса на новые технологии производства. У2. Развитая конкуренция производителей ПО.</p>	<p>Отсутствие спроса на новые технологии производства может замедлить срок выхода ПО на рынок и понизить квалификацию научного труда. Развитая конкуренция производителей ПО может привести к снижению конкурентоспособности продукта.</p>	<p>Отсутствие спроса на новые технологии производства и высокая стоимость оборудования и плат может привести к отсутствию прототипа научной разработки, отсутствию потенциальных потребителей, необходимого оборудования для проведения испытания ПО.</p>

#### 4.1.6 Оценка готовности проекта к коммерциализации

На какой бы стадии жизненного цикла не находилась научная разработка полезно оценить степень ее готовности к коммерциализации и выяснить уровень собственных знаний для ее проведения (или завершения). Для этого заполним специальную форму, содержащую показатели о степени проработанности проекта с позиции коммерциализации и компетенциям разработчика научного проекта.

При проведении анализа по таблице 4.6 по каждому показателю ставится оценка по пятибалльной шкале.

Таблица 4.6 - Бланк оценки степени готовности научного проекта к коммерциализации

№ п/п	Наименование	Степень проработанности научного проекта	Уровень имеющихся знаний у разработчика
1.	Определен имеющийся научно-технический задел	3	2
2.	Определены перспективные направления коммерциализации научно-технического задела	3	2
3.	Определены отрасли и технологии для предложения на рынке	3	2
4.	Определена товарная форма научно-технического задела для представления на рынок	3	2
5.	Определены авторы и осуществлена охрана их прав	3	2
6.	Проведена оценка стоимости интеллектуальной собственности	2	1
7.	Проведены маркетинговые исследования рынков сбыта	2	1
8.	Разработан бизнес-план коммерциализации научной разработки	1	1
9.	Определены пути продвижения научной разработки на рынок	1	1
10.	Разработана стратегия (форма) реализации научной разработки	1	1
11.	Проработаны вопросы международного сотрудничества и выхода на зарубежный рынок	1	1
12.	Проработаны вопросы использования услуг инфраструктуры поддержки, получения льгот	1	1
13.	Проработаны вопросы финансирования коммерциализации научной разработки	1	1
14.	Имеется команда для коммерциализации научной	2	1
15.	Проработан механизм реализации научного проекта	2	2
	<b>ИТОГО БАЛЛОВ</b>	<b>29</b>	<b>21</b>

Итоговое значение позволяет говорить о мере готовности научной разработки и ее разработчика к коммерциализации. Так, если значение получилось от 75 до 60, то такая разработка считается перспективной, а знания

разработчика достаточными для успешной ее коммерциализации. Если от 59 до 45 – то перспективность выше среднего. Если от 44 до 30 – то перспективность средняя. Если от 29 до 15 – то перспективность ниже среднего. Если 14 и ниже – то перспективность крайне низкая. Полученное значение позволяет говорить, что перспективность выше среднего.

## 4.2 Инициация проекта

**1. Цели и результат проекта.** В данном разделе приведена информация о заинтересованных сторонах проекта, иерархии целей проекта и критериях достижения целей. Информация по заинтересованным сторонам проекта представлены в таблице 4.7.

Таблица 4.7 - Заинтересованные стороны проекта

Заинтересованные стороны проекта	Ожидания заинтересованных сторон
НИТПУ	Рабочий прототип робота

В таблице 4.8 представлена информация о иерархии целей проекта и критериях достижения целей.

Таблица 4.8 - Цели и результат проекта

<b>Цели проекта:</b>	Разработка стратегии ведение игры и создание на основе робота с применением технического зрения
<b>Ожидаемые результаты проекта:</b>	Робота с применением технического зрения и встроенной стратегией
<b>Критерии приемки результата проекта:</b>	Рабочий прототип
<b>Требования к результату проекта:</b>	<b>Требование:</b>
	Точное распознавание карт
	Возможность выиграть партию

## 2. Организационная структура проекта.

На данном этапе работы решены следующие вопросы: кто будет входить в рабочую группу данного проекта, определены роли каждого участника в данном проекте, а также прописаны функции, выполняемые каждым из

участников и их трудозатраты в проекте. Эта информация представлена в табличной форме (таблица 4.9).

Таблица 4.9 - Рабочая группа проекта

№ п/п	ФИО, основное место работы, должность	Роль в проекте	Функции	Трудозатраты, час.
1	Липатов Д.С., НИТПУ ОАиР, магистр	Инженер	Реализация проекта	
2	Замятин С.В., НИТПУ ОАиР, доцент	Руководитель	Консультирование, контроль проекта	

### 4.3 Планирование управления научно-техническим проектом

#### 4.3.1 Иерархическая структура работ проекта

Иерархическая структура работ (ИСР) – детализация укрупненной структуры работ. В процессе создания ИСР структурируется и определяется содержание всего проекта. На рисунке 4.4 представлена иерархическая структура работ по проекту.



Рисунке 4.4 - Иерархическая структура работ по проекту

### 4.3.2 Контрольные события проекта

В рамках данного раздела определены ключевые события проекта, определены их даты и результаты, которые получены по состоянию на эти даты. Эту информацию сведены в таблицу 4.11.

Таблица 4.11 - Контрольные события проекта

№ п/п	Контрольное событие	Дата	Результат
1	Выбор метода реализации выигрышной стратегии	17.03	Выбранный метод
2	Выбор метода распознавания на изображения	22.03	Выбранный метод
3	Выбор программного обеспечения, библиотеки, языка программирования	23.03	Выбранный язык программирования
5	Разработка и реализация алгоритма выигрышной стратегии ведение игры	07.04	Алгоритм выигрышной стратегии
6	Разработка и реализация алгоритма распознавания карт на изображении	22.04	Алгоритм распознавания
7	Разработка робота без устройств ввода вывода	2.05	Робот без периферийных устройств
8	Проектирование и интеграция устройств ввода вывода в робота	10.05	Робот с устройствами ввода вывода
9	Оформление расчетно-пояснительной записки	27.05	Расчетно-пояснительная записка
10	Оформление графического материала	31.05	Графический материал

### 4.3.3 План проекта

В рамках планирования научного проекта был построен календарный план проекта в виде линейного графика в таблице (таблица 4.12) и его иллюстрация работы в виде диаграммы Ганта (таблица 4.13). Диаграммы Ганта строилась с разбивкой по месяцам и декадам (10 дневки) который охватывает весь период времени выполнения научного проекта.

Таблица 4.12 - Календарный план проекта

Код	Название	Длительность, дни	Дата начала работ	Дата окончания работ	Состав участников
1	Постановка целей и задач	1	12.03.	12.03.	НР
2	Исследование методов реализации выигрышной стратегии и выбор оптимального	5	13.03.	17.03	И НР

Код	Название	Длительность, дни	Дата начала работ	Дата окончания работ	Состав участников
3	Исследование методов распознавания на изображении	5	18.03	22.03	И, НР
4	Выбор программного обеспечения, библиотеки, языка программирования	1	23.03	23.03	И, НР
5	Разработка и реализация алгоритма выигрышной стратегии ведение игры	15	24.03	07.04	И
6	Разработка и реализация алгоритма распознавания карт на изображении	15	08.04	22.04	И
7	Разработка робота без устройств ввода вывода	10	23.04	2.05	И
8	Проектирование и интеграция устройств ввода вывода в робота	8	3.05	10.05	И
9	Тестирование	10	11.05	20.05	И, НР
10	Оформление расчетно-пояснительной записки	7	21.05	27.05	И
11	Оформление графического материала	4	28.05	31.05	И
12	Подведение итогов	3	01.06	03.06	И, НР
Итого:		84	12.03	04.06	
Инженер		83			
Научный руководитель		25			

Перечисленные работы в таблице 4.12, 4.13 были выполнены

специалистами:

- 1) И ■ – инженер – исполнитель ВКР – Липатов Дмитрий Сергеевич;
- 2) НР ■ – научный руководитель – доцент отделения автоматизации и робототехники Замятин Сергей Владимирович.

Таблица 4.13 – Диаграмма Ганта

Код	Вид работ	Исполнители	Тк, кол, дн.	Продолжительность выполнения работ, месяц												
				3			4			5			6			
				2	3	1	2	3	1	2	3	1	2	3		
1	Постановка целей и задач	НР	1	■												
2	Исследование методов реализации выигрышной стратегии и выбор оптимального	И	5	■												
НР		■														
3	Исследование методов распознавания на изображения	И	5		■											
НР		■														
4	Выбор программного обеспечения, библиотеки, языка программирования	И	1		■											
НР		■														
5	Разработка и реализация алгоритма выигрышной стратегии ведение игры	И	15			■										
6	Разработка и реализация алгоритма распознавания карт на изображения	И	15				■									
8	Разработка робота без устройств ввода вывода	И	10						■							
9	Проектирование и интеграция устройств ввода вывода в робота	И	8							■						
10	Тестирование	И	10								■					
НР		■														
11	Оформление расчетно-пояснительной записки	И	7									■				
12	Оформление графического материала	И	4										■			
13	Подведение итогов	И	3											■		
НР		■														

#### 4.3.4 Бюджет научного исследования

##### 4.3.4.1 Расчет затрат на сырье, материалы, покупные изделия и полуфабрикаты

В эту статью включены затраты на приобретение всех видов материалов, комплектующих изделий и полуфабрикатов, необходимых для выполнения

работ по данной теме. Количество потребляемых материальных ценностей определяется по нормам расхода.

Расчет стоимости материальных затрат производился по действующим прейскурантам или договорным ценам. В стоимость материальных затрат включались транспортно-заготовительные расходы (3 – 5 % от цены). В эту же статью включались затраты на оформление документации (канцелярские принадлежности, тиражирование материалов). Результаты по данной статье представлены в таблице 4.14.

Таблица 4.14 – Материальные затраты

Наименование материалов	Цена за ед., руб.	Количество	Сумма, руб.
Блокнот	40	2 шт.	80
Бумага для принтера формата А4	150	2 уп.	300
Ручка шариковая	10	4 шт.	40
Карандаш	10	2 шт.	20
Стирательная резинка	5	2 шт.	10
<b>Итого:</b>			<b>450</b>

Расходы на материалы составили

$$З_m = 450 \text{ рублей.}$$

#### 4.3.4.2 Расчет затрат на специальное оборудование для научных работ

В данную статью включается все затраты, связанные с приобретением специального оборудования, необходимого для проведения работ по конкретной теме. Определение стоимости спецоборудования произведены по действующим прейскурантам, а в ряде случаев по договорной цене.

Все расчеты по приобретению спецоборудования и оборудования, имеющегося в организации, но используемого для каждого исполнения конкретной темы, сводятся в таблице 4.15.

Таблица 4.15 – Материальные затраты на приобретение спецоборудования для научных работ

Наименование материалов	Цена за ед., руб.	Количество	Сумма, руб.
Arduino micro pro	250	2 шт.	500
Logitech c270	1200	1 шт.	1200
<b>Итого:</b>			<b>1700</b>

Расходы на приобретение спецоборудования для научных работ:

$$Z_{CO} = 1700 \text{ рублей.}$$

#### 4.3.4.3 Основная заработная плата исполнителей темы

В настоящую статью включена основная заработная плата научных и инженерно-технических работников, участвующих в выполнении работ по данной теме. Величина расходов по заработной плате определяется исходя из трудоемкости выполняемых работ и действующей системы окладов и тарифных ставок. В состав основной заработной платы включается премия, выплачиваемая ежемесячно из фонда заработной платы в размере 20 –30 % от тарифа или оклада. Расчет основной заработной платы сводится в таблице 4.16.

$$\text{Дневная з/плата} = \frac{\text{Месячный оклад}}{25,17 \text{ дней}} \quad (4.2)$$

Среднедневная заработная плата рассчитывается по формуле:

Расчеты затрат на основную заработную плату приведены в таблице 4.16. При расчете учитывалось, что в году 302 рабочих дня и, следовательно, в месяце 25,17 рабочих дня. Затраты времени на выполнение работы по каждому исполнителю брались из таблицы 4.13. Также был принят во внимание коэффициент, учитывающий коэффициент по премиям  $K_{ПР} = 0,3$  и районный коэффициент  $K_{РК} = 0,3$  ( $K = 1,3 * 1,3 = 1,69$ ).

Таблица 4.16 – Затраты на основную заработную плату

Исполнитель	Оклад, руб./мес.	Среднедневная ставка, руб./день	Затраты времени, дни	Коэффициент	Фонд з/платы, руб.
НР	23 264,86	936,97	25	1,69	39586,98
И	1854	73,66	83	1,69	10332,16
<b>Итого:</b>					<b>49919,14</b>

Таким образом, затраты на основную заработную плату составили

$$Z_{OCH} = 49919,14 \text{ руб.}$$

#### 4.3.4.4 Отчисления во внебюджетные фонды (страховые отчисления)

В данной статье расходов отражаются обязательные отчисления по установленным законодательством Российской Федерации нормам органам

государственного социального страхования (ФСС), пенсионного фонда (ПФ) и медицинского страхования (ФФОМС) от затрат на оплату труда работников.

Величина отчислений во внебюджетные фонды определены исходя из следующей формулы:

$$Z_{\text{внеб}} = k_{\text{внеб}} \cdot (Z_{\text{осн}} + Z_{\text{доп}}), \quad (4.3)$$

где  $k_{\text{внеб}}$  – коэффициент отчислений на уплату во внебюджетные фонды (пенсионный фонд, фонд обязательного медицинского страхования и пр.).

На 2014 г. в соответствии с Федеральным законом от 24.07.2009 №212-ФЗ установлен размер страховых взносов равный 30%. На основании пункта 1 ст.58 закона №212-ФЗ для учреждений, осуществляющих образовательную и научную деятельность в 2014 году водится пониженная ставка – 30%. Отчисления во внебюджетные фонды представлены в табличной форме (табл. 4.17).

Таблица 4.17 – Отчисления во внебюджетные фонды

Исполнитель	Основная заработная плата, руб.	Дополнительная заработная плата, руб.
НР	39586,98	–
И	10332,16	–
Коэффициент отчислений во внебюджетные фонды	$k_{\text{внеб}} = 30\%$	
<b>Итого:</b>	<b>14975,74</b>	

$$Z_{\text{внеб}} = 14975,74$$

#### 4.3.4.5 Накладные расходы

Накладные расходы учитывают прочие затраты организации, не попавшие в предыдущие статьи расходов: печать и ксерокопирование материалов исследования, оплата услуг связи, электроэнергии, почтовые и телеграфные расходы, размножение материалов и т.д. Их величина определяется по следующей формуле:

$$Z_{\text{накл}} = (\text{сумма статей } 1 \div 4) \cdot k_{\text{нр}}, \quad (4.4)$$

где  $k_{\text{нр}}$  – коэффициент, учитывающий накладные расходы.

Величину коэффициента накладных расходов берется в размере 16%.

$$Z_{\text{накл}} = (49919,14 + 14975,74 + 450 + 1700) * 0,16 = 10727,18$$

#### 4.3.4.6 Формирование бюджета затрат научно-исследовательского проекта

Рассчитанная величина затрат научно-исследовательской работы (темы) является основой для формирования бюджета затрат проекта, который при формировании договора с заказчиком защищается научной организацией в качестве нижнего предела затрат на разработку научно-технической продукции.

Определение бюджета затрат на научно-исследовательский проект по каждому варианту исполнения приведен в табл. 4.18.

Таблица 4.18 – Расчёт бюджета затрат НИИ

Наименование статьи	Сумма, руб.	Примечание
1. Материальные затраты НИИ	450	Пункт 3.4.1
2. Затраты на специальное оборудование для научных работ	1700	Пункт 3.4.2
3. Затраты по основной заработной плате исполнителей темы	49919,14	Пункт 3.4.3
4. Отчисления во внебюджетные фонды	14975,74	Пункт 3.4.4
5. Накладные расходы	10727,18	16 % от суммы ст. 1-4
6. Бюджет затрат НИИ	77772,06	Сумма ст. 1- 5

#### 4.3.5 Организационная структура проекта

В практике используется несколько базовых вариантов организационных структур: функциональная, проектная, матричная. Проектная организационная структура научного проекта приведен на рисунке 4.5.



Рисунке 4.5 - Проектная организационная структура научного проекта

#### **4.4 Определение ресурсной, финансовой, бюджетной, социальной и экономической эффективности исследования**

Выполнение научно-исследовательских работ оценивается уровнями достижения экономического, научного, научно-технического и социального эффектов. Научный эффект характеризует получение новых научных знаний и отображает прирост информации, предназначенной для внутринаучного потребления. Научно-технический эффект характеризует возможность использования результатов в других проектах и обеспечивает получение информации, необходимой для создания новой техники. Экономический эффект характеризуется в частности выраженной в стоимостных показателях экономией живого общественного труда, а также в возможности применения полученных знаний для создания новых разработок.

Экономическая эффективность проекта обусловлена возрастающей необходимостью обучению алгоритмам компьютерного зрения и машинного обучения. Конечная система в виде робота с возможностью распознавание карт и игры в пасьянс является актуальной на сегодняшний день. Использование подобного устройства в образовательной сфере позволяет решать такие задачи, как обучение алгоритмам распознавания и методам машинного обучение.

Подводя итог вышесказанному, экономический эффект при реализации дает использование свободно распространяемого программного обеспечение.

##### **4.4.1. Оценка научно-технического уровня НИР**

Научно-технический уровень характеризует, в какой мере выполнены работы и обеспечивается научно-технический прогресс в данной области. Для оценки научной ценности, технической значимости и эффективности, планируемых и выполняемых НИР, используется метод бальных оценок. Бальная оценка заключается в том, что каждому фактору по принятой шкале присваивается определенное количество баллов. Обобщенную оценку проводят

по сумме баллов по всем показателям или рассчитывают по формуле. На этой основе делается вывод о целесообразности НИР.

Сущность метода заключается в том, что на основе оценок признаков работы определяется коэффициент ее научно-технического уровня по формуле:

$$K_{НТУ} = \sum_{i=0}^3 R_i \cdot n_i. \quad (4.5)$$

где  $K_{НТУ}$  – коэффициент научно-технического уровня;

$R_i$  – весовой коэффициент  $i$ -го признака научно-технического эффекта;

$n_i$  – количественная оценка  $i$ -го признака научно-технического эффекта, в баллах.

Таблица 4.19 – Весовые коэффициенты признаков НТУ

Признак НТУ	Примерное значение весового коэф-та $n_i$	
1. Уровень новизны	Систематизируются и обобщаются сведения, определяются пути дальнейших исследований	0,4
2. Теоретический уровень	Разработка алгоритма	0,1
3. Возможность реализации	Время реализации в течение первых лет	0,5

Таблица 4.20 – Баллы для оценки уровня новизны

Уровень новизны	Характеристика уровня новизны	Баллы
Принципиально новая	Новое направление в науке и технике, новые факты и закономерности, новая теория, вещество, способ	8 – 10
Новая	По-новому объясняются те же факты, закономерности, новые понятия дополняют ранее полученные результаты	5 – 7
Относительно новая	Систематизируются, обобщаются имеющиеся сведения, новые связи между известными факторами	2 – 4
Не обладает новизной	Результат, который ранее был известен	0

Таблица 4.21 – Баллы значимости теоретических уровней

Теоретический уровень полученных результатов	Баллы
1. Установка закона, разработка новой теории	10
2. Глубокая разработка проблемы, многоспектральный анализ, взаимодействия между факторами с наличием объяснений	8
3. Разработка способа (алгоритм, программа и т. д.)	6
4. Элементарный анализ связей между фактами (наличие гипотезы, объяснения версии, практических рекомендаций)	2
5. Описание отдельных элементарных факторов, изложение наблюдений, опыта, результатов измерений	0,5

Таблица 4.22 – Возможность реализации научных, теоретических результатов по времени и масштабам

Время реализации	Баллы
В течение первых лет	10
От 5 до 10 лет	4
Свыше 10 лет	2

Результаты оценок признаков научно-технического уровня приведены в таблице 4.23.

Таблица 4.23 – Количественная оценка признаков НИОКР

Признак научно-технического эффекта НИР	Характеристика признака НИОКР	$R_i$
Уровень новизны	Систематизируются, обобщаются имеющиеся сведения, новые связи между известными факторами	3
Теоретический уровень	Разработка способа (алгоритм, программа мероприятий, устройство, вещество и т.п.)	6
Возможность реализации	течение первых лет	10

Обоснование оценки признаков НИОКР приводится в таблице 4.24.

Таблица 4.24 – Оценки научно-технического уровня НИР

Фактор НТУ	Значимость	Уровень фактора	Выбранный балл	Обоснование выбранного балла
Уровень новизны	0,4	Относительно новая	3	Получение выигрышной стратегии игры в пасьянс «Паук»
Теоретический уровень	0,1	Разработка способа	6	Разработана и реализован алгоритм, играющий в пасьянс «Паук»
Возможность реализации	0,5	В течение первых лет	9	Разработан прототип что позволяет быструю реализацию

Исходя из оценки признаков НИОКР, показатель научно-технического уровня для данного проекта составил:

$$K_{НТУ} = 0,4 \cdot 3 + 0,1 \cdot 6 + 0,5 \cdot 9 = 6,3$$

Таблица 4.25 – Оценка уровня научно-технического эффекта

Уровень НТЭ	Показатель НТЭ
Низкий	1-4
Средний	4-7
Высокий	8-10

Таким образом, исходя из данных в таблице 4.25, проект «Стратегия ведения игры роботом с притеснением технического зрения» имеет средний уровень научно-технического эффекта.

## **5. Социальная ответственность**

### **Введение**

В данной ВКР представлена вариант разработки алгоритма и реализация на его основе работа, который играет пасьянс «Паук». В основе работы устройства лежат алгоритмы компьютерного зрения, с помощью которых в реальном времени производится оценка расположения игральных карт и разработанный алгоритм стратегии игры. При разработке использовалось ПО: ОС Lubuntu 16.04, IDE Code Blocks 13.12 и Arduino IDE, Библиотека OpenCV 3.2.

В текущем разделе указаны основные вредные и опасные факторы рабочей зоны, их анализ и способы защиты от них, аспекты охраны окружающей среды, защиты от чрезвычайных ситуаций, а также правовые и организационные вопросы обеспечения безопасности.

### **5.1 Производственная безопасность**

#### **5.1.1 Анализ выявленных вредных факторов при разработке и эксплуатации проектируемого решения**

Согласно номенклатуре, опасные и вредные факторы по ГОСТ 12.0.003-74 делятся на следующие группы: физические; химические; психофизиологические; биологические.

Перечень опасных и вредных факторов, влияющих на персонал в заданных условиях деятельности, представлен в таблице 5.1.

Эти факторы могут влиять на состояние здоровья, привести к травмоопасной или аварийной ситуации, поэтому следует установить эффективный контроль за соблюдением норм и требований, предъявленных к их параметрам.

Таблица 5.1 – Перечень опасных и вредных факторов технологии производства

Источник фактора, наименование видов работ	Факторы		Нормативные документы
	Вредные	Опасные	
Управление механизмами поста управления, работа с ПЭВМ; Выполнение визуальных осмотров всех основных и вспомогательных механизмов до начала их использования при выполнении работ;	Повышенная температура; Повышенная напряженность зрения; Повышенная напряженность труда в течение смены; Отсутствие или недостаток естественного света; Электромагнитные излучения; Повышенный уровень шума.	Электрический ток. Короткое замыкание Статическое электричество	Гигиенические требования к микроклимату производственных помещений СанПиН 2.2.4-548-96; Нормы естественного и искусственного освещения предприятий, СНиП 23-05-95; Допустимые уровни шумов в производственных помещениях. ГОСТ 12.1.003-83. ССБТ; Гигиенические требования к персональным электронно-вычислительным машинам и организации работы, СанПиН 2.2.2/2.4.1340-03; Защитное заземление, зануление, ГОСТ 12.1.030–81 ССБТ.

#### 5.1.1.1 Освещение

Освещение рабочего места – важнейший фактор создания нормальных условий труда. Освещение делится на естественное, искусственное и совмещенное. Совмещенное сочетает оба вида освещения.

Рекомендуемые соотношения яркостей в поле зрения следующие:

- между рабочими поверхностями не должно превышать 1:3 – 1:5;
- между рабочими поверхностями и поверхностями стен и оборудования – 1:10.

Освещенность на поверхности стола в зоне размещения рабочего документа должна быть 300 - 500 лк (СНиП 23-05-95, СанПиН 2.2.2/2.4.1340-03). Освещение не должно создавать бликов на поверхности экрана. Освещенность поверхности экрана не должна быть более 300 лк. Следует ограничивать прямую блескость от источников освещения, при этом яркость светящихся поверхностей, должна быть не более 200 кд/м<sup>2</sup>. Показатель

ослепленности для источников общего искусственного освещения в производственных помещениях должен быть не более 20.

Согласно СНИП 23-05-95 нормы на освещение для оператора поста управления берутся для производственных помещений. Эти нормы представлены в таблице 5.2.

Таблица 5.2 – Нормы на освещение для оператора

Характер зрительной работы	Разряд зрительной работы	Подразряд зрительной работы	Искусственное освещение		Естественное освещение КЕО е <sub>н</sub> , % при боковом
			Освещенность при системе общего освещения, лк	Коэффициент пульсации, Кп, %	
Различение объектов высокой точности	Б	1	300	15	1,0

Для определения приемлемого уровня освещенности в помещении необходимо: определить требуемый для операторов уровень освещенности внешними источниками света; если требуемый уровень освещенности не приемлем для других операторов надо найти способ сохранения требуемого контраста изображения другими средствами.

### 5.1.1.2 Вибрация

Один из важнейших аспектов охраны труда – вибрация, которую испытывают на себе сотрудники во время исполнения своих трудовых функций.

Источниками вибрации являются: возвратно-поступательно движущиеся детали; неуравновешенность вращающихся масс; ударный характер нагрузки. К вибрирующему оборудованию относится оборудование составляющие не менее 20% допустимых санитарными нормами величин.

Общая вибрация передается через опорные поверхности телу сидящего или стоящего человека. Локальная вибрация передается через руки человека. Вибрация, воздействующая на ноги сидящего человека и предплечья, контактирующие с вибрирующими поверхностями рабочих столов, также относится к локальной вибрации.

С 2017 года уровень вибрации на рабочем месте устанавливает Раздел IV СанПиН 2.2.4.3359-16.

Способы защиты от негативного воздействия вибрации:

- уменьшить вибрации источника.
- уменьшать вибрацию по пути распространения.
- применять дистанционное управление и автоматизировать процессы.
- Негативное воздействие вибрации уменьшается при оптимальном чередовании, режимов труда и отдыха.

### 5.1.1.3 Микроклимат

Значимым физическим фактором является микроклимат рабочей зоны. Температура, относительная влажность и скорость движения воздуха влияют на теплообмен и необходимо учитывать их комплексное воздействие. Нарушение теплообмена вызывает тепловую гипертермию, или перегрев.

Оптимальные нормы температуры, относительной влажности и скорости движения воздуха производственных помещений для работ, производимых сидя и не требующих систематического физического напряжения (категория Ia), приведены в таблице 5.3, в соответствии с СанПиНом 2.2.2/2.4.1340-03 и СанПиН 2.2.4.548-96.

Таблица 5.3 – Нормы температуры, относительной влажности и скорости движения воздуха

Период года	Категория работы	Температура, С	Относительная влаж. воздуха, %	Скорость движения воздуха, не более м/с
Холодный	Ia	22-24	40-60	0,1
Теплый	Ia	23-25	40-60	0,1

Допустимые микроклиматические условия установлены по критериям допустимого теплового и функционального состояния человека на период 8-часовой рабочей смены. Допустимые величины показателей микроклимата на рабочих местах представлены в таблице 5.4.

Таблица 5.4 – Допустимые величины показателей микроклимата

Период года	Категория работы	Температура воздуха, °С	Относительная влаж. воздуха, %	Скорость движения воздуха, не более м/с
-------------	------------------	-------------------------	--------------------------------	---

Холодный	Ia	20-25	15-75	0,1
Теплый	Ia	21-28	15-75	0,1-0,2

Для обеспечения установленных норм микроклиматических параметров и чистоты воздуха на рабочих местах и в помещениях применяют вентиляцию. В холодное время года предусматривается система отопления. Для отопления помещений используются водяные системы центрального отопления.

#### 5.1.1.4 Шум

В производственных условиях имеют место шумы различной интенсивности и частотного спектра, которые генерируются источниками шумов. Для исследуемого объекта основными источниками шумов являются производственное оборудование и оборудование поста управления.

ПДУ шума для объектов типа поста управления нормируются ГОСТ 12.1.003-83 и СН 2.2.4/2.1.8.562–96. Значения ПДУ согласно этим документам представлены в таблице 5.4.

Таблица 5.4 – Нормированные значения уровня звукового давления

Рабочие места	Уровни звукового давления (ДБ) в октавных полосах со среднегеометрическими частотами, Гц									Уровни звука и эквивалентные уровни звука, дБА
	31,5	63	125	250	500	1000	2000	4000	8000	
ПУ	86	71	61	54	49	45	42	40	38	50

Для оценки соблюдения ПДУ шума необходим производственный контроль. В случае превышения уровней необходимы организационно-технические мероприятия по защите от действия шума.

#### 5.1.1.5 Электромагнитные излучения

Электромагнитным излучением называется излучение, прямо или косвенно вызывающее ионизацию среды. Контакт с электромагнитными излучениями представляет серьезную опасность для человека. Нормы электромагнитных полей, создаваемых ПЭВМ приведены в таблице 5.5 и таблице 5.6, в соответствии с СанПиНом 2.2.2/2.4.1340-03.

Таблица 5.5 – Временные допустимые ЭМП, создаваемых ПЭВМ

Наименование параметров	ВДУ ЭМП	
Напряженность электрического поля	В диапазоне частот 5 Гц – 2 кГц	25 В/м
	В диапазоне частот 2 кГц – 400 кГц	2,5 В/м

Плотность магнитного потока	В диапазоне частот 5 Гц – 2 кГц	250 нТл
	В диапазоне частот 2 кГц – 400 кГц	25 нТл
Электростатический потенциал экрана видеомонитора		500 В

Таблица 5.6 – Временные допустимые уровни ЭМП, создаваемых ПЭВМ на рабочих местах

Наименование параметров		ВДУ
Напряженность электрического поля	в диапазоне частот 5 Гц - 2 кГц	25 В/м
	в диапазоне частот 2 кГц - 400 кГц	2,5 В/м
Плотность магнитного потока	в диапазоне частот 5 Гц - 2 кГц	250 нТл
	в диапазоне частот 2 кГц - 400 кГц	25 нТл
Напряженность электростатического поля		

Для оценки соблюдения уровней необходим производственный контроль. В случае превышения уровней необходимы организационно-технические мероприятия.

#### 5.1.1.6. Психофизиологические факторы

Наиболее эффективные средства предупреждения утомления при работе на производстве – это средства, нормализующие активную трудовую деятельность человека. На фоне нормального протекания производственных процессов одним из важных физиологических мероприятий против утомления является правильный режим труда и отдыха (СанПиН 2.2.2/2.4.1340-03). Существуют следующие меры по снижению влияния монотонности:

- необходимо применять оптимальные режимы труда и отдыха в течение рабочего дня;
- соблюдать эстетичность производства.

Для уменьшения физических нагрузок организма во время работы рекомендуется использовать специальную мебель с возможностью регулировки под конкретные антропометрические данные.

#### 5.1.2. Анализ выявленных опасных факторов при разработке и эксплуатации проектируемого решения

В условиях современного интенсивного использования ЭВМ важное значение имеет изучение психофизиологических особенностей и возможностей

человека с целью создания вычислительной техники, обеспечивающей максимальную производительность труда и сохранение здоровья людей.

При внедрении улучшенной системы управления технологическим процессом важную роль играет планировка рабочего места. Она должна соответствовать правилам охраны труда и удовлетворять требованиям удобства выполнения работы, экономии энергии и времени оператора.

Основным документом, определяющим условия труда на персональных ЭВМ санитарные нормы и правила СанПиН 2.2.2/2.4.1340-03.

В Правилах указаны основные требования к помещениям, микроклимату, шуму и вибрации, освещению помещений и рабочих мест, организации и оборудованию рабочих мест.

#### **5.1.2.1 Электрический ток**

Степень опасного воздействия на человека электрического тока зависит от:

- рода и величины напряжения и тока;
- частоты электрического тока;
- пути прохождения тока через тело человека;
- продолжительности воздействия на организм человека;
- условий внешней среды.

Согласно ПУЭ пост управления №8 по степени опасности поражения электрическим током можно отнести к классу помещений без повышенной опасности. Основными мероприятиями по защите от электропоражения являются:

- использование изоляции в корпусах оборудования;
- применение средств защиты от поражения электрическим током;
- защитного заземления;
- защитного отключения;
- использование устройств бесперебойного питания.

Технические способы и средства применяют отдельно или в сочетании друг с другом так, чтобы обеспечивалась оптимальная защита.

### **5.1.2.2 Короткое замыкание**

Короткое замыкание с точки зрения электротехники представляет собой явление состоящей из нескольких проводов, крайне незначительно, и его вполне можно сопоставить с сопротивлением самих проводов.

Это приведет к тому, что электрическая цепь разорвется намного раньше, чем произойдет критическое увеличение температуры проводов.

Защита от короткого замыкания включает в себя целый комплекс мер, исходным пунктом в которых является профилактика повреждений линий электропередач и оборудования. Кроме того, чтобы предотвратить возникновение пожара, используют специальные приборы – плавкиеставки, которые при замыкании сгорают и размыкают электрическую цепь.

Мощность короткого замыкания зависит от множества факторов, главным из которых является сила тока в цепи

### **5.1.2.3 Статическое электричество**

Статическое электричество - совокупность явлений, связанных с образованием и сохранением свободного электрического заряда на поверхности и в объеме диэлектрических и полупроводниковых веществ. Причиной возникновения статического электричества являются процессы электризации.

Большую опасность разряды статического электричества и искрение в электрических цепях создают в условиях повышенного содержания горючих газов или горючих паров и пыли в помещениях. Меры защиты:

1. снижение силового воздействия
2. снижение скоростей перемещения слоёв сыпучих материалов и жидкостей
6. увеличение относительной влажности выше 65%
7. заземление оборудования
8. ионизация воздуха вблизи мест образования зарядов
9. токопроводящая обувь, полы, обивки стульев

## 10. легкосъёмные токопроводящие браслеты

### 5.1.3. Рекомендации по минимизации влияний на работника на примере освещения

Расчет системы искусственного освещения проводится для прямоугольного помещения, размерами: длина  $A = 4$  (м), ширина  $B = 3$  (м), высота  $H = 2,4$  (м), количество ламп  $N = 4$  (шт).

Согласно отраслевым нормам освещенности уровень рабочей поверхности над полом составляет  $0,8$  (м) и установлена минимальная норма освещенности  $E = 300$  (Лк).

Световой поток лампы накаливания или группы люминесцентных ламп светильника определяется по формуле:

$$\Phi = \frac{E_n \cdot K_z \cdot Z \cdot 100}{n \cdot \eta}, \quad (5.1)$$

где:  $E_n$  – нормируемая минимальная освещённость по СНиП 23-05-95, (Лк);  $S$  – площадь освещаемого помещения, (м<sup>2</sup>);  $K_z$  – коэффициент запаса, учитывающий загрязнение светильника, пыли;  $Z$  – коэффициент неравномерности освещения. Для люминесцентных ламп при расчётах берётся равным  $Z = 1,1$ ;  $n$  – число светильников;  $\eta$  – коэффициент использования светового потока, (%);  $\Phi$  – световой поток, излучаемый светильником.

Коэффициент использования светового потока показывает, какая часть светового потока ламп попадает на рабочую поверхность. Он зависит от индекса помещения  $i$ , типа светильника, высоты светильников над рабочей поверхностью  $h$  и коэффициентов отражения стен ( $\rho_{ст}$ ) и потолка ( $\rho_n$ ).

Индекс помещения определяется по формуле

$$i = \frac{S}{h \cdot (A + B)}. \quad (5.2)$$

Коэффициенты отражения оцениваются субъективно. Произведем расчет:

$$h = H - 0,8 = 2,4 - 0,8 = 1,6 \text{ м}, \quad (5.3)$$

где  $h$  – расчетная высота подвеса светильников над рабочей поверхностью.

Экономичность осветительной установки зависит от отношения, представленного в формуле:

$$l = \frac{L}{h}, \quad (5.4)$$

где  $L$  – расстояние между рядами светильников, м.

Рекомендуется размещать люминесцентные лампы параллельными рядами, принимая  $l = 1,4$ , отсюда расстояние между рядами светильников:

$$L = l \cdot h = 1,4 \cdot 1,6 = 2,24 \text{ м} \quad (5.5)$$

Два ряда светильников будут расположены вдоль длинной стены помещения. Расстояние между двумя рядами светильников и стенами вычисляется по формуле:

$$Л = \frac{В - L}{2} = \frac{3 - 2,24}{2} = 0,38 \text{ м} \quad (5.6)$$

Определим индекс помещения вычисляя по формуле (5.2) получаем:

$$i = \frac{12}{1,6 \cdot (4 + 3)} = 1,07$$

Найдем коэффициенты отражения поверхностей стен, пола и потолка.

Так как поверхность стен окрашена в серый цвет, свежепобеленные с окнами без штор, то коэффициент отражения поверхности стен  $\rho_{ст} = 50\%$ . Так как поверхность потолка светлый окрашенный, то коэффициент отражения поверхности потолка  $\rho_{п} = 30\%$ . Учитывая коэффициенты отражения поверхностей стен, потолка и индекс помещения  $i$ , определяем значение коэффициента  $\eta = 36\%$ .

Подставив все значения в формулу (5.1), по которой рассчитывается световой поток одного источника света, получаем:

$$\Phi = \frac{300 \cdot 12 \cdot 1,5 \cdot 1,1}{4 \cdot 0,36} = 4125 \text{ лм}$$

По полученному световому потоку подбираем лампу, наиболее подходящей является лампа ЛБР-80-1 со световым потоком 4160 (лм).

Выразим  $E$  из формулы (5.1):

$$E = \frac{F \cdot N \cdot \eta}{k} = \frac{4160 \cdot 4 \cdot 0,36}{1,5 \cdot 12 \cdot 1,1} = 302,5 \text{ лм} \quad (5.7)$$

Как видно из расчета, минимальная освещенность в пределах нормы. Для того чтобы доказать, что использование люминесцентной лампы ЛБР-80-1 является наиболее рациональным, рассчитаем необходимое количество светильников по формуле:

$$N = \frac{E \cdot k \cdot S \cdot Z}{F \cdot n \cdot \eta}, \quad (5.8)$$

где  $E$  – норма освещенности  $E = 300$  (Лк);  $k$  – коэффициент запаса учитывающий старение ламп и загрязнение светильников,  $k = 1,5$ ;  $S$  – площадь помещения;  $Z$  – коэффициент неравномерности освещения,  $Z = 1,1$ ;  $n$  – число рядов светильников,  $n = 2$ ;  $\eta$  – коэффициент использования светового потока,  $\eta = 0,36$ ;

$F$  – световой поток, излучаемый светильником.

Подставим численные значения в формулу (5.8), получим количество светильников в одном ряду:

$$N = \frac{300 \cdot 1,5 \cdot 12 \cdot 1,1}{2 \cdot 4160 \cdot 0,36} = 1,98 \text{ шт}$$

Длина одного светильника равна 1,5 (м), в одном светильнике 2 лампы ЛБР-80-1.

Так как в рассматриваемом помещении количество ламп 4 (шт), по одному светильнику в двух рядах, следовательно, нормы безопасности по искусственному освещению в данном случае соблюдены.

## 5.2 Экологическая безопасность

В данном разделе рассматривается воздействие на окружающую среду деятельности по разработке проекта, а также самого продукта в результате его реализации на производстве.

Разработка программного обеспечения и работа за ПЭВМ не являются экологически опасными работами, потому объект, на котором производилась

разработка продукта, а также объекты, на которых будет производиться его использование операторами ПЭВМ относятся к предприятиям пятого класса.

Непосредственно программный продукт, разработанный в ходе выполнения магистерской диссертации, не наносит вреда окружающей среде ни на стадиях его разработки, ни на стадиях эксплуатации. Однако, средства, необходимые для его разработки и эксплуатации могут наносить вред окружающей среде.

### **5.2.1 Утилизация комплектующих ПК**

Современные ПЭВМ производят практически без использования вредных веществ, опасных для человека и окружающей среды. Исключением являются аккумуляторные батареи компьютеров и мобильных устройств. В аккумуляторах содержатся тяжелые металлы, кислоты и щелочи, которые могут наносить ущерб окружающей среде, попадая в гидросферу и литосферу, если они были неправильно утилизированы. Для утилизации аккумуляторов необходимо обращаться в специальные организации, специализировано занимающиеся приемом, утилизацией и переработкой аккумуляторных батарей.

### **5.2.2 Утилизация люминесцентных ламп**

Люминесцентные лампы, применяющиеся для искусственного освещения рабочих мест, также требуют особой утилизации, т.к. в них присутствует от 10 до 70 мг ртути, которая относится к чрезвычайно-опасным химическим веществам и может стать причиной отравления живых существ, а также загрязнения атмосферы, гидросферы и литосферы. Сроки службы таких ламп составляют около 5-ти лет, после чего их необходимо сдавать на переработку в специальных пунктах приема. Юридические лица обязаны сдавать лампы на переработку и вести паспорт для данного вида отходов.

## 5.3 Защита в чрезвычайных ситуациях

### 5.3.1 Перечень возможных ЧС на объекте

На объекте возможные следующие ЧС:

- короткое замыкание;
- пожар.

Наиболее типичной ЧС для данного объекта является короткое замыкание.

Для повышения устойчивости объекта к данной ЧС приняты следующие меры:

- используется внешний стабилизатор напряжения.

В случае возникновения на объекте ЧС будут произведены следующие действия:

- немедленное аварийное отключение устройства;
- обесточивание всей лаборатории во избежание короткого замыкания.

В этом разделе наиболее актуальным будет рассмотрение вида ЧС - пожар, определение категории помещения по пожаровзрывобезопасности в котором происходит управление технологическим процессом, то есть пост управления №8 и регламентирование мер противопожарной безопасности.

Рабочее место оператора поста управления, должно соответствовать требованиям ФЗ Технический регламент по ПБ и норм пожарной безопасности (НПБ 105-03) и удовлетворять требованиям по предотвращению и тушению пожара по ГОСТ 12.1.004-91 и СНиП 21-01-97.

По пожарной, взрывной, взрывопожарной опасности помещение (ПУ№8) относится к категории В – горючие и трудногорючие жидкости, твердые горючие и трудногорючие вещества и материалы, вещества и материалы, способные при взаимодействии с водой, кислородом воздуха или друг с другом только гореть.

Основным поражающим фактором пожара для помещений данной категории является наличие открытого огня и отравление ядовитыми продуктами сгорания оборудования.

### **5.3.2 Меры по предотвращению и ликвидации ЧС и их последствий**

Пожарная безопасность объекта должна обеспечиваться системами предотвращения пожара и противопожарной защиты, в том числе организационно-техническими мероприятиями.

Пожарная защита должна обеспечиваться применением средств пожаротушения, а также применением автоматических установок пожарной сигнализации.

Должны быть приняты следующие меры противопожарной безопасности:

- обеспечение эффективного удаления дыма;
- обеспечение правильных путей эвакуации;
- наличие огнетушителей и пожарной сигнализации;
- соблюдение всех противопожарных требований к системам отопления и кондиционирования воздуха.

Для тушения пожаров на участке производства необходимо применять углекислотные и порошковые огнетушители, которые обладают высокой скоростью тушения, большим временем действия, возможностью тушения электроустановок, высокой эффективностью борьбы с огнем.

Помещение оборудовано пожарными извещателями, которые позволяют оповестить дежурный персонал о пожаре. В качестве пожарных извещателей в помещении устанавливаются дымовые фотоэлектрические извещатели типа ИДФ-1 или ДИП-1. Выведение людей из зоны пожара должно производиться по плану эвакуации. План эвакуации людей при пожаре из помещения, где расположен диспетчерский пункт (пост управления), представлен на рисунке 5.1.

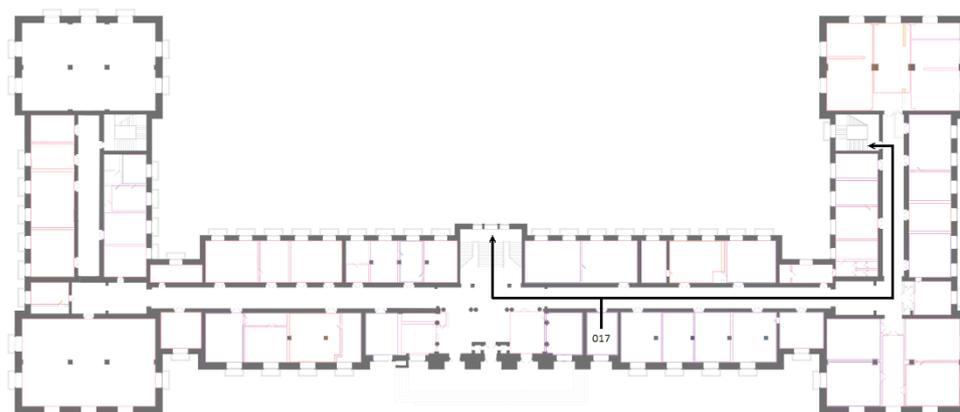


Рисунок 5.1 - План эвакуации при пожаре

Ответственность за нарушение Правил пожарной безопасности, согласно действующему федеральному законодательству, несет руководитель объекта.

## **5.4 Правовые и организационные вопросы обеспечения безопасности**

### **5.4.1. Специальные правовые нормы трудового законодательства**

Нормы трудового права – это правила трудовых отношений, установленные или санкционированные государством посредством законодательных актов. Нормы трудового права регулируют любые отношения, связанные с использованием личного труда.

Рассмотрим регулирование коллективных отношений. Настоящий коллективный договор является правовым актом, регулирующим социально-трудовые отношения работников ФГАОУ НИ ТПУ с работодателем. Основной задачей коллективного договора является создание необходимых организационно-правовых условий для достижения оптимального согласования интересов сторон трудовых отношений.

По заключенному коллективному договору работодатель обязан:

- соблюдать трудовое законодательство и иные нормативные правовые акты;
- предоставлять работникам работу, обусловленную трудовым договором;
- обеспечивать безопасность и условия труда;

- обеспечивать работников оборудованием, инструментами, технической документацией;
- обеспечивать работникам равную оплату за труд равной ценности;
- выплачивать в полном размере причитающуюся работникам заработную плату в сроки;
- знакомить работников под роспись с принимаемыми локальными нормативными актами;
- осуществлять обязательное социальное страхование работников в порядке;
- принимать необходимые меры по профилактике производственного травматизма
- положениями специальной одеждой и обувью, другими средствами индивидуальной защиты;
- постоянно контролировать знание и соблюдение работниками всех требований инструкций по охране труда.

Ежегодно издавать приказ о мероприятиях по охране труда и промышленной безопасности, считать эти мероприятия соглашением по охране труда на год. Обеспечивать за счет средств работодателя:

- Проведение инструктажей по охране труда.
- Проведение медицинских осмотров работников.

Наличие на производственных участках аптечек для оказания первой помощи пострадавшим и обработки микротравм; наличие в аптечках рекомендованного МЛПУ «Городская клиническая больница №1» перечня средств и медикаментов, их ежегодную замену.

#### **5.4.2. Организационные мероприятия при компоновке рабочей зоны**

Проектирование рабочих мест, снабженных видеотерминалами, относится к числу важных проблем эргономического проектирования в области вычислительной техники.

Организация рабочего места программиста или оператора регламентируется следующими нормативными документами: ГОСТ 12.2.032-78 ССБТ, ГОСТ 12.2.033-78 ССБТ, СанПиН 2.2.2/2.4.1340-03 и рядом других.

Эргономическими аспектами проектирования видеотерминальных рабочих мест являются: высота рабочей поверхности, размеры пространства для ног, требования к расположению документов на рабочем месте, характеристики рабочего кресла, требования к поверхности рабочего стола, регулируемость элементов рабочего места.

Главными элементами рабочего места программиста или оператора являются стол и кресло. Основным рабочим положением является положение сидя. Рациональная планировка рабочего места предусматривает четкий порядок и постоянство размещения предметов, средств труда и документации.

Моторное поле - пространство рабочего места, в котором могут осуществляться двигательные действия человека. Максимальная зона досягаемости рук - это часть моторного поля рабочего места. Оптимальная зона - часть моторного поля рабочего места, ограниченного дугами, описываемыми предплечьями при движении в локтевых суставах с опорой в точке локтя и с относительно неподвижным плечом.

### **Заключение**

В результате создания данного раздела был проведен анализ выявленных вредных и опасных факторов при эксплуатации проектируемого решения и даны рекомендации по улучшению освещения на рабочем месте. Также рассмотрены утилизации комплектующих ПК и люминесцентных ламп. Проведены мероприятия по защите от чрезвычайных ситуаций в частности пожара, а также рассмотрены правовые и организационные вопросы обеспечения безопасности. Таким образом рабочее место и помещение, где оно находится, соответствует всем нормам и безопасно для работников.

## Заключение

В данной работе была проведена разработка стратегии ведения игры, и на её основе с применением технического зрения был создан робот.

В ходе работы были выбраны методы реализации выигрышной стратегии и распознавания карт на изображения. Для выигрышной стратегии использовался метод ветвей и границ, а для распознавания карт на изображении - сочетание контурного анализа и распознавания по ключевым точкам.

Кроме того, для робота была выбрана камера и разработан эмулятор клавиатуры. В качестве камеры использовалась Logitech c270, а эмулятор клавиатуры разрабатывался на основе двух Arduino micro pro.

Также было выбрано программное обеспечение для реализации алгоритма и язык программирования. В качестве операционной системы выбрана Lubuntu 16.10, среды разработки - Code Blocks, языка программирования - C++. Для разработки программного обеспечения эмулятора использована Arduino IDE.

Было проведено сравнение работы разработанного робота с выборкой результатов партий различной сложности, сыгранных людьми. В итоге процент выигранных партий оказался несколько ниже.

В дальнейшем разработанные алгоритмы и устройства могут быть использованы как вместе, так и по отдельности в образовательной деятельности или игровой сфере.

В ходе выполнения работы был получен опыт работы с языком программирования C++, работой с библиотеками OpenCV на Linux и использования сред разработки Code Blocks и Arduino IDE.

## Список использованных источников

1. Искусственный интеллект победил чемпиона мира по игре в го [Электронный ресурс] / URL: <http://masculist.ru/forum/thread3241.html> – Загл. с экрана – Яз. рус. Дата обращения 30.05.2018 г.
2. Компьютерное зрение [Электронный ресурс] / URL: [https://ru.wikipedia.org/wiki/Компьютерное\\_зрение](https://ru.wikipedia.org/wiki/Компьютерное_зрение) – Загл. с экрана – Яз. рус. Дата обращения 30.05.2018 г.
3. История возникновения Пасьянса Паука [Электронный ресурс] / URL: <http://spider-solitaire.ru/histoiry.html> – Загл. с экрана – Яз. рус. Дата обращения 30.05.2018 г.
4. Паук (пасьянс) [Электронный ресурс] / URL: [https://wiki2.org/ru/Паук\\_\(пасьянс\)](https://wiki2.org/ru/Паук_(пасьянс)) – Загл. с экрана – Яз. рус. Дата обращения 30.05.2018 г.
5. Пасьянс «Паук» [Электронный ресурс] / URL: <http://solitaireclub.ru/pauk.html> – Загл. с экрана – Яз. рус. Дата обращения 30.05.2018 г.
6. How Many Games of Spider Solitaire are Winnable? / Mark S. Weisser – 2012. – 92 с.
7. Winnable Spider Solitaire Games [Электронный ресурс] / URL: <http://www.tranzoa.net/alex/plspider.htm> – Загл. с экрана – Яз. англ. Дата обращения 30.05.2018 г.
8. Spider Solitaire is NP-Complete / Jesse Stern Weisser – 2011. – 7 с.
9. Spider Solitaire is NP-Compleet / Kenneth Verstraete – 2016. – 6 с.
10. Xiang Yan Solitaire: Man Versus Machine / Xiang Yan Persi Diaconis Paat Rusmevichientong† Benjamin Van Roy – 2010. – 8 с.
11. Sturtevant Nathan R. Feature Construction for Reinforcement Learning in Hearts / Nathan R. Sturtevant, Adam M. White – 2006. – 12 с.
12. Implementing a Doppelkopf Card Game Playing AI Using Neural Networks / Johannes Obenaus – 2017. – 35 с.

13. Spider Solitaire Winning Strategies/ Steve N. Brown– 2016– 250 с.
14. Пасьянс «Паук» [Электронный ресурс] / URL: <http://www.kartabita.ru/?idGame=4&id=71857> – Загл. с экрана – Яз. рус. Дата обращения 30.05.2018 г.
15. Computer vision: past and future, in Informatics, Springer / J.-O. Eklundh, H. I. Christensen – 2001 – P. 328-340.
16. On what it means to see, and what we can do about it, in Object categorization: computer and human vision perspectives/ S. Edelman, S. – 2009 – 10 с.
17. From image analysis to computer vision: motives, methods, and milestones / A. Rosenfeld– 1998 – 15 с.
18. Guest Introduction: The changing shape of computer vision in the twenty-first century, International Journal on Computer Vision / M. Shah – 2002 – Vol. 50, № 2. – P. 103-110.
19. Гонсалес Р. Цифровая обработка изображений: пер. с англ. / Р. Гонсалес, Р. Вудс – Москва: Техносфера, 2005. – 1072 с.
20. Прэтт У. Цифровая обработка изображений: пер. с англ. / У. Прэтт – М.: Мир, 1982. – Кн.1 – 312 с.
21. Прэтт У. Цифровая обработка изображений: пер. с англ. / У. Прэтт – М.: Мир, 1982. – Кн. 2 – 480 с.
22. Хуанг Т. С. Быстрые алгоритмы в цифровой обработке изображений: пер. с англ. / Т. С. Хуанг, Дж.-О. Эклунд, Г. Дж. Нус-сбаумер и др – М.: Радио и связь, 1984. – 224 с.
23. Форсайт Д.А. Компьютерное зрение. Современный подход: пер. с англ. / Форсайт Д.А., Понс Ж. – М. : Издательский дом "вильямс", 2004. – 928 с.
24. Шапиро Л. Компьютерное зрение/ Л. Шапиро, Дж. Стокман Бином – 2006. – 752 с.
25. Computer Vision: Algorithms and Applications [Электронный ресурс] / URL: <http://szeliski.org/Book/> – Загл. с экрана – Яз. англ. Дата обращения 30.05.2018 г.

26. Computer Vision: Models, Learning, and Inference [Электронный ресурс] / URL:<http://www.computervisionmodels.com/> – Загл. с экрана – Яз. англ. Дата обращения 30.05.2018 г.
27. Programming Computer Vision with Python Inference [Электронный ресурс] / URL:<http://programmingcomputervision.com/> – Загл. с экрана – Яз. англ. Дата обращения 30.05.2018 г.
28. OpenCV Inference [Электронный ресурс] / URL:<http://opencv.org/> – Загл. с экрана – Яз. англ. Дата обращения 30.05.2018 г.
29. SimpleCV [Электронный ресурс] / URL:<http://www.simplecv.org/> – Яз. англ. Дата обращения 30.05.2018 г.
30. ViSP [Электронный ресурс] / URL:<http://visp.inria.fr> – Яз. англ. Дата обращения 30.05.2018 г.
31. Shark machine learning library [Электронный ресурс] / URL:[http://image.diku.dk/shark/sphinx\\_pages/build/html/index.html](http://image.diku.dk/shark/sphinx_pages/build/html/index.html) – Яз. англ. Дата обращения 30.05.2018 г.
32. OpenVIDIA : Parallel GPU Computer Vision [Электронный ресурс] / URL:<https://www.openhub.net/p/openvidia> – Яз. англ. Дата обращения 30.05.2018 г.
33. Computer Vision System Toolbox [Электронный ресурс] / URL:<http://www.mathworks.com/products/computer-vision/> – Яз. англ. Дата обращения 30.05.2018 г.
34. CSE/EE486 Computer Vision I [Электронный ресурс] / URL:<http://www.cse.psu.edu/~rtc12/CSE486/> – Яз. англ. Дата обращения 30.05.2018 г.

35. CS395T: Visual Recognition [Электронный ресурс] / URL: <http://www.cs.utexas.edu/~grauman/courses/fall2011/schedule.html> – Яз. англ. Дата обращения 30.05.2018 г.
36. Grounding Object Recognition and Scene Understanding [Электронный ресурс] / URL: [http://people.csail.mit.edu/torr/alba/courses/6.870\\_2011f/6.870.grounding](http://people.csail.mit.edu/torr/alba/courses/6.870_2011f/6.870.grounding) – Яз. англ. Дата обращения 30.05.2018 г.
37. Castillo Matias Computer Vision for Card Games / Matias Castillo, Benjamin Goeing, Jesper Westell – 2009 – 5 с.
38. Paulo Joao Machine Vision in Casino Game Monitoring / Joao Paulo, Mauricio Pimentel – 2009 – 5 с.
39. NP-полная задача [Электронный ресурс] / URL: [http://ru.science.wikia.com/wiki/NP-полная\\_задача](http://ru.science.wikia.com/wiki/NP-полная_задача) – Яз. рус. Дата обращения 30.05.2018 г.
40. Методы решения np-полных задач [Электронный ресурс] / URL: <https://studfiles.net/preview/2213464> – Яз. рус. Дата обращения 30.05.2018 г.
41. Задача о ранце [Электронный ресурс] / URL: <https://studfiles.net/preview/1955920> – Яз. рус. Дата обращения 30.05.2018 г.
42. Анализ алгоритмов компьютерного зрения [Электронный ресурс] / URL: <https://arealidea.ru/articles/analiz-algoritmov-kompyuternogo-zreniya-poiska-obektov-i-sravneniya-izobrazheniy/> – Яз. рус. Дата обращения 30.05.2018 г.
43. OpenCV шаг за шагом. Нахождение контуров и операции с ними [Электронный ресурс] / URL: <http://robocraft.ru/blog/computervision/640.html> – Яз. рус. Дата обращения 30.05.2018 г.
44. USB HID [Электронный ресурс] / URL: [https://ru.wikipedia.org/wiki/USB\\_HID](https://ru.wikipedia.org/wiki/USB_HID) – Яз. рус. Дата обращения 30.05.2018 г.
45. Lubuntu [Электронный ресурс] / URL: <https://ru.wikipedia.org/wiki/Lubuntu> – Яз. рус. Дата обращения 30.05.2018 г.

46. Installation in Linux [Электронный ресурс] / URL: [https://docs.opencv.org/trunk/d7/d9f/tutorial\\_linux\\_install.html](https://docs.opencv.org/trunk/d7/d9f/tutorial_linux_install.html) – Яз. англ. Дата обращения 30.05.2018 г.
47. Фильтры в OpenCV. Average и GaussianBlur [Электронный ресурс] / URL: <http://techcave.ru/posts/66-filtry-v-opencv-average-i-gaussianblur.html> – Яз. рус. Дата обращения 30.05.2018 г.
48. Пороговое преобразование [Электронный ресурс] / URL: [http://altamisoft.ru/products/altami\\_studio/user\\_manual\\_AS\\_3\\_1\\_0/filters/gray\\_transformations/threshold/](http://altamisoft.ru/products/altami_studio/user_manual_AS_3_1_0/filters/gray_transformations/threshold/) – Яз. рус. Дата обращения 30.05.2018 г.
49. Pablo Fernandez Alcantarilla KAZE Features / Pablo Fernandez Alcantarilla, Adrien Bartoli, and Andrew J. Davison – 2012 – 14 с.
50. Arduino Micro и Arduino Pro Micro [Электронный ресурс] / URL: <https://all-arduino.ru/arduino-micro/> – Яз. рус. Дата обращения 30.05.2018 г.

# Приложение А

(обязательное)

## Раздел 1

### Review of literature and scientific articles

Студент:

Группа	ФИО	Подпись	Дата
8АМ61	Липатов Дмитрий Сергеевич		

Консультант школы отделения (НОЦ) ИШИТиР ОАиР:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Доцент ОАиР	Замятин С.В.	Доцент, к.т.н		

Консультант – лингвист отделения (НОЦ) школы ИШИТиР ОАиР:

Должность	ФИО	Ученая степень, звание	Подпись	Дата
Старший преподаватель ОИЯ	Шепетовский Денис Владимирович			

## Introduction

Modern robots have penetrated into many areas of human life. In many ways they are superior or on a par with the person. The gaming field robots were marked by recent victories in Go and Dota 2[1].

In this paper, we consider the development of the algorithm and the implementation of it on the basis of a robot that plays Spider solitaire .

The algorithm was tested, comparing the number of games won and the time spent on the game at different amounts of suits with the statistics of people playing solitaire.

Then a robot was designed and created that imitates human from the point of view of the computer and the game. For this purpose, the technology of technical vision was used, which allows the detection, tracking and classification of objects [2], as if imitating human vision. An emulator has been created for the input device, which is recognized by the computer as a keyboard and allows you to emulate key presses or their combinations.

The purpose of this work is to develop a strategy for the game and on its basis to create a robot using technical vision.

To achieve this goal, you must perform the following tasks:

- Study of the methods of implementation of winning strategies and choosing the best method;
- Investigation of recognition methods in image;
- The development and implementation of the algorithm a winning strategy in the game;
- Assessment of the effectiveness of the algorithm;
- Development and implementation of the map recognition algorithm on the image;
- Development of a robot without I / o devices;
- Design and integration of I / o devices into the robot;
- Testing and error correction.

This goal solves the problems of control theory, robotics, computer vision:

In the Control theory, this work solves the following problems:

- determination of the optimal course;
- game strategy.

In the Robotics, this work solves the following problems:

- design of the robot;
- data collection and transmission.

In the Computer vision, this work solves the following problems:

- recognition of the playing field;
- recognition of playing cards;
- determination of position of the cards.

The object of research in this work is the algorithms of winning strategy of the game and technical vision used for the game robot in Spider solitaire .

The subject of the research is the problem of designing and developing a robot that implements the algorithms of winning strategy of the game and technical vision.

The scientific novelty of this work is the use of algorithms of winning strategy of the game and recognition of cards in the image.

Implementation of the work: in the course of the work, a prototype was developed.

## **1. Review of literature and scientific sources**

In this Chapter, works relating to the topics covered in my master's thesis are described.

### **1.1 Review of research on Spider solitaire**

This subsection presents the history and rules of the game spider solitaire and scientific articles, which consider the game as an object of research.

#### **1.1.1 The history of Spider solitaire**

Solitaire such as "Spider" is known to almost everyone thanks to its presence in the set of Aisleriot, mobile applications and Windows software, but before the electronic era of this game, which began in the early nineties, it had its her card history.

Most people who are keen on studying of games believe that the birthplace of all solitaire was the famous Bastille in French Paris. These layouts of card became well-known in the 16th century. Solitaire rules began to be established only in the days of Napoleon Bonaparte, when these layouts have become even more common. In the Russian Empire, the first book about solitaire guided "Collection of card layouts Grand solitaire." It was released in 1826. Another well-known edition was published in 1870 by the English woman Adelaide Cadogan, where solitaire "Spider" was described[3].

### **1.1.2 Rules of the game Spider solitaire**

Solitaire requires 2 decks of 52 cards. First, 54 cards are divided into 10 columns, the remaining 50 are divided into 5 piles of 10 cards. In each column every top card is opened. It is allowed to transfer cards between columns. Cards can be moved according to these rules:

- you can move any top card to an empty column;
- it is allowed to move the next by its precedence card of any suit on the top card of any column;
- the sequence of cards of the same suit spread out by the consolation value can be moved as a single card.

If the top card of the column is closed, it opens automatically. If a sequence of ordered cards of the same suit is collected, this sequence is discarded. It is allowed at any time to take one of the piles of cards and spread out one card in each column. The game is won if all cards are discarded[4].

There are three levels of difficulty of this solitaire, which differ in the number of suits. The first level (One Suit) is the easiest, in it the player lays out a deck of cards of suit spades (each card is repeated 8 times). The second level of difficulty (Two Suits) offers to play with two suits - spades and hearts (each card is repeated 4 times). At the third level of difficulty (Four Suits), the player will have to lay out cards of all four suits (the deck contains 2 identical cards of each suit) [5].

### **1.1.3 The research of the game Spider solitaire**

Scientific articles in Russian language that consider the game were not found. However, they are available in foreign languages. So in Mark Weisser's article "How Many Games of Spider Solitaire are Winnable?" [6] the percentage of games in which it is theoretically possible to win is calculated mathematically. There it is equal to 99.9%. There is also a website called "Winnable Spider Solitaire Games" [7], created by Alex Robinson. In it, the author collected the results of 32,000 random games in solitaire "Spider". The percentage of winning these games was 99.991%.

Another topic that the researchers are considering is what type of task of creating a winning strategy game of solitaire "Spider". So in the article "Spider Solitaire is NP-Complete" [8] the author Jesse Stern proves that winning Spider solitaire is an NP-complete problem. Further on in the article "Spider Solitaire is NP-Complete" [9] author Kenneth Verstraete confirms his proof.

As for other card games, many of them are solved with the help of machine learning. For example, in the article "Solitaire: Man Versus Machine" [10], the authors use the deployment method for playing solitaire. The strategy, which was derived, uses re-deployment. It wins on average twice as many games as human. In addition, using machine learning methods, strategies were developed for the games of Worms and Shafkopf [11-12]. These games are already designed for several people, and the developed strategies allow achieving a high percentage of victories over human opponents.

To compare the developed algorithm and human you need to find statistics. The theoretical calculation of the win percentage was taken from the book "Spider Solitaire Winning Strategies" [13] by Steve Brown. In it, the author writes that the winning percentage for four suits is 45% at two 80% at one 99%. Also, statistics were taken from the site [14] where people actually play spider solitaire.

## **1.2 Technical vision**

Technical vision (Computer vision) — theory and technology of creating machines that can detect, track and classify objects.

As a scientific discipline, computer vision is concerned with the theory behind artificial systems that extract information from images.

As a technological discipline, computer vision seeks to apply its theories and models to the construction of computer vision systems.

Sub-domains of computer vision include scene reconstruction, event detection, video tracking, object recognition, 3D pose estimation, learning, indexing, motion estimation, and image restoration. [2].

The organization of a computer vision system is highly application dependent. Some systems are stand-alone applications that solve a specific measurement or detection problem, while others constitute a sub-system of a larger design that, for example, also contains sub-systems for control of mechanical actuators, planning, information databases, man-machine interfaces, etc. The specific implementation of a computer vision system also depends on if its functionality is pre-specified or if some part of it can be learned or modified during operation. Many functions are unique to the application. There are, however, typical functions that are found in many computer vision systems.

**Image acquisition** – A digital image produced by one or several image sensors, which, besides various types of light-sensitive cameras, include range sensors, tomography devices, radar, ultra-sonic cameras, etc.

**Pre-processing** – Before a computer vision method can be applied to image data in order to extract some specific piece of information, it is usually necessary to process the data in order to assure that it satisfies certain assumptions implied by the method.

**Feature extraction** – Image features at various levels of complexity are extracted from the image data.

**Detection/segmentation** – At some point in the processing, a decision is made about which image points or regions of the image are relevant for further processing.

**High-level processing** – At this step the input is typically a small set of data, for example a set of points or an image region which is assumed to contain a specific object.[2]

### **1.2.1 History of computer vision**

Computer vision has a short history of development by the standards of science. As an independent discipline, computer vision originated in the early 1950s. In 1951, John von Neumann proposed to analyze micro-images using computers by comparing the brightness in neighboring areas of images.[13].

In the 1960s, research began in the field of recognition of typewritten and handwritten text, as well as in the classification of chromosomes and cells in images obtained from a microscope. The first success in the field of technical vision can be associated with the development of the psychologist Frank Rosenblatt, Cornell Aeronautics laboratory — perceptron. The perceptron was first modeled on the IBM-740 mainframe in 1958. The hardware version of the perceptron-Mark I Perceptron was made in 1960 and was intended for recognition of visual images. In 1963, Roberts' thesis appeared, which offered a simple contour detector and proposed the first methods of recognition of three-dimensional objects[13].

The existence of an extremely complex relationship between the properties of three-dimensional objects of the world and their two-dimensional images was realized in the early 1970s. As a result, in the 1970s, the British neuroscientist and psychologist Marr became the founder of the scientific program, which was based on the study of the human visual system with the aim of its formalization and implementation in the form of algorithms.

Levels Associated with these stages of processing are called respectively: processing of the lower level, the middle level, the high level[14].

In the 1980s, another research program became popular, which consisted in finding new complex mathematical methods for solving computer vision problems. While the usage of a number of mathematical methods has allowed solving complex problems of computer vision, in many cases, this approach has allowed only to find the use of mathematical methods, and not to solve the problem of vision [15].

The nineties of the twentieth century with the advent of a new generation of sensors for two-dimensional digital information of the fields significantly reduced the

cost, which made it possible to expand the scope considerably. The increasing performance of microprocessors, the reduction in prices of cameras and data transmission speed in the transmission of video has allowed the realization of operating in real-time computer vision algorithms on standard computers [16].

In the late 1990s and during the first decade of the XXI century in computer vision there was a qualitative leap in several directions. In this period, it is difficult to identify any new paradigms covering the whole discipline. Rather, the jump is caused by a sharp increase in interest in computer vision and, as a consequence, a great enthusiasm in the transfer of methods from other disciplines into computer vision. First of all, the result of this breakthrough became the progress in the methods of image description [16].

### **1.2.2 The literature on computer vision**

One of the first books that reveals the technical problems of solving computer vision problems is the work of W. Pratt "Digital image processing"[18,19]. The book is devoted to the mathematical representation of continuous and discrete images. It considers in detail two-dimensional transforms, including Fourier, Hadamard and Caruana-Loewe. Further, one of the famous authors is T. S. Huang, who in 1981 outlined the basics of the theory and application of new computationally efficient algorithms for digital image processing in the book "Fast algorithms in digital image processing"[20].

Among modern publications on this subject are such books as "Computer vision. A modern approach"[21] by D. A. Forsythe and "Digital image processing"[17] by R. Gonzalez, published in 2004 and 2005. These books describe the basic concepts and methodology of computer image processing, provide the basis for further study of the field.

The book "Computer vision" [22] by L. Shapiro, J. Stockman can also be noted. In this book, the theoretical aspects of visual data processing are considered with the involvement of large number examples of practical problems. Along with the classic themes, the book deals with image databases and virtual and augmented

reality systems. Examples of applications in industry, medicine, land use, multimedia and computer graphics are given.

Among those that have not been translated yet can also be noted "Computer Vision: Algorithms and Applications"[23] by Richard Szeliski, "Computer Vision: Models, Learning, and Inference" by Simon J. D. Prince and "Programming Computer Vision with Python"[24] by Jan Erik Solem. In the first book, the author explores the various methods most commonly used in analyzing and interpreting images. It also describes complex algorithms that are successfully used in the real world, both for specialized applications and for consumer-level tasks. The second book describes best practices in detail. The third one explains the theory of computer vision by the example of programs with explanations.

### **1.2.3 Online sources on computer vision**

In addition to literary sources, a lot of information can be found on the websites of library developers containing computer vision algorithms. The most popular library is Open Computer Vision (OpenCV). On site [opencv.org](http://opencv.org), the developer can find documentation, books (Learning OpenCV, Mastering OpenCV) and examples of implemented programs [26]. On site [vlfeat.org](http://vlfeat.org) computer vision algorithms on pure C are presented, there are interfaces for MathLab. Website [www.simplecv.org](http://www.simplecv.org) contains a C/C++ library built on top of OpenCV. There is also a manual for beginners - "Practical Computer Vision with SimpleCV"[27]. The main goal of the project is to provide a simplified interface to all algorithms.

Also there are ViSP libraries written in C++ with computer vision algorithms (mainly in the field of tracking-tracking and observation) [28] and SHARK – Machine learning library with machine learning algorithms, which compares favorably with alternatives by the presence of algorithms not implemented anywhere else [29]. The OpenVIDIA: Parallel GPU Computer Vision website contains computer vision algorithms on the GPU (CUDA) [30].

Along with OpenCV, MATLAB (toolbox + sample) can be used to start learning and using computer vision algorithms [31].

### **1.2.4 Materials from universities on computer vision**

Along with the websites of developers, many universities put materials of the courses in computer vision in free access. Therefore, on the website of the University of Pennsylvania you can find lectures of Robert Collins on the subject of introduction to computer vision [32]. At the University of Texas at Austin, seminars on computer vision were prepared. There were examined and discussed current concepts relating to object recognition, automatic image annotation and scene understanding. The aim of the seminars was to understand modern approaches to some important problems and identify possible directions for future research [33]. The Massachusetts Institute of technology has a course developed by Professor Antonio Torralba that covers modern approaches to object recognition and the connection of computer vision with other disciplines. The purpose of this course is in-depth presentation of computer vision methods for recognition of objects, scenes, materials, actions [34].

### **1.2.5 Game card recognition studies**

A lot of articles about the recognition of playing cards can be found in English-language scientific publications. This is often due to the work of monitoring and analysis of games in the casino. In the articles "Computer Vision for Card Games" [35] and "Machine Vision in Casino Game Monitoring" [36] the authors consider the recognition of cards located on a real playing table, and use different methods. In the first article, the authors use a neural network for recognition and in the second article contour analysis and search by key points were used.

## **2. Selection of methods and technical solutions**

This section provides an overview and comparison of the methods of implementation of the winning strategy and methods of recognition on the image. Comparisons were made and the optimal ones were chosen. Methods and technologies of input device emulation are also considered and selected.

### **2.1 Selection of methods for implementing the winning strategy**

#### **2.1.1 Statement of the problem**

As considered in section 1.1.3, the task of finding a winning strategy in the game of Spider solitaire is an NP-complete problem. In algorithm theory, an NP-complete problem is a problem from the NP class to which any other problem from the NP class can be reduced. NP-complete problems form a subset of the "most complex" problems in the NP class; and if for some of them a "fast" solution algorithm is found, then any other problem from the NP class can be solved as "fast". [39]

The relationship between the classes P, NP, NP-complete (NP-complete problem), NP-hard (NP-hard problem) if  $P \neq NP$  and if  $P = NP$  are shown in figure 2.1.

Based on the rules of the game, you can create an alphabet. Make a matrix  $C_n = 2(\{\spadesuit, \clubsuit, \heartsuit, \diamondsuit\} \times \{1, \dots, n\})$  where  $n$  is the field number and is an integer  $n \in \mathbb{N}$ , and the elements are the cards. Each map is described by two variables  $(s, v)$  where  $s$  are suits and  $v$  is a value. Cards are black if its suit  $\spadesuit$  or  $\clubsuit$ , and red if  $\heartsuit$  or  $\diamondsuit$ .  $n \times w$  table is a set of non-empty sequences of size no more than  $w$  and more than  $C_n$ . In addition, each card appears in two sequences in the table. The last card in the sequence is called the top card. The subsequence obtained by removing the top card is called the hidden part of the sequence. A card belongs to a sequence if card  $c$  has a value one less than the value of the top card.

Spider solitaire is a four-dimensional ordered set of fixed length  $(n, w, d, T)$ , where

- $n \in \mathbb{N}$  number of fields;
- $w \in \mathbb{N}$  width table;
- $d \in \mathbb{N}$  the size of the field
- $T$  is the initial  $n \times w$  table.

### **2.1.2 Description of the methods of solving the problem**

There are two general categories of approaches to the problem of finding an acceptable algorithm for solving the NP-complete problem. The classification of methods is shown in figure 2.2.

The first group includes approaches that attempt to minimize the search volume, while recognizing the inevitability of exponential time.

Approaches belonging to the second category, applicable only to optimization problems and include the reception, which can be called “reduced requirements”. It is to abandon the search for an optimal solution and instead find a good solution in an acceptable time. Algorithms based on this technique are usually called heuristic or approximate, because they use different reasonable considerations without strict justification. [40]

Let's consider the solution methods in more detail:

- Exhaustive search.

The method of solving mathematical problems, which is to search all possible options. An example of the method is given in figure 2.3.[41]

- Dynamic programming.

Dynamic programming is the process of step-by-step solution of the problem, when at each step one solution is selected from the set of acceptable solutions, and one that optimizes a given target function or criterion function.

However, the possibilities of its application are limited, in particular, by the following factors: the algorithm based on dynamic programming is not always an efficient algorithm, although it can be significantly better than the full search method; the problem can be arranged so that it is impossible to find a "reasonable" way to reduce it to a number of subtasks.[41]

- Branch and bound method.

The branch and bound method is a variation of the full search, with exceptions not known to be optimal solutions. The solution consists in building “partial solutions”, presented in the form of a search tree, and the use of powerful methods for constructing estimates, allowing to identify unpromising partial solutions, resulting

from the search tree at one step is cut off the whole branch. An example of the method is given in figure 2.4.[41]

- "Greedy" or gradient algorithms

Applying a gradient algorithm to a selection problem yields an exact solution. However, due to its simplicity, it is used to construct approximate or heuristic algorithms. At each individual stage, the "greedy" algorithm chooses the option that is locally optimal in one sense or another. [41]

- random search methods

The methods in this group are based on generating a random sequence, then checking if it is a solution. The best solution is chosen from the set of solutions found. The criterion for stopping the search can be either the search time or the number of iterations.

Further development of the idea of greedy algorithms is the development of their probability analogues, the so-called GRASP-algorithms.[40]

- FF type methods and their modifications

These methods are based on the selection of the first suitable for any criterion. The complexity of the algorithm  $O(n)$ . These methods are used to find the original solution, which is then improved by another method.[38]

### **2.1.3 Comparison of methods and selection of optimal**

Let us compare the described methods and summarize in the table in table 2.1.

Random search methods Approximate high speed; can work with large values of  $N$ ; independence from the type of source data does not guarantee finding the optimal solution

As a result of the comparison, the branch and bound method was chosen, which reduces the time; the simplicity of the implementation, however, the disadvantage of the method is that it works as a full search.

The task of obtaining a winning strategy game Spider solitaire must be divided into several subtasks such as:

- Limited search and evaluation of each sequence;

For this subtask, it is necessary to develop a system of search constraints, criteria for evaluating sequences;

- The choice of the optimal sequence.

In this sub-task it is necessary to develop criteria by which the sequence is considered optimal.

- Estimation of the optimal sequence in relation to the passed stages;

In this subtask, you need to develop criteria to compare with the optimal sequence with the stages passed and as a result choose actions for the stage.

- The choice of evaluation criteria and end results.

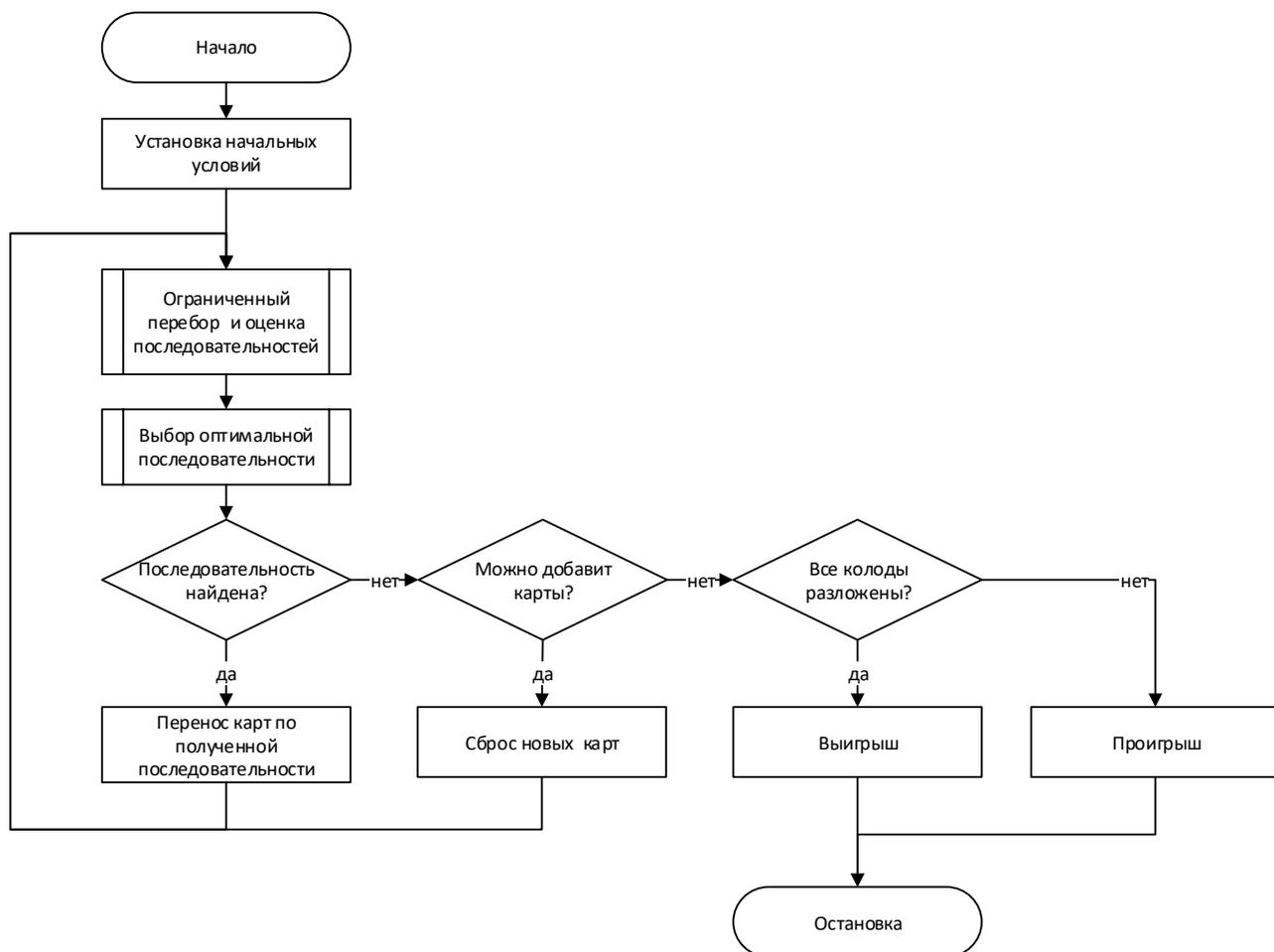
In this sub-task, it is necessary to develop a criterion by which the success of the work will be assessed.

As a result, we can make a generalized block diagram of the algorithm, which is shown in figure 2.5.

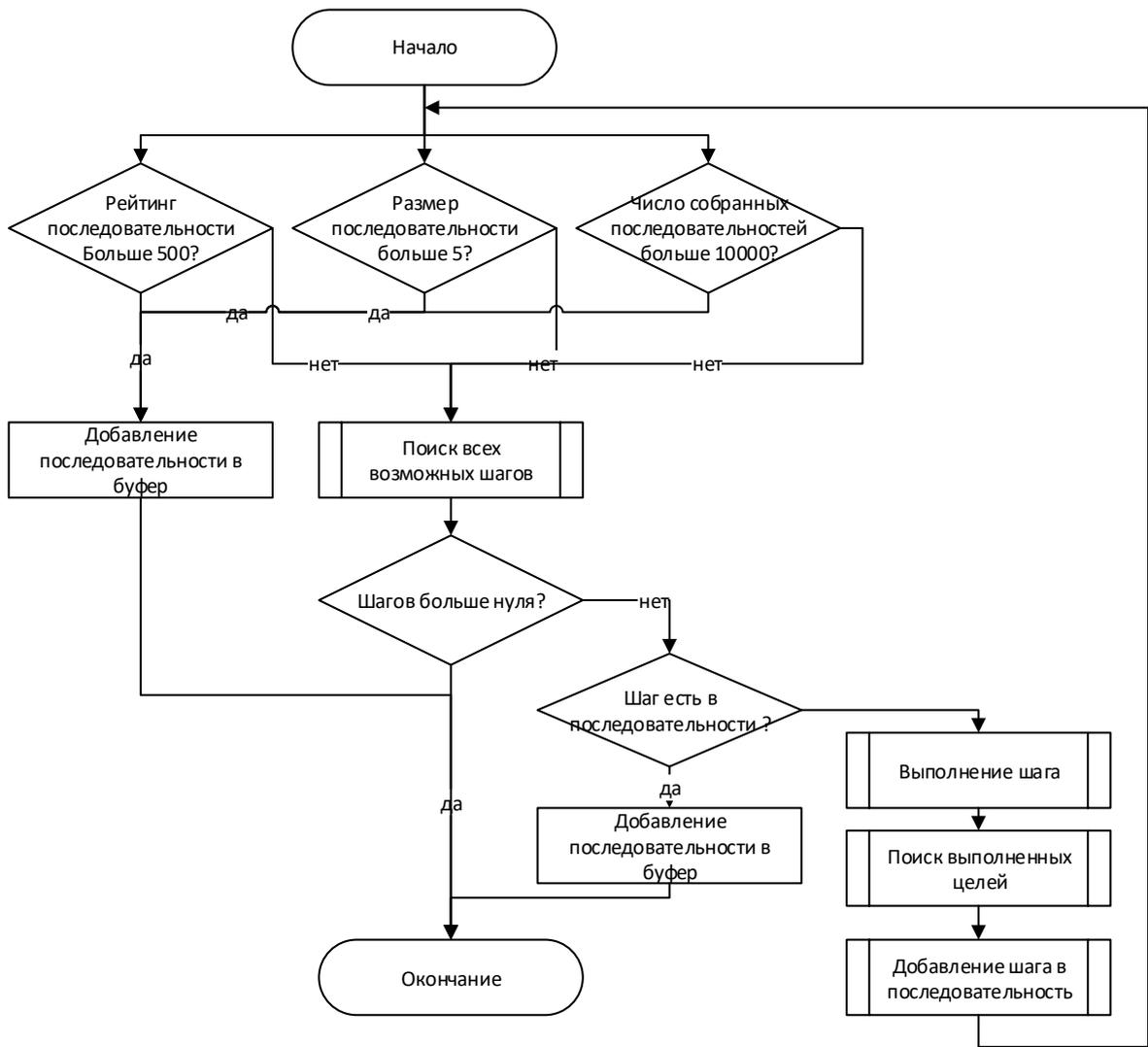
# Приложение Б

(обязательное)

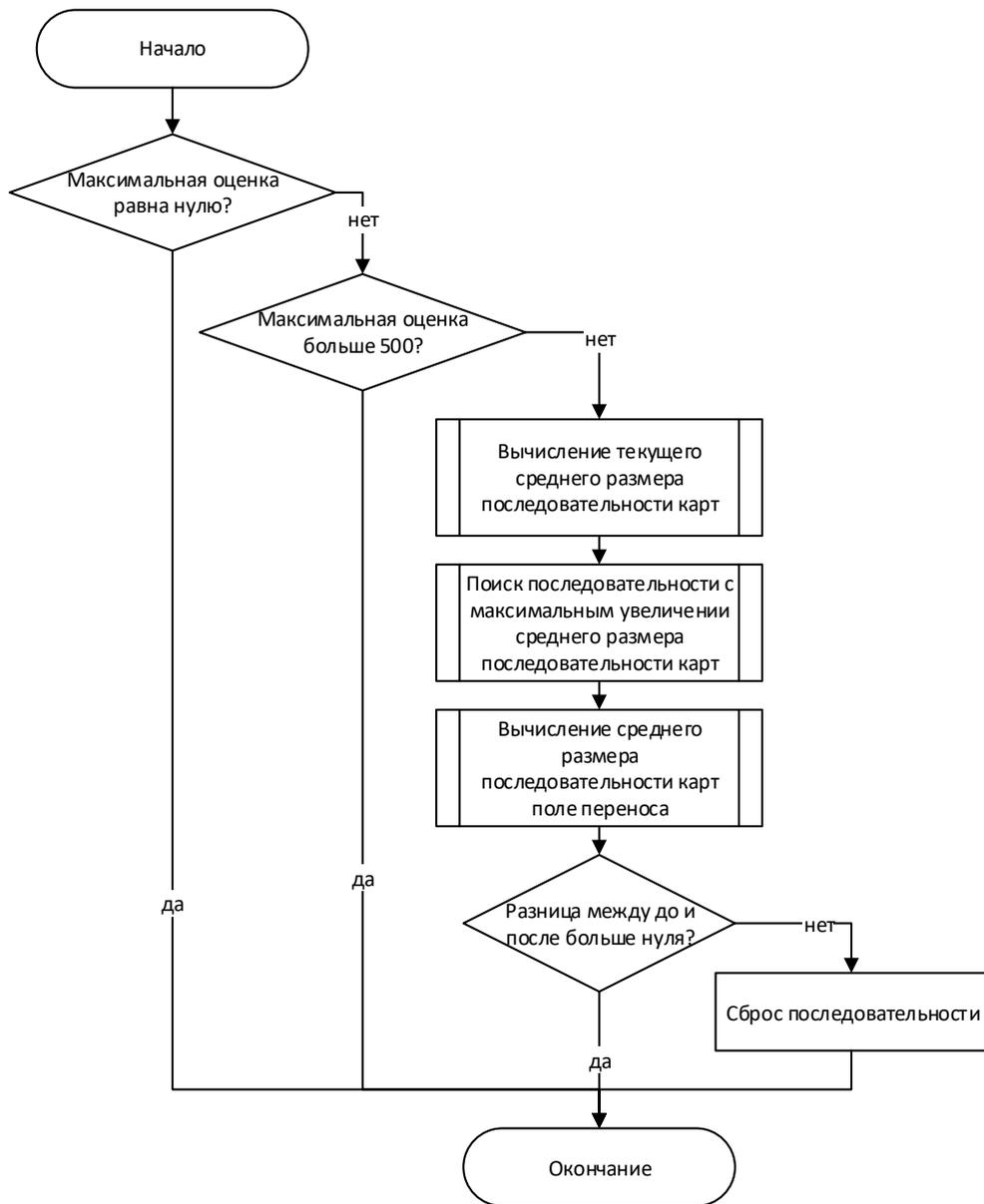
Блок-схема алгоритма стратегии игры



Блок-схема алгоритма перебора



Блок-схема алгоритма поиска оптимальной последовательности



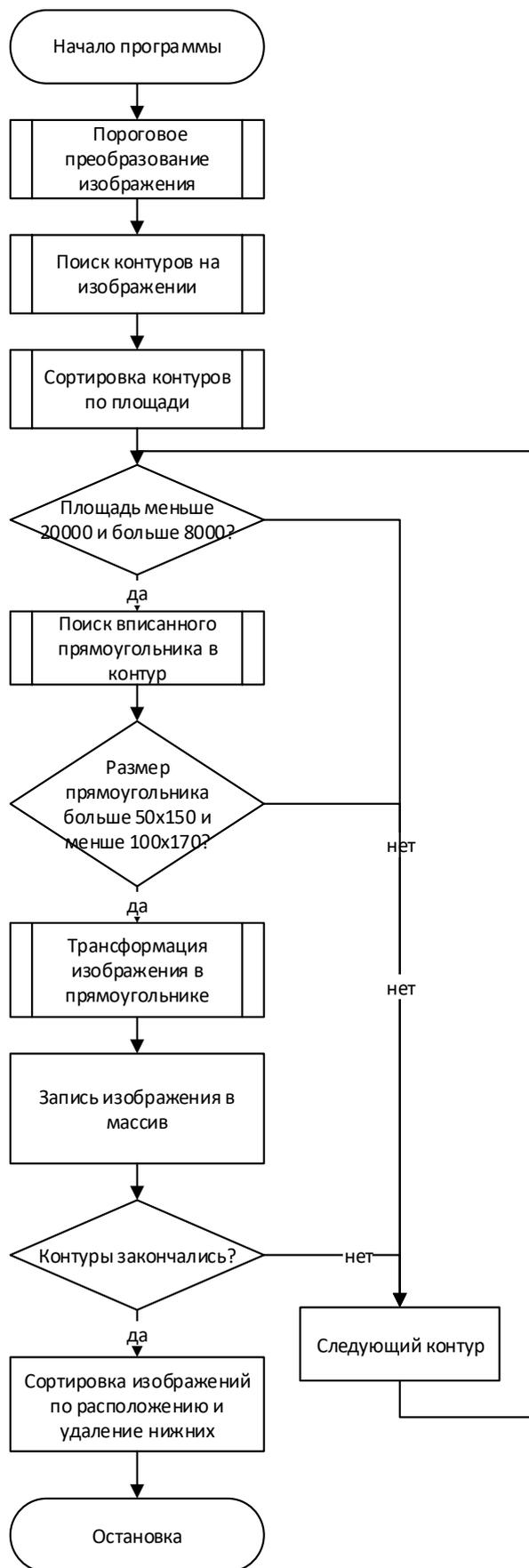
## Приложение В

(обязательное)

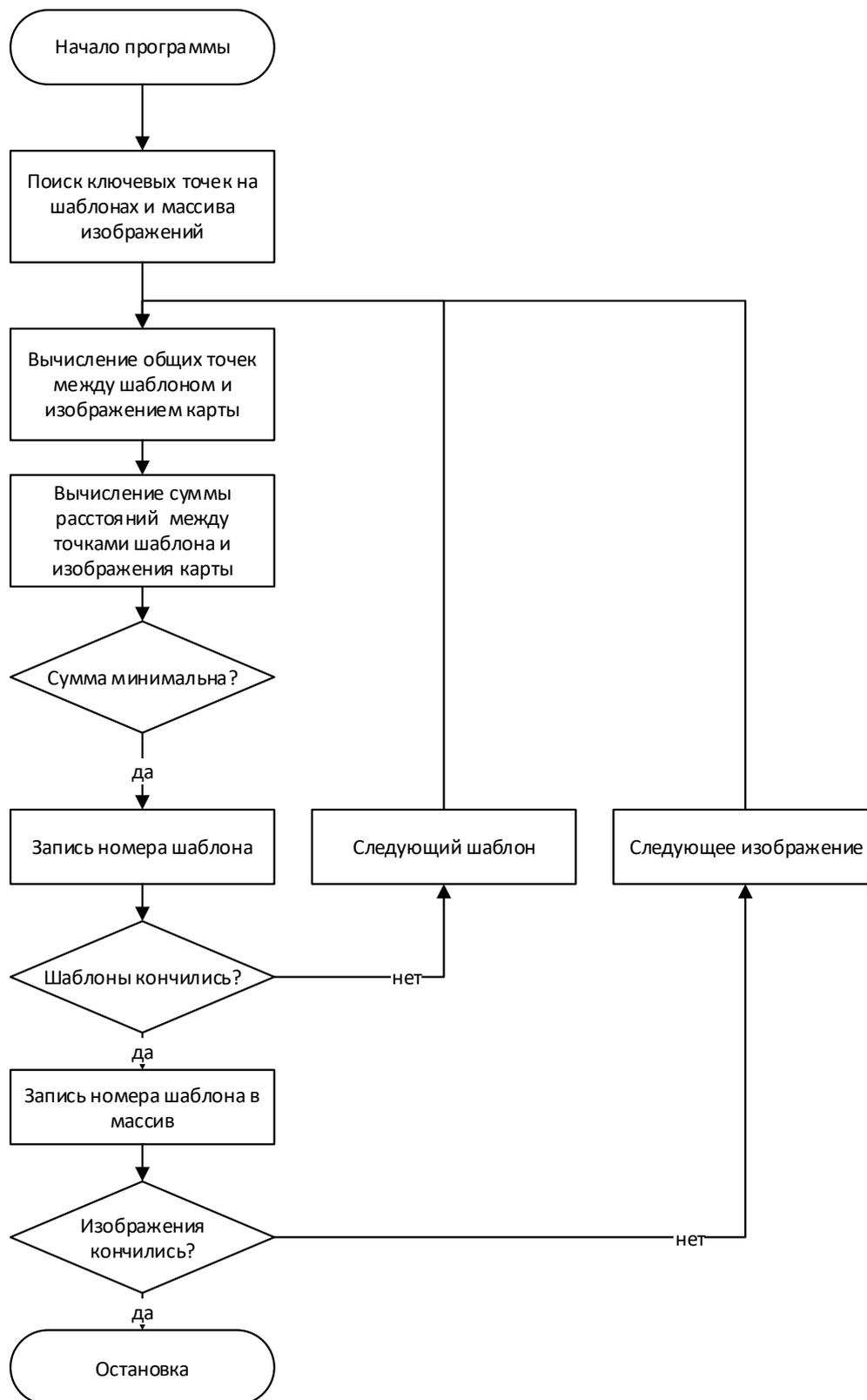
Блок-схема алгоритма распознавания карт на изображении



## Блок-схема алгоритма обработки и поиска изображений карт



# Блок- схема сравнения шаблонов и изображений карт по ключевым точкам



## Приложение Г

(необязательное)

Интерфейс класса Cards

```
class Cards
{
protected:
    int suit; .                //масть
    int name; .                //имя
    int position[2];          //расположение           position[0]=x
position[1]=y
public:
    void new_card( int, int, int, int); //создание новой карты
    void reset();              //сброс карты
    void set_suit(int );      //запись масти
    void set_name(int );      //запись имя
    void set_position(int , int ); //запись расположения
    void Write_string(char*); //преобразование в строку
    void Read_string(char*);  //преобразование из строки
    void Read_key();          // чтение из консоли
    void Print_full();        //вывод полны в консоль
    void Print_low();         //вывод в консоль
    int get_suit();           //чтение масти
    int get_name();           //чтение имя
    int get_x();              //чтение x координаты
    int get_y();              //чтение y координаты
    bool operator ==(Cards);  // оператор равенства для объектов
    bool operator !=(Cards); // оператор неравенства для объектов
    void operator =(Cards);   //оператор присваивания для
объектов
};
```

## Приложение Д

(необязательное)

### Объявление класса Steps

```
class Steps
{
protected:
    int rating_step;           //Оценка шага
    int position_start[2];     //Начальная позиция карты
    Cards card_start;         //Переносимая карта
    int position_end[2];      //Конечная позиция карты
    Cards card_end;          //Карта на которую переносят
public:
    void Reset();             //Сброс перемещения
    void New_Step(int,int,int,Cards,int,int,Cards); //Новый шаг
    void set_rating_step(int); //Запись оценки
    void add_rating_step(int); //Добавить оценку
    void set_position_start(int,int); //Запись начальной позиции
    void set_card_start(Cards); //Запись переносимой карты
    void set_position_end(int,int); //Запись конечной позиции
    void set_card_end(Cards); //Запись конечной карты

    int get_rating_step();    //Чтение оценки
    int get_position_start_x(); //Чтение начальной позиции x
    int get_position_start_y(); //Чтение начальной позиции y
    Cards get_card_start();   //Чтение переносимой карты
    int get_position_end_x(); //Чтение конечной позиции x
    int get_position_end_y(); //Чтение конечной позиции y
    Cards get_card_end();     //Чтение конечной карты

    void Print_full();        //Полный вывод в консоль
    void Print_low();        //Вывод в консоль

    bool operator ==(Steps); //оператор равенства для шагов
    bool operator !=(Steps); //оператор  неравенства  для
шагов
};
```

## Приложение Е

(необязательное)

### Объявление класса GameTable

```
class GameTable
{
protected:
    vector< vector<Cards> > card;           //Столбцы карт
    int base;                             //Число собранных стопок карт
    Cards deck;                           //Верхняя карта сброса
    int numberdeck;                       //Число стопок сброса
    int widthcards;                       //Ширина карт
    int heightcards;                     //Высота карт
public:
    void Reset();                         //Сброс поля
    void Reset_Card(int,int);             //Сброс карты

    void set_card(int,int,Cards);         //Запись карты
    void set_base(int );                 //Запись числа собранных стопок карт
    void set_deck(Cards );               //Запись верхней карта сброса
    void set_numberdeck(int );           //Запись числа стопок сброса
    void set_size(int ,int);             //Запись размеров карт

    Cards get_card(int,int);              //Чтение карты
    void add_card(vector <Cards>);        //Добавить карты
    void add_deck(vector <Cards>);        //Добавить карты после сброса
    Cards get_deck();                     //Чтение верхней карта сброса
    int get_base();                       //Чтение числа собранных стопок карт
    int get_numberdeck();                 //Чтение числа стопок сброса
    int get_widthcards();                 //Чтение ширины карт
    int get_heightcards();                //Чтение высоты карт
    int get_sizecard(int);                 //Чтение размеров столбца

    void Print_full();                    //Полный вывод в консоль
    void Print_low();                     //вывод в консоль
    void Save_to_file(char*);             //Сохранить в файл
    void Load_from_file(char*);           //Загрузить в файл

    bool true_card(int ,int);              //Проверка карты на перенос
    vector <Cards> true_cards();           //Карты которые можно переносить
};
```

```

bool true_step(int ,int,Cards); //Проверка шага
vector <Steps> true_steps(); //Шаги которые можно сделать
void transfer_step(Steps); //Выполнение переноса
bool serach_column(bool); //Поиск собранной стопки
bool open_column(); //Поиск столбца без скрытых карт
bool secret_last_card(); //Поиск столбца с последней скрытой
карты
int cloculation_raiting(int, int, int, int); //Вычисление рейтинга шага
bool complite_task(Steps*); //Проверка выполнения целей
float calculation_sequence(); //Вычисление среднего размера
последовательностей

};

```

## Приложение Ж

(необязательное)

### Объявление класса SeriesSteps

```
class SeriesSteps
{
public:
    vector <Steps> step;           //Массив шагов
    int rating_series;           //Оценка последовательности

    void Reset();                //Сброс последовательности

    void set_rating_series();     //Запись оценки
    void set_step(Steps);         //Запись шага
    int get_rating_series();      //Чтение оценки
    int get_size_step();         //Чтение размера последовательности
    void transfer_steps(GameTable *); //Перенос карт по шагам
    void print_series();         //Вывод последовательности
    bool serach_step (Steps);     //Поиск шага
    bool search_seras(vector<SeriesSteps>); //Поиск последовательности
    void delet_loop ();          //Удаление петель
};
```

## Приложение 3

(необязательное)

Код передающей платы

```
#include <Wire.h>

void setup()
{
  Wire.begin();
  Wire.setClock(400000);
  Serial.begin(115200);
}

int get_number_button (char namebutton[10])
{
  String str(namebutton);
  if (str=="F1")
    return 194;
  else if (str=="F2")
    return 195;
  else if (str=="F3")
    return 196;
  else if (str=="F4")
    return 197;
  else if (str=="F5")
    return 198;
  else if (str=="F6")
    return 199;
  else if (str=="F7")
    return 200;
  else if (str=="F8")
    return 201;
  else if (str=="F9")
    return 202;
  else if (str=="F10")
    return 203;
  else if (str=="F11")
    return 204;
  else if (str=="F12")
    return 205;
```

```
else if (str=="SPACE")
    return 32;
else if (str=="LCTRL")
    return 128;
else if (str=="LSHIFT")
    return 129;
else if (str=="LALT")
    return 130;
else if (str=="LGUI")
    return 131;
else if (str=="RCTRL")
    return 132;
else if (namebutton=="RSHIFT")
    return 133;
else if (str=="RALT")
    return 134;
else if (str=="RGUI")
    return 135;
else if (str=="UP")
    return 218;
else if (str=="DOWN")
    return 217;
else if (str=="LEFT")
    return 216;
else if (str=="RIGHT")
    return 215;
else if (str=="BACKSPACE")
    return 178;
else if (str=="TAB")
    return 179;
else if (str=="RETURN")
    return 176;
else if (str=="ESC")
    return 177;
else if (str=="INSERT")
    return 209;
else if (str=="DELETE")
    return 212;
else if (str=="PAGEUP")
    return 211;
else if (str=="PAGEDOWN")
```

```

    return 214;
else if (str=="HOME")
    return 210;
else if (str=="END")
    return 213;
else if (str=="CAPSLOCK")
    return 193;
else
    return (int)namebutton[0];
}

void to_i2c(int sum,int numberbutton)
{
    Wire.beginTransmission(8); // transmit to device #8
    Wire.write(sum);
    Wire.write(numberbutton);
    Wire.endTransmission(); // stop transmitting
}

void loop()
{
    if (Serial.available()) { //если есть принятый символ
        char buf[7]="\0";
        char key[10]="\0";
        int sum=0;

        for (int i=0;i<6;i++){
            buf[i] = Serial.read(); // то читаем его и сохраняем в val
        }
        String str(buf);
        Serial.println(str);
        if (str=="Press:"){
            sum=2;
        }else
        if (str=="Print:"){
            sum=1;
        }else
        if (str=="Reset:"){
            sum=3;
        }
        if (str=="AltTab"){

```

```

    sum=4;
}
Serial.println(String(sum));

for (int i=0;i<10;i++){
    if (Serial.available())
    {
        char flag=Serial.read();
        if (flag!='\0')
            key[i] = flag; // то читаем его и сохраняем в val
        else
        {
            key[i]='\0';
            break;
        }
    }
    else
    {
        key[i]='\0';
        break;
    }
}
String str1(key);
Serial.println(str1);
int num=get_number_button(key);
Serial.println(String(num));
to_i2c(sum,num);
}
else
{
    Serial.write('1');
    delay(100);
}
}

```

## Приложение И

(необязательное)

Код выполняющей платы

```
#include "Keyboard.h"
#include <Wire.h>

int x=0;
int num=0;

void setup() {
  Wire.begin(8);          // join i2c bus with address #8
  Wire.setClock(400000);
  Wire.onReceive(receiveEvent); // register event
  Keyboard.begin();
  Serial.begin(115200);
}

void loop() {
  delay(100);
}

void receiveEvent(int howMany) {
  x = Wire.read(); // receive byte as an integer
  Serial.println(String(x));
  num = Wire.read(); // receive byte as an integer
  Serial.println(String(num));
  if (x==1){
    Serial.println(String(num));
    Keyboard.write(num);
    x=0;
  }
  else if(x==2) {
    Serial.println(String(num));
    Keyboard.press(num);
    x=0;
  }
  else if (x==3){
    Serial.println(String(num));
    Keyboard.releaseAll();
  }
}
```

```
    x=0;
  }
  else if (x==4){
    Serial.println("ALT+TAB");
    Keyboard.press(KEY_LEFT_ALT);
    for (int i=0; i<(num-48);i++)
    {
      Keyboard.press(KEY_TAB);
      Keyboard.release(KEY_TAB);
      delay(50000);
    }
    Keyboard.releaseAll();
    x=0;
  }
}
```

# Приложение К

(необязательное)

Блок-схема программы робота

