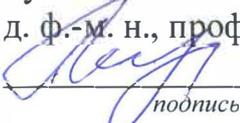


Министерство науки и высшего образования Российской Федерации  
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ (НИ ТГУ)  
Физико-технический факультет

ДОПУСТИТЬ К ЗАЩИТЕ В ГЭК

Руководитель ООП

д. ф.-м. н., профессор

  
Г.Р. Шрагер

подпись

« \_\_\_\_\_ » 2022г

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА МАГИСТРА  
(МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ)

РАЗРАБОТКА АЛГОРИТМА ОБХОДА ПРЕПЯТСТВИЙ РОБОТОМ  
ОР-3

по направлению подготовки 15.04.06 Мехатроника и робототехника  
направленность (профиль) «Моделирование робототехнических систем

Лопатин Лев Дмитриевич

Руководитель ВКР

канд. техн. наук, доцент

  
С. А. Волков

подпись

« \_\_\_\_\_ » 2022г.

Автор работы

студент группы № 102013

  
Л.Д. Лопатин

подпись

« \_\_\_\_\_ » 2022 г.

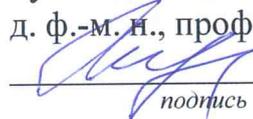
Министерство науки и высшего образования Российской Федерации  
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ (НИ ТГУ)

Физико-технический факультет

ДОПУСТИТЬ К ЗАЩИТЕ В ГЭК

Руководитель ООП

д. ф.-м. н., профессор

 Г.Р. Шрагер

подпись

« \_\_\_\_\_ » \_\_\_\_\_ 202\_ г.

ЗАДАНИЕ

по выполнению выпускной квалификационной работы магистра обучающемуся

Лопатину Льву Дмитриевичу

по направлению подготовки 15.04.06 Мехатроника и робототехника, образовательная программа «Моделирование робототехнических систем»

1 Тема выпускной квалификационной работы

Разработка алгоритма обхода препятствий роботом OP-3

2 Срок сдачи обучающимся выполненной выпускной квалификационной работы: а) в учебный офис / деканат – \_\_\_\_\_ б) в ГЭК – \_\_\_\_\_

3 Исходные данные к работе:

Объект исследования – Робот OP-3

Предмет исследования – Алгоритмы управления движением робота

Цель исследования – Разработать алгоритм обхода препятствий роботом OP-3

3 Задачи: Изучить имеющего робота и оценить технические возможности. Изучить существующие алгоритмы обхода препятствий. Ознакомиться с фреймворком Robot Operations System(ROS) и с его принципами работы. Рассмотреть вопросы управления роботом OP-3. Изучить вопросы по внедрению разработанного алгоритма обхода препятствий в систему управления робота, путем внесения изменений в имеющуюся программу (её редактирования) или разработать новую управляющую программу контроллера. Провести испытания. Подготовить отчет по выпускной квалификационной работе.

Методы исследования:

В выпускной квалификационной работе использованы следующие методы исследования: теоретический анализ и синтез, моделирование, эксперимент, метод дедукции и индукции, метод обобщения.

Организация или отрасль, по тематике которой выполняется работа, – \_\_\_\_\_

4 Краткое содержание работы

В данной работе рассматривается создание алгоритма обхода препятствий роботом OP-3, в котором работа и создание любой управляющей программы робота происходит с помощью Robot Operations System(ROS).

Руководитель выпускной квалификационной работы

доцент, НИ ТГУ, физико-  
технический факультет

*должность, место работы*

Задание принял к исполнению

студент, НИ ТГУ, физико-  
технический факультет

*должность, место работы*

/  /

С.А.Волков

*И.О. Фамилия*

/  /

Л.Д.Лопатин

*И.О. Фамилия*

## АННОТАЦИЯ

В данной выпускной квалификационной работе речь пойдет о разработке алгоритма обхода препятствий роботом ОР-3. ВКР состоит из введения, основной части, состоящей из 4 глав, заключения, списка использованных источников и приложений.

Во введение представлена общая информация на выбранную тему, также её актуальность в современном мире. Поставлены задачи для достижения конечной цели.

В первой главе <<Обзор существующих алгоритмов обхода препятствий>> рассмотрены две основные категории алгоритмов, строящиеся на картах местности и биоинспирированные.

Во второй главе <<Описание и изучения функционала робототехнической платформы>> описывается робот ОР-3 и его технический функционал.

В третьей главе <<Программная реализация>> приведена информация про фреймворк Robot Operation System. Рассмотрены вопросы, связанные с созданием управляющей программы для работы созданного алгоритма обхода препятствий роботом.

В четвертой главе <<Испытания>> приведены результаты работы управляющей программы.

## ОГЛАВЛЕНИЯ

Введение.....	6
1. Обзор существующих препятствий.....	10
2. Описание и изучения функционала робототехнической платформы.....	22
3. Программная реализация.....	27
3.1 Robot Operation System.....	27
3.2 Создание управляющего комплекса.....	30
4. Испытания.....	34
Заключение.....	36
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ И ЛИТЕРАТУРЫ.....	37
ПРИЛОЖЕНИЯ А.....	38
ПРИЛОЖЕНИЯ Б.....	39
ПРИЛОЖЕНИЯ В.....	40
ПРИЛОЖЕНИЯ Г.....	41

## ВВЕДЕНИЕ

Концепция автоматизированных механизмов, умеющих самостоятельно выполнять любые операции, занимала умы людей еще в древние времена. Одним из примеров может служить чертеж изобретения Леонардо Да Винчи “Железный рыцарь” (рисунок 1).

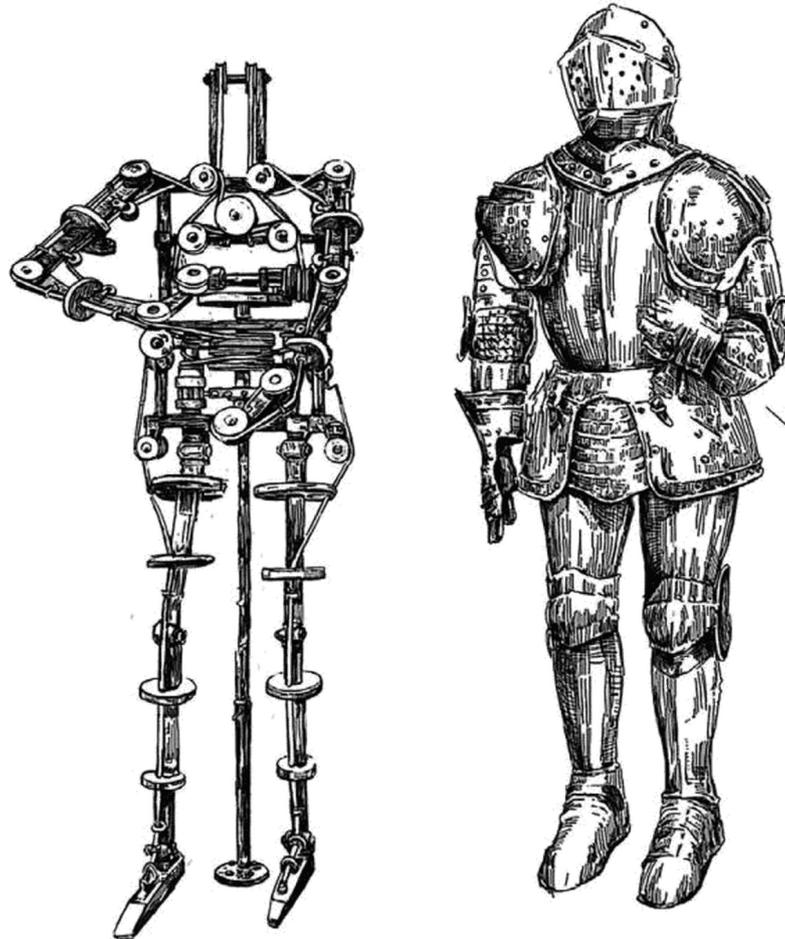


Рисунок 1 - Железный рыцарь

Современная робототехника появилась во время промышленной революции с открытием пара и электричества, давшим начало созданию силовых двигателей и механизмов. Благодаря изобретениям и открытиям Томаса Эдисона и Николы Тесла робототехника перешла в новую эпоху. В 1898 году изобретатель Никола Тесла представил свою радиоуправляемую лодку. Тем не менее, термин «Робот» впервые был упомянут чешским драматургом Карелом Чапек в своей научно-фантастической пьесе «R. U. R.», описывающей восстание роботизированных рабочих фабрики против людей-владельцев. В 1941 году Айзек Азимов придумал термин

«робототехника» и описал три закона робототехники, которые продолжают влиять на разработку робототехнических систем с искусственным интеллектом.

Вот эти законы:

- Робот не может причинить вред человеку или своим бездействием допустить, чтобы человеку был причинён вред;
- Робот должен подчиняться всем приказам человека, кроме тех случаев, когда эти приказы противоречат первому закону;
- Робот должен заботиться о своей безопасности в той мере, в которой это не противоречит первому и второму закону.

Благодаря достижениям и изобретениям XX века, таких как цифровой компьютер, транзистор и интегральная схема, у ученых появилась возможность создать программируемый мозг для робота. В современном мире роботы стали неотъемлемым явлением в жизни человека, особенно в производственной деятельности. Они решают различные задачи – от транспортировки материалов до сборки деталей.

Большая часть использования роботов приходится на:

- медицинскую область.

Например, робот KUKA LBR Med. Он оснащен сенсорной системой распознавания, которая гарантирует безопасную работу машины с человеком, простым контролем управления и специальным покрытием;

- космическую область.

Роботы активно разрабатываются и применяются человеком в освоении просторов вселенной. Для этого есть ряд основных причин: первая из них это стоимость полета. Роботизированные миссии требуют меньше затрат по сравнению со стоимостью пилотируемой программы, несут намного меньше риска для людей и дают более полезную, эмпирическую информацию;

- военную область.

Большая часть военных роботов преимущественно создаются для задач, связанных с разведкой, работой в тылу или для технической помощи в ведении боевых действий. Причиной ограниченного применения является то, что роботы недостаточно совершенны, уязвимы для ведения боевых действий непосредственно в поле и их ремонт в полевых условиях затруднен или невозможен. Но существуют образцы, которые используются непосредственно на поле боя, они, как правило, более примитивны с технологической точки зрения и управляются с помощью человека-оператора, находящегося вдали от управляемого робота или внутри. Например, робот Kuratas (рисунок 2), в котором присутствует кабина для пилота, расположенная в верхней части данного робота;



Рисунок 2 - Боевой робот Kuratas

- промышленную область.

Автоматизированные промышленные роботы применяются для многих технологических операций, требующие многократного повторения и высокой точности. Обычно такие роботы являются механизмом, напоминающем руку человека.

Вершиной современных разработок в области робототехники является создание роботов гуманоидного типа. Гуманоидный робот – это робот, напоминающий по своей конструкции человеческое тело. Такая конструкция обусловлена тем, что большинство рабочих сфер спроектированы под

человека. Да и не всегда есть возможность переоборудовать или модернизировать рабочее место в робототехнический комплекс. Также конструкция дает возможность роботу взаимодействовать с человеческими инструментами.

Для любого разрабатываемого типа робота вопросы, связанные с навигацией, играют немаловажную роль в успешном выполнении любой поставленной задачи. Казалось бы, задача очевидная: робот из точки А должен переместиться в точку В. Но в результате выполнения могут возникнуть препятствия на пути следования робота. И для решения данной задачи робот должен уметь строить маршрут, управлять параметрами движения, четко интерпретировать информацию об окружающей среде, получаемую от датчиков, и постоянно отслеживать свое местоположение в пространстве. На сегодняшний день навигационные алгоритмы малоэффективны, так как информация поступает в течении некоторого времени, и робот не обладает полной информацией о местности, в которой будет использоваться. Также в большинстве алгоритмов информация считывается с тактильных датчиков. Поэтому разрабатываются всё более новые алгоритмы обхода препятствий на основе компьютерного зрения, и данная тема еще долго не потеряет свою актуальность.

Целью данной работы является разработка алгоритма обхода препятствий роботом ОР-3.

Для достижения поставленной цели требуется решить следующие задачи – перечислить:

- Изучить алгоритмы обхода объектов, использующие в качестве входной информации изображение окружающего мира.
- Подобрать параметры для нахождения препятствий и их детектирования.
- Построить ориентиры для определения местоположения объектов
- Написать программу управления

## 1. Обзор существующих алгоритмов обхода препятствий

На данный момент большая часть роботов разрабатывается для таких отраслей, как военное дело, охрана и патрулирование специальных территорий. Роботы, разрабатываемые для этих отраслей, работают в автоматическом режиме. Автоматический режим предполагает, что робот выполняет поставленные задачи без вмешательства человека. Вследствие этого возникает ряд проблем, связанных с навигацией робота. Самый простой способ построения маршрута является прямая траектория, проложенная к требуемой точке. Этот способ построения подходит в том случае, когда в рабочем пространстве находятся только робот и цель. Но в реальных рабочих условиях могут внезапно возникнуть препятствия перед роботом. Поэтому к роботам применяется вариант поведения, называемый «избегание препятствий». Такое поведение практически является алгоритмом. В основном такие алгоритмы можно разделить на две категории.

К первой категории относятся алгоритмы, работающие с картами. Такие алгоритмы строятся по нескольким методам. Рассмотрим некоторые из них.

- Метод на основе графа.

Граф является отражением состояния, в которых может находиться робот. Таких состояний может быть  $n$ -количество и принято их называть узлами. Узел может принимать различные параметры (положения на карте, угол ориентации, скорость или ускорение) робота. Переходы между узлами являются функцией затрат. Данные функции затрат позволяют определить путь, имеющий минимальную сумму между начальной точкой и конечной точкой. На рисунке 3 представлен узловый граф расстояния (перемещения).

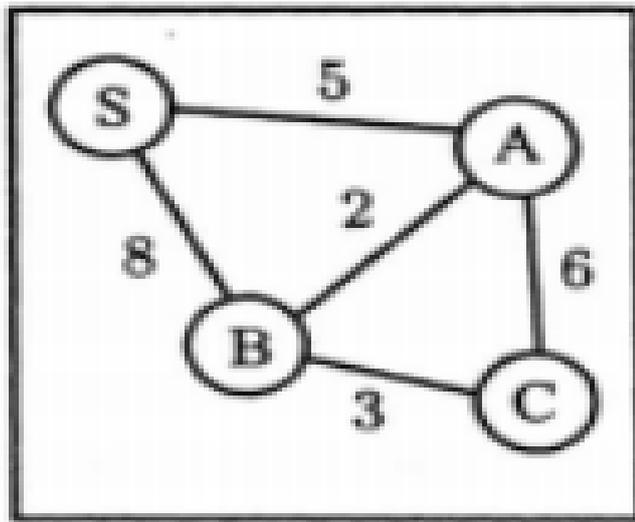


Рисунок 3 - Узловой граф расстояния

Примером метода графа может служить алгоритм Дейстера и Алгоритм А\*.

Такие алгоритмы являются эффективными в том случае, когда препятствия являются статистическими (неподвижными). Входной информацией для алгоритма является расстояние между вершинами графа.

*Алгоритм Дейстера* позволяет находить короткие пути между конкретной вершиной (источником) графа и остальными его вершинами. Рассмотрим работу алгоритма на примере ориентированного графа G (рисунок. 4)

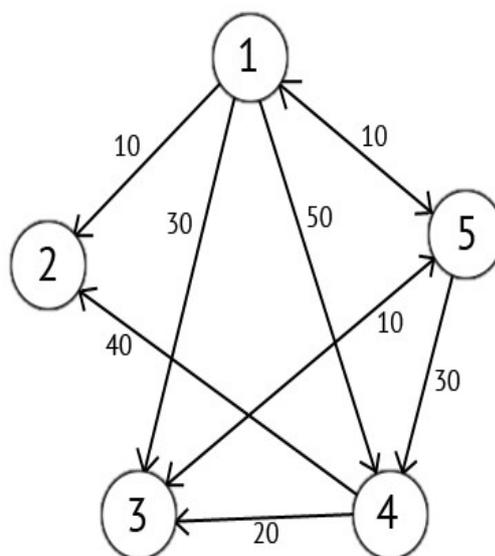


Рисунок 4 - Ориентированный граф G

В начале алгоритм составляет пустое множество, состоящее из расстояний всех вершин. Начальной точке присваивается значение 0, так как это точка является источником. Всем остальным вершинам присваиваются номер, равные бесконечности (Рисунок 5). Номера вершин 2,3,4,5 являются случайным. Бесконечность присваивается для удобства последующий расчетов и сокращение времени работы первого шага.

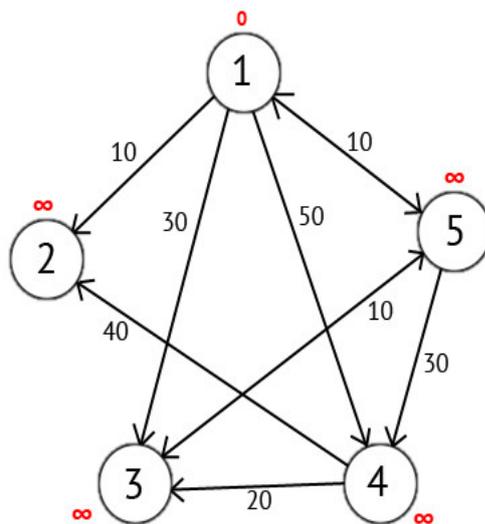


Рисунок 5 - Вершины с метками

После чего выбирается вершина с наименьшим номером. В данном примере это вершина под номером 1. Затем рассматриваются пути из вершины 1 в вершины 2, 3, 4, 5. После рассмотрения вершин 2,3,4,5 присваиваются значения, равные сумме номера и длины пути от вершины 1 в рассматриваемую вершину. В том случае, если получившиеся сумма меньше чем предыдущего значения вершины. В противном случае оставляем предыдущую вершину без изменений. (рисунок 6)

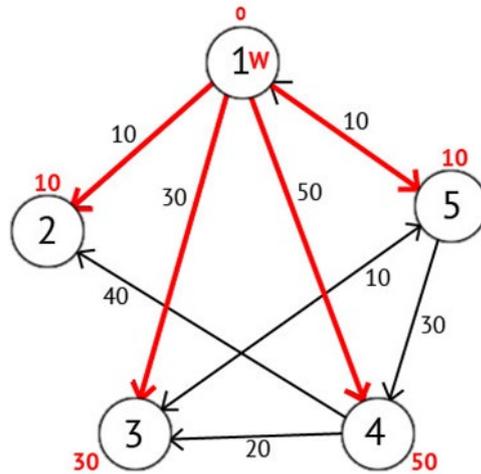


Рисунок 6 - Присвоенные метки

Далее отмечаем вершину 1 как пройденную и выбираем из еще не отмеченных (т.е. из 2, 3, 4, 5) вершину, которая имеет наименьшее значение. В нашем примере вершины 2 и 5 имеют одинаковое значение. Поэтому не имеет значения в каком мы направлении пойдём дальше. Но вершина под номером 2 является тупиковым путем, так как из нее путь не ведет к другим вершинам. Поэтому очевидной вершиной для следующего шага является вершина под номером 5. Далее рассматриваем все пути, исходящие из вершины 5, в которых еще не были. Опять находим сумму метки и длины пути. (рисунок 7)

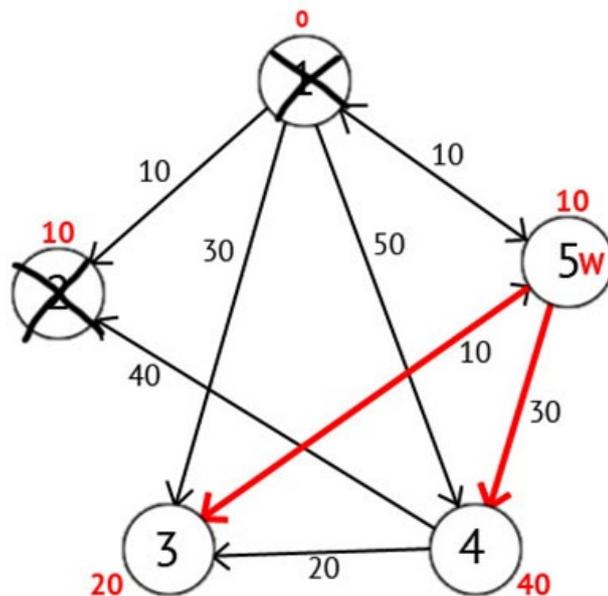


Рисунок 7 - Сумма меток и длины пути

Из рисунка ясно, что метки вершин 3 и 4 имеют наименьшую суммарную длину пути. Это означает, что был найден короткий маршрут от вершины источника до этих вершин. После этого отмечаем вершину 5 как отмеченную и выбираем следующую вершину, которая имеет минимальное значения длины пути. После того как все вершины будут найдены, алгоритм заполнить пустое множество векторами направлений и закончит свою работу. (рисунок 8).

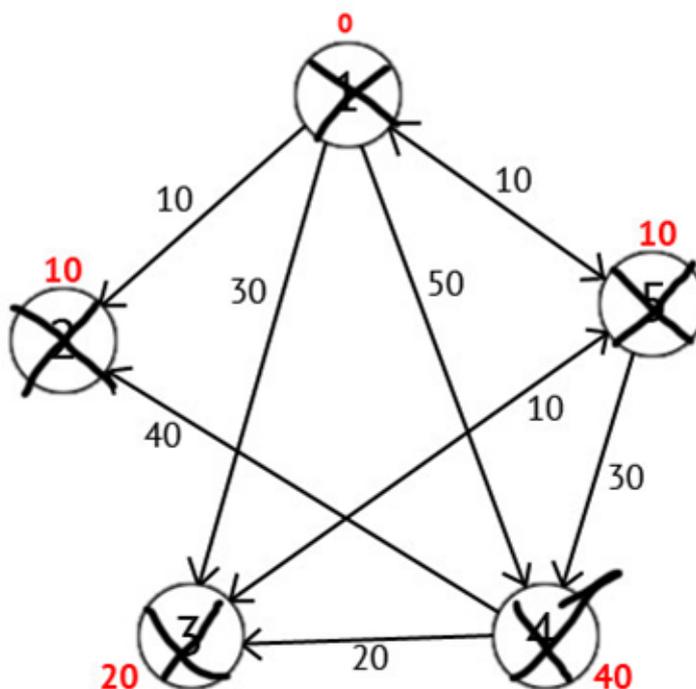


Рисунок 8 - Завершенный алгоритм в виде графа

Также ответом может быть вектор, из которого можно построить кратчайший маршрут. Вектор P будет включать в себя каждый элемент, содержащий промежуточную вершину на кратчайшем пути. На первом этапе вектор равен вершине-источнику. В данном примере  $P = \{1,1,1,1,1\}$ . После этого значения метки сравниваются с исходными данными для одной вершины. В том случае, когда метка вершины становится меньше изначальной, в вектор записывается новое значение текущей вершины по формуле:

В нашем примере 3 вершина при  $W=1$  поменяло свое значение с 30 на 20 при  $W=5$ . Поэтому третий элемент вектора будет записан в виде:  $P-P[3]=5$ . Также при  $W=5$  изменилось значение. Поэтому четвертый элемент будет записан в виде:  $P-P[4]=5$ .

Окончательный вектор равен  $P= \{1, 1, 5, 5, 1\}$ .

*Алгоритм A\**. Данный алгоритм является эвристическим уточнением вышеописанного алгоритма. Отличие заключается в том, что алгоритм  $A^*$  создает список, в котором находятся отсортированные пути до цели, из них выбирается короткий путь в данный момент времени.

- *Метод на основе клеточной декомпозиции.*

Этот метод разбивает рабочую область на множество одинаковых ячеек. Каждая ячейка закрашивается в два основных цвета: черный цвет означает, что в ячейке есть препятствия, а белый означает, что ячейка свободна от препятствия. (рисунок 9)

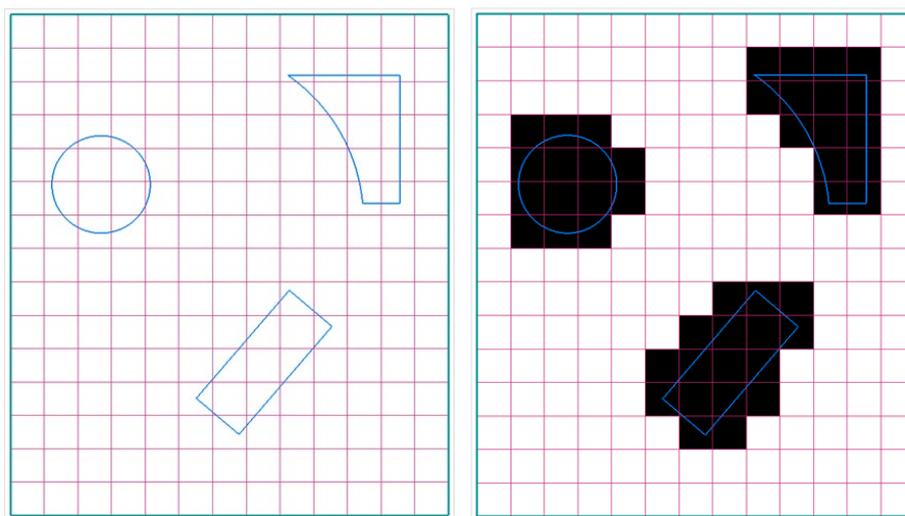


Рисунок 9 - Метод клеточной декомпозиции

*Метод потенциальных полей.*

Суть метода заключается в том, что маршрут робота строится с помощью некоторой виртуальной функции потенциального поля  $P(q)$ . Функция  $P$

состоит из двух компонент: аттрактивной и репульсивной

Аттрактивный компонент является виртуальной силой притяжения, а репульсивный - силой отталкивания.

Виртуальная сила притяжения притягивает робота к конечной точке маршрута, в то время как сила отталкивания не дает роботу слишком близко подъезжать к препятствиям и начальной точке (т.е. отталкивает робота). Значение этих сил пропорционально расстоянию до робота.

На рисунке 10 представлено движение робота с помощью функции  $P$  в самом поле.

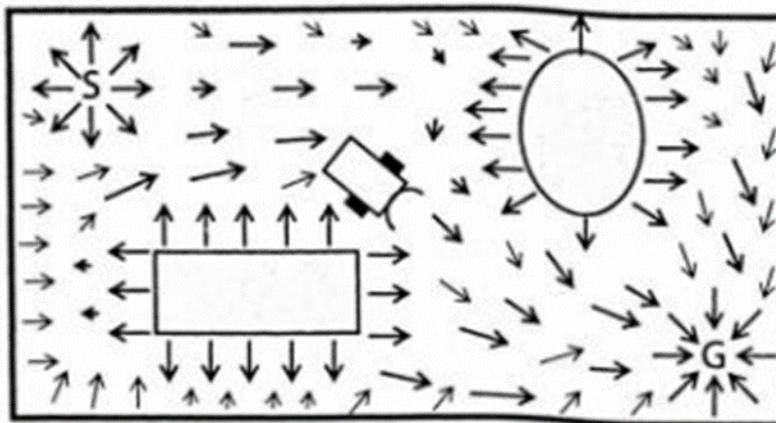


Рисунок 10 - Движение робота по методу потенциальных полей.

Данный метод имеет недостаток. Робот может заблудиться, если попадет в точки локального минимума, при которых обе силы будут равны.

Ко второй категории можно отнести биоинспирированные алгоритмы. Данный тип алгоритмов имитирует поведения или мышления человека, также некоторых биологических сообществ. Рассмотрим каждый из них.

*Муравьиный алгоритм.* Принцип действия данного алгоритма осуществляется по роевому интеллекту. Роевой интеллект описывает коллективное поведения децентрализованной самоорганизующейся системы, которая состоит из множества особей (агентов), взаимодействующих между собой и пространством вокруг себя. Рассмотрим принцип действия на примере многоагентной системы. Агентов будем называть муравьями.

Каждый муравей хранит в своей памяти список пройденных им узлов. При выборе нового узла муравей сверяется со своей памятью, после чего не рассматривает их в качестве возможных для перехода. С каждым новым проходом память дополняется новыми узлами. Данные действия составляют 1 итерацию. При новом списке (память) стирается. Также при выборе следующего узла, муравей использует не только свою память, но и “привлекательность” пути. Привлекательность зависит от двух факторов: от расстояния между узлами и следу феромона, оставленного предыдущим муравьем. При обновлении итерации изменяется только след феромона, а расстояние остается прежним.

*Искусственная нейронная сеть (ИНС).* Искусственная нейронная сеть - это математическая модель, схожая по принципу действия с сетью нервных клеток в живом организме. ИНС состоит из одинаковых элементов (нейронов). Один нейрон может иметь несколько входов и выходов. Входы и выходы соединяются в единую сеть. На вход нейронов подается сигнал, а на выходе формируется результат. Каждый нейрон обладает настраиваемыми параметрами, влияющими на работу сети. Изменение этих параметров является обучением нейросети.

Примером может служить автопилот в машине. В автопилоте используется трехслойная нейронная сеть. Первым шагом является создание двух границ, которые позволят разделить поле зрения на три зоны. После разделения задается область поиска препятствий. В данном примере (рисунок 11) для простоты обнаружения все препятствия находятся в круге.

Вторым шагом является проверка наличия препятствий в каждой зоне, входящих в круг. После чего создается массив, состоящий из трех элементов. Первый элемент хранит расстояния до препятствия в левой зоне, соответственно второй элемент относится к центральной зоне, третий элемент к правой зоне. На рисунке 11 массив состоит из трех значений ((0.8), (1.0), (0.4)). Данные значения составляют некоторый процент от максимального значения ( $\max=100\%$ ).

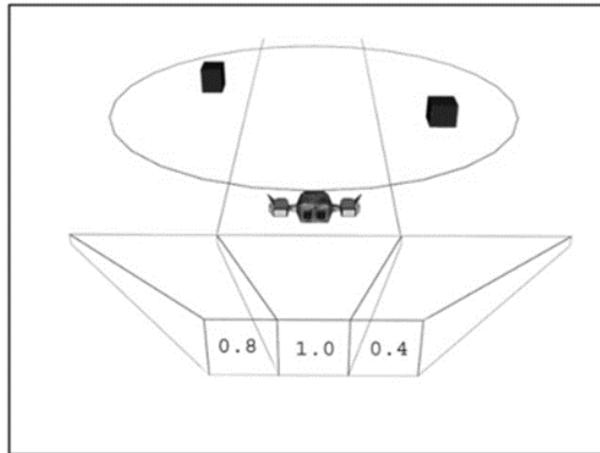


Рисунок 11 - Поле видимости машины с препятствиями, соответствующих своей зоне.

Для левой зоны это значение составляет 80% от max, а для правой зоны значение составляет 40% от max. Если значение составляет 100% от max, это значит, что в зоне нет препятствий. Эти значения корректируются во время движения. На рисунке 12 представлено поле зрения со всеми необходимыми параметрами для получения элементов массива.

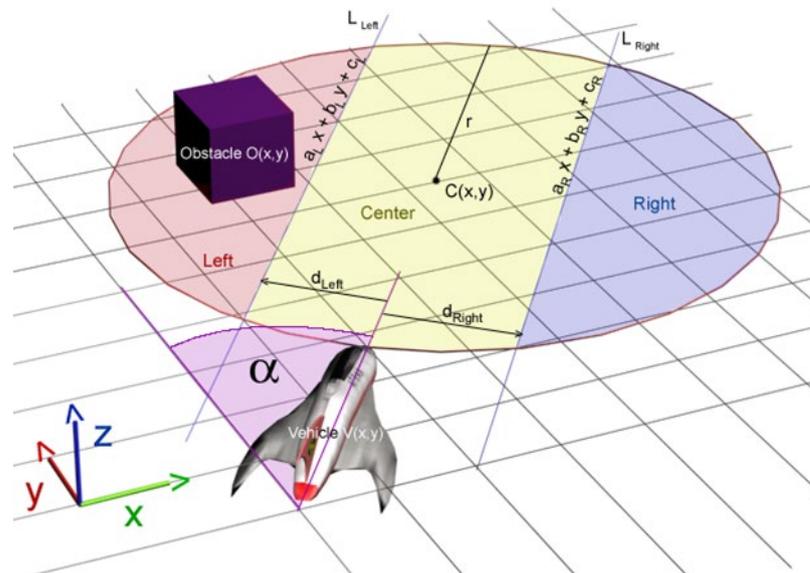


Рисунок 12 - Визуализация поле зрения с математической точки зрения

Границами зон являются линии , изображенные на рисунке 12.

Граница представляет уравнение прямой, имеющий вид:  $ax+by+c=0$ . На вход нейронной сети подается 5 параметров:

$V(x, y)$  - координаты и угол (альфа) объекта

$O(x, y)$  – координаты препятствия

$d_{left}$  и  $d_{right}$  – вектор, задающие размер центральной зоны

$r$  – радиус окружности поискового круга

Последовательность действий для получения трех элементов такова:

1. Задаем векторы  $d_{left}$  и  $d_{right}$ . Далее вычисляем координаты двух точек на границах с помощью уравнения

$$x_1 = V_x + d_x^{left}$$

$$x_2 = V_x + d_x^{right}$$

$$y_1 = V_y + d_y^{left}$$

$$y_2 = V_y + d_y^{right}$$

2. Вычисляем с:

$$C_L = -x_1 * a_L - y_1 * b_L$$

$$C_R = -x_2 * a_R - y_2 * b_R$$

где  $a_L = d_x^{left}$

$$a_R = d_x^{right}$$

$$b_L = d_y^{left}$$

$$b_R = d_y^{right}$$

3. Строим уравнения прямой для левой и правой зоны
4. Находим центр окружности по формуле:

$$C_x = V_x + r * \sin(\alpha)$$

$$C_y = V_y + r * \cos(\alpha)$$

где  $\alpha$  – угол поворота машины

5. Проводим проверку на наличие препятствия внутри круга. Если  $\sqrt{(O_x - C_x)^2 + (O_y - C_y)^2} \leq r$ , то препятствие находится в поисковой области с координатами  $O(x, y)$ . Далее проверяется в какой именно зоне находится препятствие.

Если  $a_R * O_x + b_R * O_y + C_R > 0$ , то препятствие находится в правой зоне иначе  $a_L * O_x + b_L * O_y + C_L > 0$ , то в левой части иначе по центру.

6. Находим расстояние от машины до препятствия

$$dist = \sqrt{(O_x - V_x)^2 + (O_y - V_y)^2}$$

Если препятствий на пути несколько, то повторяем 5 и 6 шаг пока не найдем все помехи на пути. Расстояние будут сохранены в массив. Первоначально

массив имеет значение  $2r$ . Если препятствие не будет найдено в любой зоне, то значение ячейки, относящиеся к конкретной зоне, остается прежней. Диапазон значения будет варьироваться от 0,0 до 1,0. Перевод расстояний в нужный формат осуществляется выражением:  $Vision[i] = Vision[i]/(2r)$ .

На выходе нейронная сеть будет выдавать два параметра. Один будет контролировать скорость. Для контроля скорости значение 0.0 – означает полную остановку, 0.5 – означает, что скорость остается неизменной, а 1.0 – означает максимальное ускорение. Второй угол поворота. Для управления значение 0.0 – означат полностью повернуть налево, 0.5 – ехать прямо, 1.0 – полностью повернуть направо. Скорость обучения нейронной сети зависит от количества нейронов, а эффективность от количества обучающей информации. В случае с обходом препятствий нейронная сеть обучается на различных вариациях размещения препятствий. В таблице 1 приведены несколько ситуаций.

Таблица 1. Варианты расположения препятствий на пути следования машины.

Входные нейроны			Выходные нейроны	
Относительное расстояние до препятствия				
Слева	По центру	Справа	Ускорение	Направление
Нет препятствий	Нет препятствий	Нет препятствий	Максимальное ускорение	Прямо
Половина пути	Нет препятствий	Нет препятствий	Небольшое ускорение	Немного правее

Нет препятствий	Нет препятствий	Половина пути	Небольшое ускорение	Немного левее
Объект рядом	Объект рядом	Объект очень рядом	Без изменений	Влево

В значениях таблица будет выглядеть так:

Таблица 2. Числовой вид таблицы 1.

Входные нейроны			Выходные нейроны		
Относительное расстояние до препятствия					
а	Слев	По центру	Справа	Ускорение	Направление
	1.0	1.0	1.0	1.0	0.5
	0.5	1.0	1.0	0.6	0.7
	1.0	1.0	0.5	0.7	0.3
	0.3	0.4	0.1	0.5	0.2

## 2. Описание и изучения функционала робототехнической платформы.

В качестве объекта был выбран антропоморфный робот OP-3 (рисунок 13). Робот является третьей моделью семейства роботов OP, разработанных компанией Robotis. Данная компания специализируется на создание обучающих роботов и комплектующих к ним. Первой моделью робота в семействе является Darwin op. Робот Darwin тоже имеет антропоморфный вид что и последующие модели в данной линейки. Линейка роботов OP построена на открытой платформе. Это означает, что робот имеет открытый исходный код и может модифицироваться в зависимости от поставленных задач. Модификация предполагает улучшение и (или) полную замену компонентов.



Рисунок 13 - Робот OP-3

Существенными изменениями в конструкции робота OP3 по сравнению с предыдущей моделью является:

- Замена сервоприводов на новые с более высоким крутящим моментом

- Увеличенный объём оперативной памяти
- Увеличенная вычислительная мощность, путем замены главного компьютера на Intel NUC 3.

- Замена субконтроллера
- Улучшенная видеокамера с более высоким разрешением.

Робот имеет 20 степеней свободы, которые обеспечиваются 20 сервоприводами. Данная модель обладает хорошей функциональностью для исследований. Данная функциональность представлена 3 режимами работы робота. В 2 из них используется компьютерное зрение. Компьютерное зрение – это область робототехники, связанная с анализом изображений и видео. Более часто встречающиеся задачи, связанные с техническим зрением, являются: сегментация изображений, детектирование объектов, классификация изображений, отслеживание подвижных объектов в реальном времени, распознавание лиц, оптическое распознавание символов, генерация изображений. Рассмотрим все три режима робота по отдельности.

➤ Автономный режим. В этом режиме робот находит красный мяч и идет к нему, после чего наносит удар нижней конечностью по нему. Также если в процессе работы робот потеряет равновесие и упадет, то он может встать и продолжить выполнения программы. Для корректного обнаружения мяча требуется сначала подогнать параметры под рабочую среду. К основным параметрам можно отнести: цвет мяча, размытие картинки, диапазон размера мяча. Выбор цвета мяча основан на HSV (цветовой моделью). Цветовая модель HSV включает в себя три настраиваемых пара: Hue-цветовой тон. Диапазон цветового тона составляет от 0 до 360 градусов. Saturation-насыщенность. Value-яркость. Диапазоны Saturation и Value совпадают и составляют от 0-100 или 0-1. В большинстве компьютерным программ значения Saturation и Value преобразовываются в целые числа. И тогда

диапазон составляет от 0 до 255. Размытие картинки увеличивает точность нахождения контуров.

➤ Режим зрения. Данный режим позволяет находить и отслеживать лица людей.

➤ Интерактивный режим. Робот выполняет заранее запрограммированные движения по порядку. Каждому движению соответствует конкретная фраза, которую робот произносит во время выполнения движений. Порядок движений и фраз, соответствующих им, представлен на рисунке 14.

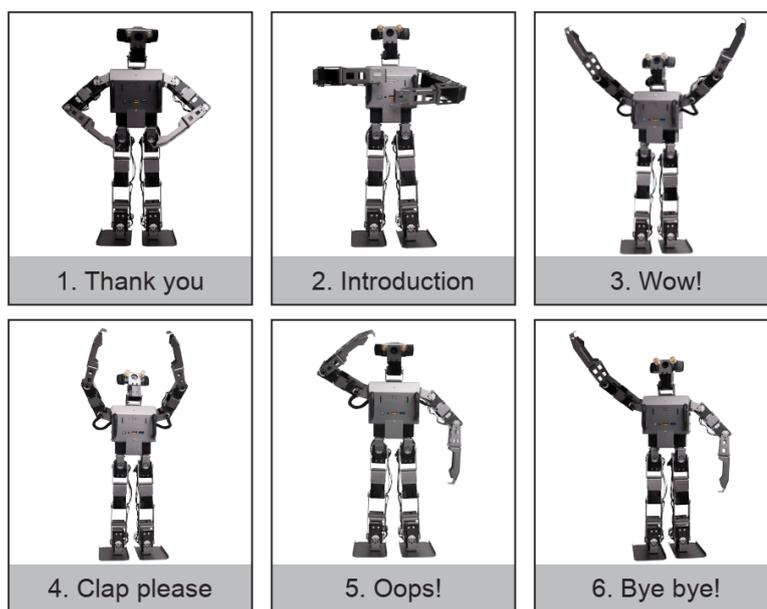


Рисунок 14 - Последовательность движений со звуковым сопровождением.

Движения и фразы можно редактировать в зависимости от желаемых целей исследователя. Добавление новых движений происходит с помощью узла редактора движений. Редактор включает в себя 256 страниц (рисунок 15). Каждая страница хранит в себе 7 шагов данных о движении.

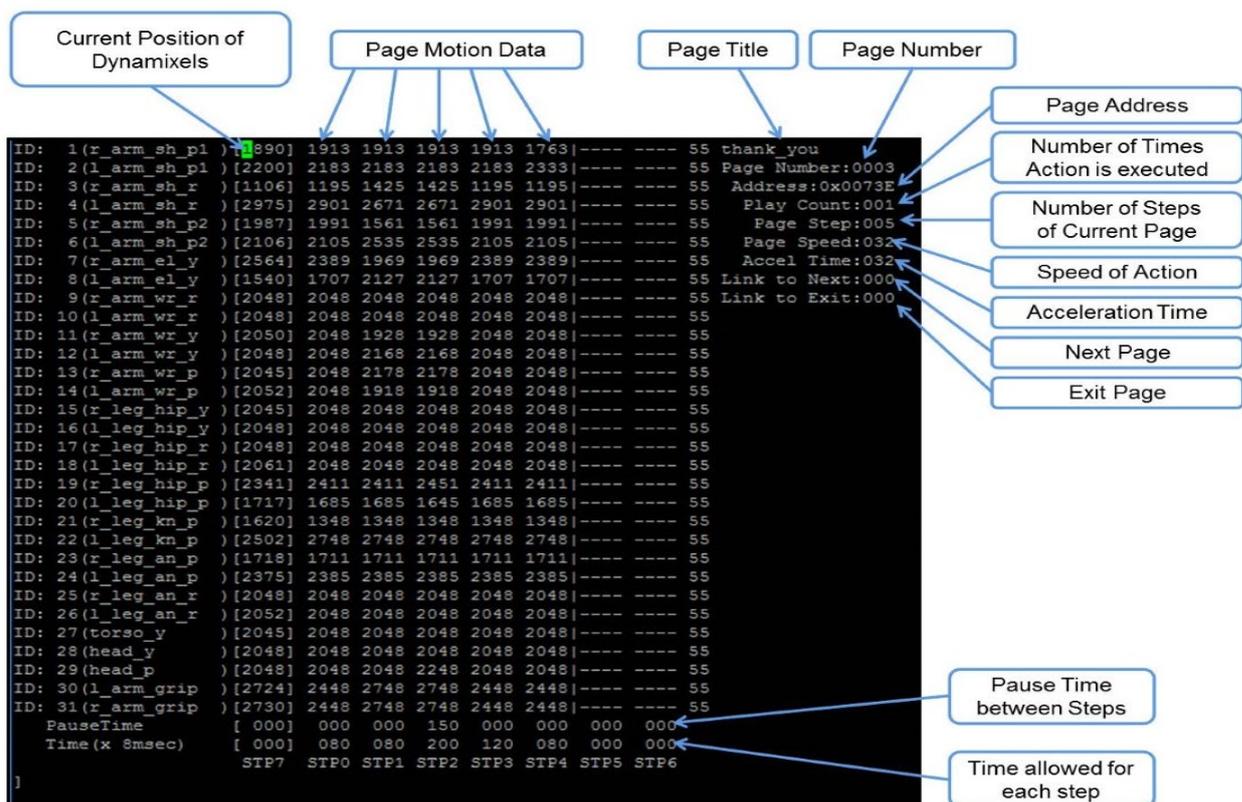


Рисунок 15 - Интерфейс страницы

Интерфейс страницы состоит из:

ID – индикационный номер каждого привода, расположенных справа.

Current position of dynamixels – текущее положения привода

Page motion data – сохранённое положения привода

Page title – название страницы

Page number – номер страницы

Page address- адрес страницы в оперативной памяти

Number of times action is executed - количество повторов воспроизведения страницы.

Number of steps of current page – количество шагов для текущей страницы

Speed of action – скорость выполнения

Acceleration time – ускорение

Next page – следующая страница. Движения могут быть разбиты на несколько страниц. Данный параметр позволяет подряд воспроизвести указанную страницу.

Exit page – выход страницы. Схожий параметр с next page. Отличие лишь в том, что после воспроизведения страницы редактор движений закрывается.

Pause time between step - время паузы между шагами

Time allowed for each step – время выполнения шага.

### 3. Программная реализация

#### 3.1 Robot Operation System

Для работы с роботом предусмотрено два типа связи: беспроводная и проводная. Беспроводной типом связи является wi-fi. Пользователь подключается к роботу через сеть Wi-Fi. Для проводной связи используется маршрутизатор, который объединяет робота и пользователя в одну локальную сеть. Операционная система, установленная в роботе, является linux mint. Для разработки системы управления любых роботов используется популярная платформа Robot Operation System(ROS). ROS – является гибким и универсальным фреймворком(платформой) для создания программного обеспечения роботов. Универсальность заключается в совместные разработки программного обеспечения для любой поставленной цели. Зачастую решение многоуровневых задач требует несколько человек. ROS позволяет нескольким командам одновременно работать над решением конкретной задачи, лишь с небольшими изменениями в исходном коде. Свою популярность среди многих ROS приобрел благодаря тому, что он является бесплатным программным обеспечением и позволяет работать с уже существующими пакетами, также создавать свои собственные пакеты для роботов, поддерживаемых ROS. Еще одним плюсом данного фреймворка является широкий спектр возможностей для визуализации поведения робота в любых ситуациях, которые могут возникнуть в результате работы. Для оценки поведения робота в реальном мире посредством визуализации в ROS имеется 2 приложения: Gazebo и rviz.

Ros Gazebo – это 3D симулятор, учитывающий разнообразные физические параметры мира и модели робота. В симуляторе можно проводить различные эксперименты без риска повредить реальную модель робота. Модель мира является конфигурируемой. Это означает, что можно менять окружения путем добавления любых элементов в рабочую область. Также в симуляторе можно установить разные модули(плагины), имитирующие работу различных датчиков. Отличие Rviz от Gazebo

заключается в том, что данные импортируемые с симулятора gazebo, позволяют работать с реальной моделью.

Архитектура ROS представляет распределенную архитектуру системы реального времени, основанной на ориентированном графе. Вершинами графа являются узлы(node). В узлах содержится информация о функциональных частях робота и о системе контроля, планирования шагов. К функциональным частям можно отнести: датчики, приводы, камеры. Обмен информацией между узлами осуществляется с помощью тем(topic), благодаря публикации определенных сообщений. Темы являются ребрами графа. Также узлы можно объединять в пакеты, если они нужны для решения одной задачи. Пакеты - это атомарная единица среды. Структура пакета состоит из:

- Include. Содержит в себе заголовочные файлы для языка программирования C++ ;
- Msg. Содержит типы сообщений;
- Src. В папке хранится исходный код, передаваемый в другие пакеты;
- CMakeLists.txt. Cmake содержит инструкции по сборки пакета;
- Manifest.xml. В данном файле описываются свойства, созданного пакета. К свойствам манифеста относятся: имя пакета, номер версии сборки, автор, зависимости от других пакетов.

Любые пакеты могут иметь шесть типов зависимостей:

- Build Dependencies (Зависимость сборки) – определяет, необходимые пакеты для сборки созданного пакета;
- Build Export Dependencies(Зависимости экспорта сборки) – определяет, необходимые пакеты для сборки библиотек для созданного пакета;
- Execution Dependencies(Зависимости выполнения) – определяют, необходимые пакеты для запуска кода в созданном пакете;

- Test Dependencies(тестовые зависимости) - определяют дополнительные зависимости для тестовых;
- Build Tool Dependencies(Зависимости инструментов сборки) – указываются инструменты системы сборки, которые пакет собирает сам;
- Doc Dependencies (Зависимости документов) – определяется набор инструментов для создания документации пакета.

Как упоминалось выше, что обмен информации происходит с помощью тем (англ. Topic), состоящий из определенных сообщений в ROS. В свою очередь сообщения принимают набор данных. Набор данных может иметь несколько стандартных типов (целое число, число с плавающей точкой, логические и так далее), кроме стандартных также массивы данных. Структура файла сообщений состоит из пары значений: тип и имя данных (fieldtype fieldname). Строк с типом и именем данных может быть несколько в зависимости от желаемой информации, которую мы хотим получить с узла.

Типы данных ROS являются уникальными, так как они не взяты напрямую из известных языков программирования. В процессе сборки пакета происходит преобразования типа ROS к типу языка, используемого исходным кодом.

Также в сообщениях используется принцип наследования типов. Принцип наследования заключается в том, что сообщения могут содержать в себе уже созданные типы сообщений. Например, для получения позы используют тип сообщений geometry\_msgs/Pose. В этом типе конфигурация описывается двумя сообщениями: положения точки и ориентация кватерниона. Где положения точки и ориентация кватерниона это уже созданные типы данных.

Таблица 3. Стандартные типы данных ROS и представления в двух языках программирования C++ и Python.

в ROS	для C++	для Python
bool	uint8_t	bool
int8	int8_t	int
uint8	uint8_t	int
int16	int16_t	int
uint16	uint16_t	int
int32	int32_t	int
uint32	uint32_t	int
int64	int64_t	long
uint64	uint64_t	long
float32	float	float
float64	double	float
string	std::string	str
time	ros::Time	rospy.Time
duration	ros::Duration	rospy.Duration

### 3.2 Создание управляющего комплекса

В работе Robotis OP-3 система управления робота уже имеет готовый фреймворк ROBOTIS-Framework. Он включает в себя несколько главных узлов, таких как менеджер робота (Op3\_manager), контроллер (OP3\_controller) и исполняемый модуль (action\_module). При запуске робота сначала происходит запуск узла менеджера, после чего вызывается контроллер. В узле контроллера происходит инициализация всех комплектующих частей из которых состоит робот. После чего происходит выбор одного из исполняемых модулей. Для того чтобы робот мог обойти препятствия необходимо реализовать определенный алгоритм (рисунок 16). Принцип работы данного алгоритма заключается в том, что поле зрения робота разделяется на три зоны, и после обнаружения препятствия происходит проверка его площади(sq) и принадлежность к одной из трех зон, путем сравнения значений координат контуров объекта(p) с диапазонами

координат границы зоны. Данные диапазоны зависят от разрешения видеокamеры. Для камеры Logitech C920 HD PRO разрешение составляет 1920x1080. Для работы алгоритма достаточно ширины камеры. Значение площади объекта могут варьироваться в зависимости от расстояния обнаружения объекта. Также значения площади было получено эмпирическим путем.

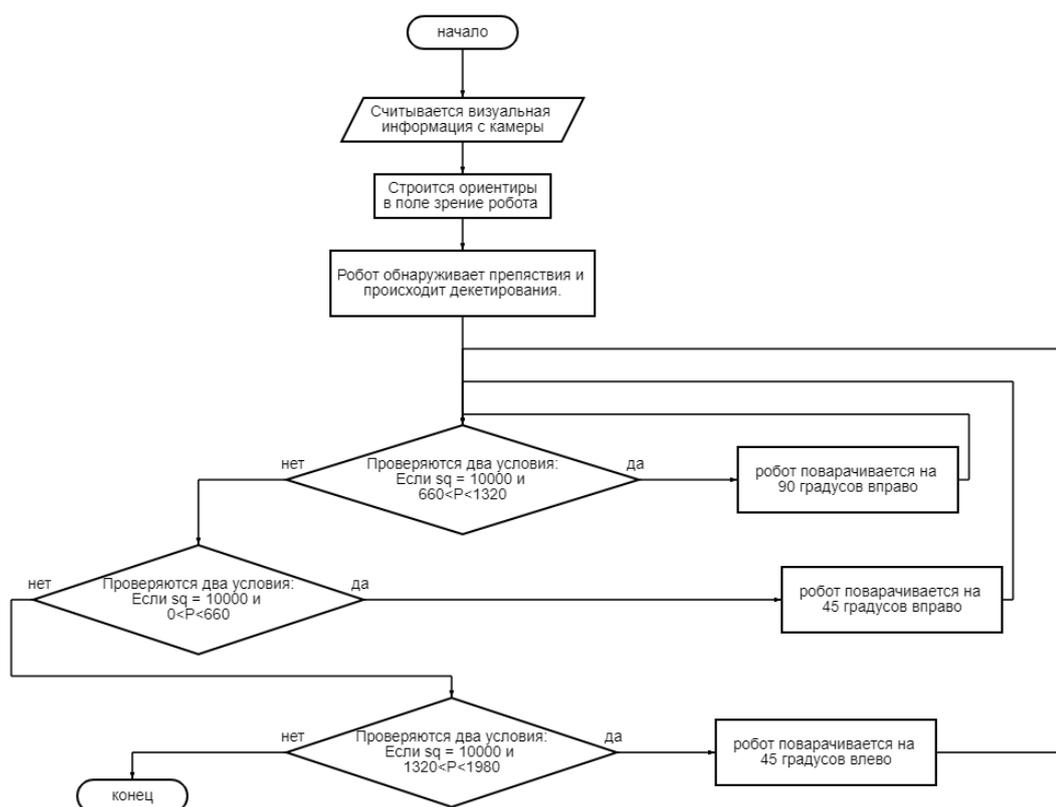


Рисунок 16 - Алгоритм

Программным путем можно реализовать с помощью добавления модуля обработки визуальной информации, который будет строить ориентиры в поле зрения робота и находить препятствия для дальнейшего отслеживания. После чего на основе полученных сравнений программа передает данные в модуль управления ходьбой. Выходной тип данных являются смещение по трем осям (x, y, z), угол поворота. Данные для настройки ходьбы публикуются с помощью двух тем /robotis/walking/command и /robotis/walking/set\_params Первая тема

устанавливает начало и конец ходьбы. Вторая тема устанавливает необходимые параметры ходьбы. Схема управления для этого алгоритма будет выглядеть следующим образом:

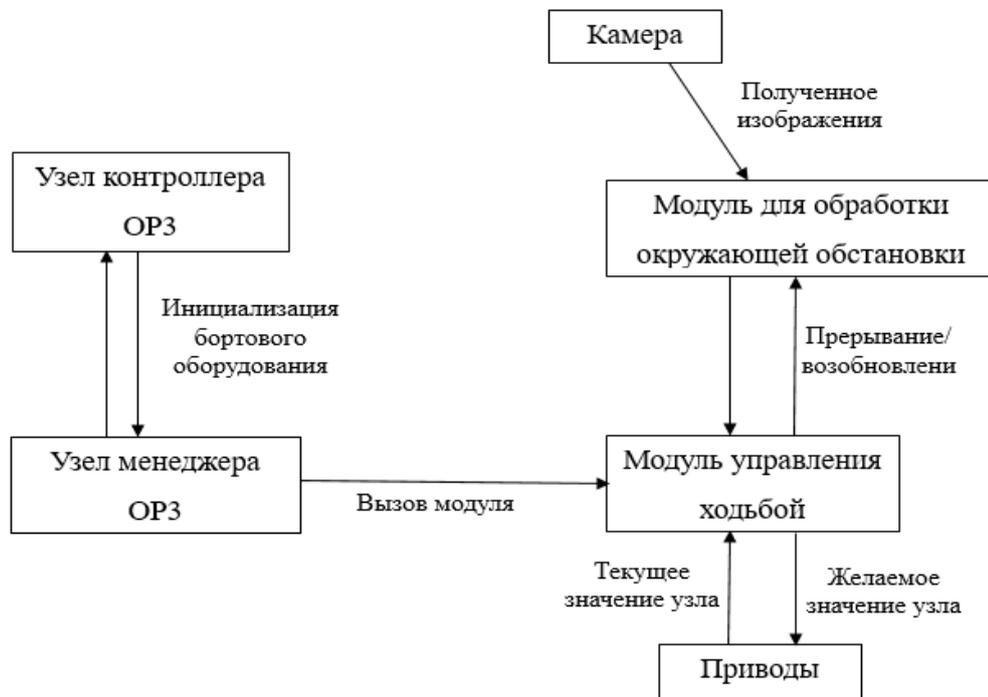


Рисунок 17 - Схема управления робота с добавлением модуля обработки окружающей обстановки.

В качестве основного языка программирования для создания модуля был выбран Python. Для проверки, получаемого изображения и выбора нужной кодировки используется узел `robot_vision`. Модуль для обработки окружающей среды создается с помощью пакета, который включает в себя узел (англ. Noda) под названием `detector_object`. Узел `robot_vision` будет иметь несколько зависимостей: `rospy`, `sensor_msgs`, `cv_bridge`. Узел `detector_object` будет иметь почти такие же зависимости, что и `robot_vision`. Отличие заключается в том, что к `detector_object` добавиться еще одна зависимость под названием `geometry_msgs`.

Также один узел может быть подписчиком и издателем одновременно. Для простоты отслеживания взаимосвязи между созданными узлами и проверки их работоспособности используем программную утилиту `rqt_graph`.

Данная утилита выведет на экран активные узлы и темы на которые они подписаны.

#### 4. Испытания робота

Сначала тестировалось работа отдельных узлов, входящий в схему управления робота. Тестирование узла `robot_vision`, показала, что ROS напрямую не может взаимодействовать с библиотекой компьютерного зрения `OpenCV`. Так как имеют разные форматы отображения изображений. Решением этой сложности являлось добавления библиотеки `CV_bridge` в зависимости пакета. Библиотека `CV_bridge` является посредником между ROS и `OpenCV`.

Для тестирования узла `detector_object` были выбраны два значения для нижней и верхней границы определения контура. Значение для нижней границы равнялось 100, а для верхней 255. В качестве двух объектов были выбраны две фигуры с разными цветами и два разных фона (рисунок 18).

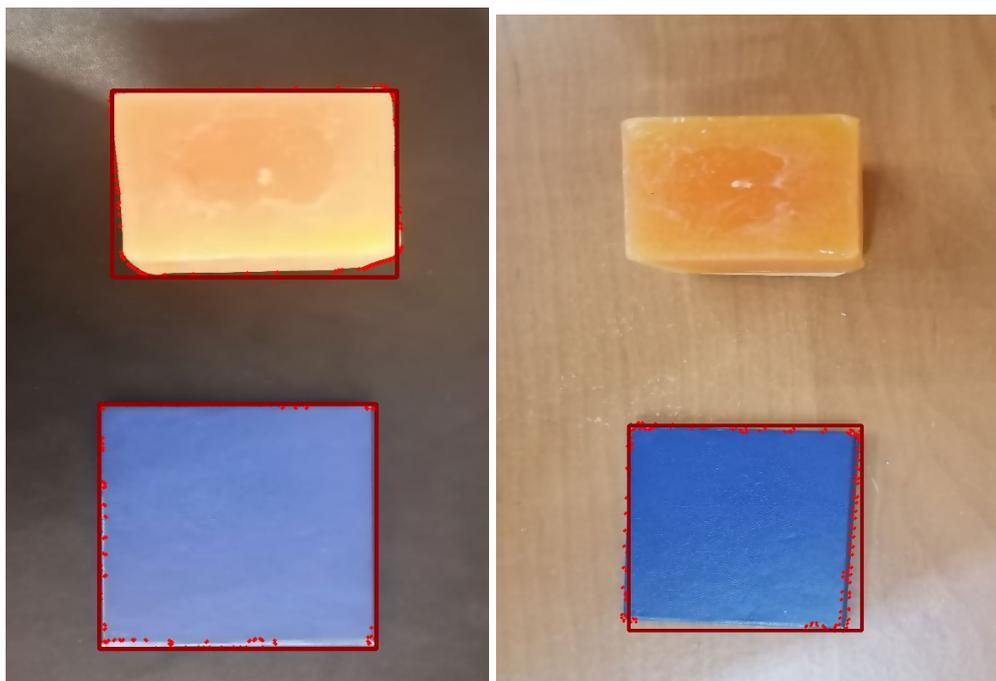


Рисунок 18 - Результат работы узла `detector_object`

Как видно из рисунка 18 в результате работы узла `detector_object` стало, то что нахождения препятствия зависит от фона, на котором оно расположено. А именно влияет контрастность. Контрастность – это отношения яркости какого-либо элемента изображения к яркости фона. Также на конечный результат влияет выбор фильтрации для определения контуров

объекта. В качестве фильтрации была выбрана двухсторонняя фильтрация, так как она лучше удаляет шумы и сохраняет четкость краев объекта.

Во время тестирования модуля управления ходьбой было замечено, что робот имеет небольшую погрешность со значением, получаемым с помощью визуализации с реальными. Еще одним результатом тестирования стало, то что после непрерывной работы робота за несколько часов, корпус робота начинает греться.

В качестве расширения функционала поведения робота можно усложнить алгоритм обхода препятствий путем добавления дополнительных условий. Допустим если высота робота превышает высоту препятствия и в зависимости на сколько превышает, робот может принять решения проползти под объектом. В противном если высота объекта меньше чем высота самого робота, то он может перешагнуть или перепрыгнуть.

## ЗАКЛЮЧЕНИЕ

Одной из важных проблем в современном мире остаются навигация роботов в мире людей, особенно антропоморфных, имеющих в своем арсенале для ориентации на местности только систему технического зрения.

В результате проделанной работы были изучены следующие вещи:

- Изучены существующие алгоритмы обхода препятствий
- Изучен Фреймворк Robotis Operation System и особенности его работы.
- Разработан алгоритм работы обхода препятствий роботом
- Составлена схема управления
- Разработаны управляющие программы на языке Python
- Предложено методика по расширению функционала поведения робота ОР-3

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ И ЛИТЕРАТУРЫ

1. <https://zen.yandex.ru/media/id/5a20825dad0f22233a285e05/istoriia-razvitiia-robototekniki-5a82d2211410c33286ea1e01>
2. <https://top3dshop.ru/blog/the-latest-medical-robots.html>
3. <https://cyberleninka.ru/article/n/metody-planirovaniya-puti-v-srede-s-prepyatstviyami-obzor/viewer>
4. [https://studref.com/548927/tehnika/neyro\\_nechetkaya\\_model\\_avtopilota\\_transportnogo\\_sredstva\\_rezhime\\_realnogo\\_vremeni](https://studref.com/548927/tehnika/neyro_nechetkaya_model_avtopilota_transportnogo_sredstva_rezhime_realnogo_vremeni)
5. <https://emanual.robotis.com/docs/en/platform/op3/tutorials/#how-to-create-the-motions>
6. <http://docs.voltbro.ru/starting-ros/ros-about.html>
7. <http://wiki.ros.org/Manifest>
8. <http://docs.voltbro.ru/starting-ros/messaging/message.html>
9. [https://docs.ros.org/en/api/sensor\\_msgs/html/msg/Image.html](https://docs.ros.org/en/api/sensor_msgs/html/msg/Image.html)  
<http://arch.altspu.ru/files/lection4.pdf>
10. [http://wiki.ros.org/op3\\_walking\\_module](http://wiki.ros.org/op3_walking_module)
11. <https://www.dmitrymakarov.ru/python/numpy-09/>
12. <https://pythonclass.ru/python/python-massivy-biblioteka-numpy/>
13. <https://pythonru.com/osnovy/if-else>

## ПРИЛОЖЕНИЕ А

### Узел подписчика(subscriber)

```
1 #!/usr/bin/env python
2 # license removed for brevity
3 import rospy
4 from std_msgs.msg import String
5
6 def talker():
7     pub = rospy.Publisher('chatter', String, queue_size=10)
8     rospy.init_node('talker', anonymous=True)
9     rate = rospy.Rate(10) # 10hz
10    while not rospy.is_shutdown():
11        hello_str = "hello world %s" % rospy.get_time()
12        rospy.loginfo(hello_str)
13        pub.publish(hello_str)
14        rate.sleep()
15
16 if __name__ == '__main__':
17     try:
18         talker()
19     except rospy.ROSInterruptException:
20         pass
```

## ПРИЛОЖЕНИЕ Б

### Издателем(publisher)

```
1 #!/usr/bin/env python
2 import rospy
3 from std_msgs.msg import String
4
5 def callback(data):
6     rospy.loginfo(rospy.get_caller_id() + "I heard %s", data.data)
7
8 def listener():
9
10    # In ROS, nodes are uniquely named. If two nodes with the same
11    # name are launched, the previous one is kicked off. The
12    # anonymous=True flag means that rospy will choose a unique
13    # name for our 'listener' node so that multiple listeners can
14    # run simultaneously.
15    rospy.init_node('listener', anonymous=True)
16
17    rospy.Subscriber("chatter", String, callback)
18
19    # spin() simply keeps python from exiting until this node is
20    # ready to be destroyed.
21    rospy.spin()
22
23 if __name__ == '__main__':
24     listener()
```

## ПРИЛОЖЕНИЕ В

Узел robot\_vision

```
#!/usr/bin/env python

import rospy
from sensor_msgs.msg import Image
from cv_bridge import CvBridge
import cv2

def callback(data):
    br = CvBridge()
    rospy.loginfo("receiving video frame")
    current_frame = br.imgmsg_to_cv2(data)
    cv2.line(current_frame, (213,0),(213,480),(0,0,255),2)
    cv2.line(current_frame, (426,0),(426,480),(0,0,255),2)
    cv2.waitKey(1)
    cv2.imshow("zona_metok", current_frame)

def receive_message():
    rospy.init_node('image_obzor', anonymous=True)
    rospy.Subscriber('usb_cam_node/image_raw', Image, callback)
    rospy.spin()
    cv2.destroyAllWindows()

if __name__ == '__main__':
    receive_message()
```

## ПРИЛОЖЕНИЕ Г

### Узел detector\_object

---

```
#!/usr/bin/env python

import rospy
import cv2
import numpy as np
from sensor_msgs.msg import Image
from cv_bridge import CvBridge
from geometry_msgs.msg import Twist
|
def nothing(x):
    pass

def callback(data):
    br = CvBridge()
    rospy.loginfo("receiving video frame")
    cv2.namedWindow("track")
    cv2.createTrackbar("T1", "track", 0, 255, nothing)
    cv2.createTrackbar('T2', 'track', 0, 255, nothing)
    kernel = np.ones((5,5))
    current_frame = br.imgmsg_to_cv2(data)
    res_img = cv2.resize(current_frame,(1980,1080), cv2.INTER_NEAREST)

    frame = cv2.bilateralFilter(res_img, 9, 75, 75)
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)

    thresh1 = cv2.getTrackbarPos ("T1", "track")
    thresh2 = cv2.getTrackbarPos ("T2", "track")

    canny=cv2.Canny(gray,thresh1,thresh2)
    dil=cv2.dilate(canny,kernel,iterations=1)
    _,contours,h=cv2.findContours(dil,cv2.RETR_EXTERNAL,cv2.CHAIN_APPROX_NONE)
    for contour in contours:
        sq = cv2.contourArea(contour)
        if sq > 10000:
            cv2.drawContours(frame,contour, -1, (0 ,0 ,158), 3)
            p = cv2.arcLength(contour, True)
            num= cv2.approxPolyDP(contour, 0.03*p, True)
```

```

x,y,w,h =cv2.boundingRect(num)
cv2.rectangle(frame, (x,y),(x+w,y+h),(0,78,100),3)

rows=len(num)
for i in range(rows):
    if(num[0][-1][0]>0 and num[rows-1][-1][0]<660):
        move_cmd=Twist()
        move_cmd.linear.x=1.0
        move_cmd.linear.y=0.0
        move_cmd.angular.z=0.45
        continue
    else:
        if (num[0][-1][0]>660 and num[rows-1][-1][0]<1320):
            move_cmd=Twist()
            move_cmd.linear.x=0.0
            move_cmd.linear.y=0.0
            move_cmd.angular.z=90.0
            continue
        else:
            if (num[0][-1][0]>1320 and num[rows-1][-1][0]<1980):
                move_cmd=Twist()
                move_cmd.linear.x=0.0
                move_cmd.linear.y=0.0
                move_cmd.angular.z=-45.0
                continue
            else:
                move_cmd=Twist()
                move_cmd.linear.x=0.0
                move_cmd.linear.y=0.0
                move_cmd.angular.z=0.0
                continue

cv2.waitKey(1)
cv2.imshow('frame', frame)
cv2.imshow('dil', dil)
cv2.imshow('canny', canny)
print(move_cmd)

def receive_message():
    rospy.init_node('object', anonymous=True)
    rospy.Subscriber('usb_cam_node/image_raw', Image, callback)
    rospy.Subscriber('usb_cam_node/camera_info', Image, callback)
    ygol_publisher=rospy.Publisher('move_cmd', Twist,queue_size=10)
    while not rospy.is_shutdown():
        rot_msg=move_cmd
        ygol_publisher.publish(Twist(rot_msg))
        rospy.spin()
    cv2.destroyAllWindows()

if __name__ == '__main__':
    receive_message()

```

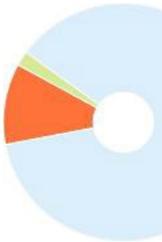
# Отчет о проверке на заимствования №1



Автор: Лопатин Лев Дмитриевич

Проверяющий: Лопатин Лев ([apach490@gmail.com](mailto:apach490@gmail.com) / ID: 8095676)

Отчет предоставлен сервисом «Антиплагиат» - <http://users.antiplagiat.ru>

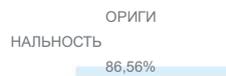
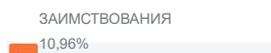


## ИНФОРМАЦИЯ О ДОКУМЕНТЕ

№ документа: 6  
Начало загрузки: 07.06.2022  
06:47:50  
Длительность загрузки: 00:00:03  
Имя исходного файла: Лопатин  
ВКР  
магистратура .pdf  
ВКР  
Название документа: Лопатин  
магистратура  
Размер текста: 1 кБ  
Символов в тексте: 38750  
Слов в тексте: 4850  
Число предложений: 353

## ИНФОРМАЦИЯ ОБ ОТЧЕТЕ

Последний готовый отчет (ред.)  
Начало проверки: 07.06.2022  
06:47:54  
Длительность проверки:  
00:00:31  
Комментарии: не указано  
Поиск с учетом редактирования: да  
Модули поиска: ИПС Адилет, Библиография, Сводная коллекция ЭБС, Интернет  
Плюс, Сводная коллекция РГБ, Цитирование, Переводные заимствования (RuEn), Переводные заимствования по eLIBRARY.RU (EnRu), Переводные заимствования по eLIBRARY.RU (KkRu), Переводные заимствования по eLIBRARY.RU (KyRu), Переводные заимствования по Интернету (EnRu), Переводные заимствования по Интернету (KkRu), Переводные заимствования по Интернету (KyRu), Переводные заимствования (KkEn), Переводные заимствования (KyEn), Переводные заимствования издательства Wiley (RuEn), eLIBRARY.RU, СПС ГАРАНТ, Медицина, Диссертации НББ, Перефразирование по eLIBRARY.RU, Перефразирование по Интернету, Перефразирование по коллекции издательства Wiley, Патенты СССР, РФ, СНГ, СМИ России и СНГ, Шаблонные фразы, Кольцо вузов, Издательство Wiley, Переводные заимствования



Заимствования — доля всех найденных текстовых пересечений, за исключением тех, которые система отнесла к цитированиям, по отношению к общему объему документа. Самоцитирования — доля фрагментов текста проверяемого документа, совпадающий или почти совпадающий с фрагментом текста источника, автором или соавтором которого является автор проверяемого документа, по отношению к общему объему документа.

Цитирования — доля текстовых пересечений, которые не являются авторскими, но система посчитала их использование корректным, по отношению к общему объему документа. Сюда относятся оформленные по ГОСТу цитаты; общеупотребительные выражения; фрагменты текста, найденные в источниках из коллекций нормативно-правовой документации. Текстовое пересечение — фрагмент текста проверяемого документа, совпадающий или почти совпадающий с фрагментом текста источника. Источник — документ, проиндексированный в системе и содержащийся в модуле поиска, по которому проводится проверка.

Оригинальность — доля фрагментов текста проверяемого документа, не обнаруженных ни в одном источнике, по которым шла проверка, по отношению к общему объему документа. Заимствования, самоцитирования, цитирования и оригинальность являются отдельными показателями и в сумме дают 100%, что соответствует всему тексту проверяемого документа. Обращаем Ваше внимание, что система находит текстовые пересечения проверяемого документа с проиндексированными в системе текстовыми источниками. При этом система является вспомогательным инструментом, определение корректности и правомерности заимствований или цитирований, а также авторства текстовых фрагментов проверяемого документа остается в компетенции проверяющего.

№ отчета	Доля	Источник	Упущено на	Акты поиска	Модуль	Комментарии
01]	,43%	3334-5355-1-SM.pdf	2 юл 2021	↓	Кольцо вузов	
02]	,91%	Диплом Беляк.pdf	7 юн 2021	↓	Кольцо вузов	
03]	%	ВКР Субач-2.docx	1 юн 2021	↓	Кольцо вузов	
04]	%	ВКР Субач.docx	6 нв 2022	⌘	Кольцо вузов	
05]	,18%	Кирилина, преддиплом. практика	0 ая 2021	⌘	Кольцо вузов	
06]	%	<a href="https://www.tsu.ru/upload/medialibrary/759/rekomenduemyy_s_hablon_programmy_gia-26.01.2021.docx">https://www.tsu.ru/upload/medialibrary/759/rekomenduemyy_s_hablon_programmy_gia-26.01.2021.docx</a> <a href="https://tsu.ru">https://tsu.ru</a>	4 нв 2022	⌘ Плюс	Интернет	
07]	,77%	не указано	3 нв 2022	⌘ фразы	Шаблонные	
08]	%	<a href="https://www.tsu.ru/upload/medialibrary/9ff/metodicheskie-ukazaniya-k-otformleniyu-rabot-obuchayushchikhsya-ni-tgu.pdf">https://www.tsu.ru/upload/medialibrary/9ff/metodicheskie-ukazaniya-k-otformleniyu-rabot-obuchayushchikhsya-ni-tgu.pdf</a> <a href="https://tsu.ru">https://tsu.ru</a>	1 нв 2022	⌘ Плюс	Интернет	
09]	%	<a href="https://lib.tsu.ru/win/produkcija/metodichka/pril1.pdf">https://lib.tsu.ru/win/produkcija/metodichka/pril1.pdf</a> <a href="https://lib.tsu.ru">https://lib.tsu.ru</a>	5 ая 2022	⌘ Плюс	Интернет	
10]	,61%	НЕЙРО-НЕЧЕТКИЕ МЕТОДЫ В ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМАХ ОБРАБОТКИ И АНАЛИЗА МНОГОМЕРНОЙ ИНФОРМАЦИИ. <a href="http://eilibrary.ru">http://eilibrary.ru</a>	9 оя 2014	⌘ ования по U	Перефразир eLIBRARY.R	
11]	%	Управление транспортным средством с помощью нейронной сети / Хабрахабр <a href="http://habrahabr.ru">http://habrahabr.ru</a>	1 нв 2017	⌘ ования по	Перефразир Интернету	

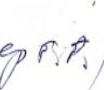
		Нейро-нечеткие методы в интеллектуальных системах обработки и					
12]	%	анализа многомерной информации НЕЙРО-НЕЧЕТКИЕ МЕТОДЫ В ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМАХ	6 ая 2016	Л	Сводная коллекция ЭБС		
13]	,35%	ОБРАБОТКИ И АНАЛИЗА МНОГОМЕРНОЙ ИНФОРМАЦИИ.  <a href="http://elibrary.ru">http://elibrary.ru</a>	9 оя 2014	Т	eLIBRARY.RU		
14]	,58%	Искусственный интеллект в сфере спорта: правовые аспекты.  <a href="http://elibrary.ru">http://elibrary.ru</a>	7 кт 2018	С	Перефразирования по eLIBRARY.RU		
15]	,08%	Управление транспортным средством с помощью нейронной сети / Хабр  <a href="https://habr.com">https://habr.com</a>	7 юн 2022	В	Интернет Плюс		
16]	,91%	Правовое обеспечение избирательного процесса в интересах национальной безопасности  <a href="http://viperson.ru">http://viperson.ru</a>	4 нв 2019	Ф	СМИ России и СНГ		
17]	%	ТЕХНОЛОГИИ MICROSOFT В ТЕОРИИ И ПРАКТИКЕ ПРОГРАММИРОВАНИЯ - PDF  <a href="https://docplayer.ru">https://docplayer.ru</a>	6 юн 2019	В	Интернет Плюс		
18]	%	Управление транспортным средством с помощью нейронной сети / Хабр  <a href="https://habr.com">https://habr.com</a>	7 юн 2022	В	Интернет Плюс		
19]	%	Управление транспортным средством с помощью нейронной сети / Хабр  <a href="https://habr.com">https://habr.com</a>	7 юн 2022	В	Интернет Плюс		
20]	%	Современные технологии. Системный анализ. Моделирование. №4  (36), 2012  <a href="http://bibliorossica.com">http://bibliorossica.com</a>	6 ая 2016	Л	Сводная коллекция ЭБС		
21]	%	171390  <a href="http://e.lanbook.com">http://e.lanbook.com</a>	0 ар 2016	Л	Сводная коллекция ЭБС		
22]	,15%	Сознание как синергетический феномен.  <a href="http://elibrary.ru">http://elibrary.ru</a>	1 ев 2021	Д	eLIBRARY.RU		
23]	%	tenyaeva_t_a_fenomen-populyarnosti-janra-sci-fi-v-kino-i-serialah-i-ego-vospriyatie-auditoriey-na-strimingovyh-se.docx	5 ая 2022	Л	Кольцо вузов		
24]	,17%	Сборник Республиканской научно-практической конференции «Третья модернизация Казахстана вклад ученых и студенческой молодежи в индустр (2/4)  <a href="https://kazutb.kz">https://kazutb.kz</a>	9 оя 2018	Т	Интернет Плюс		
25]	%	Информатика: прошлое, настоящее, будущее  <a href="http://ibooks.ru">http://ibooks.ru</a>	9 ек 2016	Д	Сводная коллекция ЭБС		
26]	%	Этические аспекты социальной робототехники.  <a href="http://elibrary.ru">http://elibrary.ru</a>	6 кт 2020	С	eLIBRARY.RU		
		Информатика: прошлое, настоящее, будущее		Д	Медицина		

27]	%		<a href="http://studentlibrary.ru">http://studentlibrary.ru</a>	9 ек 2016		
28]	%	экранизаци-	kononov_a_a_dinamika-razvitiya-obraza-robota-ot-izvestnyh-proizvedeniy-fantastiki-do-sovremennyh-fil.docx	4 юн 2020	∨	Кольцо вузов
29]	%	катастрофа?	Футурология. XXI век: бессмертие или глобальная катастрофа? <a href="http://bibliorossica.com">http://bibliorossica.com</a>	6 ая 2016	∩	Сводная коллекция ЭБС
30]	%		65631 <a href="http://e.lanbook.com">http://e.lanbook.com</a>	9 ар 2016	∩	Сводная коллекция ЭБС
31]	%	Война и словарь	[Данилевич А. А. и др.] ; под общ. ред. Дмитрия Рогозина мир в терминах и определениях военно-политический Москва печ. 2011 <a href="http://dlib.rsl.ru">http://dlib.rsl.ru</a>	5 вг 2019	∧	Сводная коллекция РГБ
32]	%	аспекты.	Искусственный интеллект в сфере спорта: правовые аспекты. <a href="http://elibrary.ru">http://elibrary.ru</a>	7 кт 2018	∩	eLIBRARY.RU
33]	%		ПРАВОВОЙ СТАТУС РОБОТОВ. <a href="http://elibrary.ru">http://elibrary.ru</a>	9 юл 2020	∨	eLIBRARY.RU
34]	%	АСПЕКТЫ	УГОЛОВНО-ПРАВОВЫЕ И КРИМИНОЛОГИЧЕСКИЕ ПРИМЕНЕНИЯ РОБОТОТЕХНИКИ. <a href="http://elibrary.ru">http://elibrary.ru</a>	1 ев 2021	∩	eLIBRARY.RU
35]	%		реферат Аникина.rtf	0 ен 2013	∩	Кольцо вузов
36]	%		НИР_РАНХиГС_Госзадание_2018_тема_19.21	2 ев 2019	∩	Кольцо вузов
37]	%		МехедкоЕВ.docx	7 ая 2022	∩	Кольцо вузов
38]	.85%	Методы планирования пути в среде с препятствиями (обзор) – тема научно-исследовательской работы в электронной библиотеке КиберЛенинка <a href="https://yandex.ru">https://yandex.ru</a>	научной статьи по кибернетике читайте бесплатно текст	9 ая 2019	∩	Интернет Плюс
39]	%	катастрофа?	Футурология. XXI век: бессмертие или глобальная катастрофа? <a href="http://studentlibrary.ru">http://studentlibrary.ru</a>	0 ек 2016	∩	Медицина
40]	%		Три закона роботехники — Википедия <a href="https://ru.wikipedia.org">https://ru.wikipedia.org</a>	7 юн 2022	∨	Интернет Плюс
41]	%		Лучшие и худшие фильмы об искусственном интеллекте <a href="https://cameralabs.org">https://cameralabs.org</a>	7 юн 2022	∨	Интернет Плюс
42]	%		Три закона роботехники — Википедия <a href="https://ru.wikipedia.org">https://ru.wikipedia.org</a>	7 юн 2022	∨	Интернет Плюс
43]	%		Рынок роботов в 2017 году перевалил за \$39 млрд	0 ек 2018	∩	СМИ России и СНГ
44]	%	общения	<a href="http://forum.guns.ru">http://forum.guns.ru</a>	5 ек 2020	∩	Интернет Плюс
45]	%		149_95_25_0_0.600_60110037 Архив новостей <a href="http://hardnsoft.ru">http://hardnsoft.ru</a>	2 ек 2020	∩	Интернет Плюс
			99_255_193_0_0.600_60106490 Архив новостей		∩	Интернет Плюс

46]	%		<a href="http://hardnsoft.ru">http://hardnsoft.ru</a>	7 ек 2020		
47]	%		Робот — Википедия <a href="https://ru.wikipedia.org">https://ru.wikipedia.org</a>	7 юн 2022	∨	Интернет Плюс
48]	%	что	«Каждый третий опасается искусственного интеллекта». Но будет, если роботы научатся шутить и сострадать? <a href="https://pravmir.ru">https://pravmir.ru</a>	7 вг 2019	∧	СМИ России и СНГ
49]	%		237_250_33_0_0.600_58487290 Скачать/Азимов - Я, робот (сборник).fb2 <a href="http://e-reading.org.ua">http://e-reading.org.ua</a>	6 ек 2020	∩	Интернет Плюс
50]	%		177_28_18_1_0.1_8773977 <a href="http://artefact.lib.ru">http://artefact.lib.ru</a>	5 нв 2022	∫	Интернет Плюс
51]	%		244_70_186_0_0.1_8773976 <a href="http://artefact.lib.ru">http://artefact.lib.ru</a>	6 ек 2020	∩	Интернет Плюс
52]	%		202_126_200_0_0.600_57809888 Фантастика 1973, 1974 <a href="http://lib.rus.ec">http://lib.rus.ec</a>	5 ек 2020	∩	Интернет Плюс
53]	%		250_123_131_0_0.1_8776158 <a href="http://artefact.lib.ru">http://artefact.lib.ru</a>	6 ек 2020	∩	Интернет Плюс
54]	%	df	<a href="http://tic.tsu.ru/www/uploads/smartsection/703_Sbornik_KRT.p">http://tic.tsu.ru/www/uploads/smartsection/703_Sbornik_KRT.p</a> <a href="http://tic.tsu.ru">http://tic.tsu.ru</a>	0 пр 2022	∧	Интернет Плюс
55]	%	df	<a href="http://tic.tsu.ru/www/uploads/smartsection/703_Sbornik_KRT.p">http://tic.tsu.ru/www/uploads/smartsection/703_Sbornik_KRT.p</a> <a href="http://tic.tsu.ru">http://tic.tsu.ru</a>	7 юн 2022	∨	Интернет Плюс
56]	%	Книга 2	246_135_7_1_0.600_57826487 Фантастика и футурология. <a href="http://lib.rus.ec">http://lib.rus.ec</a>	6 ек 2020	∩	Интернет Плюс
57]	,71%	декабрь	Права на результаты искусственного интеллекта (В.Н. Синельникова, О.В. Ревинский, журнал "Копирайт (вестник Академии интеллектуальной собственности)", N 4, октябрь-2017 г.) <a href="http://ivo.garant.ru">http://ivo.garant.ru</a>	8 ев 2018	∩	СПС ГАРАНТ
58]	%		Вагнер, Дмитрий Викторович Высоочастотные электромагнитные характеристики композиционных радиоматериалов на основе гексагональных ферритов : диссертация ... кандидата технических наук : 01.04.03 Томск 2019 <a href="http://dlib.rsl.ru">http://dlib.rsl.ru</a>	7 ек 2019	∩	Сводная коллекция РГБ
59]	%		Методы планирования пути в среде с препятствиями (обзор) – тема научной статьи по компьютерным и информационным наукам читайте бесплатно текст научно-исследовательской работы в электронной библиотеке КиберЛенинка <a href="https://cyberleninka.ru">https://cyberleninka.ru</a>	5 ая 2022	∩	Интернет Плюс
60]	%		Asimov) Список книг и других произведений Айзек Азимов (Isaac Asimov) Сортировка по году написания - LibreBook.me <a href="https://librebook.me">https://librebook.me</a>	7 юн 2022	∨	Интернет Плюс
			Искусственный интеллект: «ларец Пандоры» или новая		∫	СМИ России и СНГ

61]	%	надежда?		4 нв 2019		
62]	%	razvitiya-	<a href="https://www.ncfu.ru/export/uploads/Dokumenty-Nauka/sbornik-materialov-IV-konferencii-studencheskaya-nauka-dlya-informacionnogo-obshchestva_tom-1_28-30.04.2016.pdf">https://www.ncfu.ru/export/uploads/Dokumenty-Nauka/sbornik-materialov-IV-konferencii-studencheskaya-nauka-dlya-informacionnogo-obshchestva_tom-1_28-30.04.2016.pdf</a> <a href="https://ncfu.ru">https://ncfu.ru</a>	9 ая 2022	Л	Интернет Плюс
63]	,23%	Хабр	Осваиваем компьютерное зрение — 8 основных шагов / <a href="https://habr.com">https://habr.com</a>	7 юн 2022	И	Интернет Плюс
64]	,44%		137222 <a href="http://biblioclub.ru">http://biblioclub.ru</a>	5 пр 2016	А	Сводная коллекция ЭБС
65]	,09%	и говорить живучести и	Исследование тематики аппаратных средств для создания построения мобильных роботизированных транспортных платформ для складских логистических систем позволяет о целом ряде способов их построения с точки зрения надежности. <a href="http://vibibl.ru">http://vibibl.ru</a>	8 нв 2017	Ф	Перефразирования по Интернету
66]	,34%		История робототехники. Происхождение первых роботов <a href="https://sitekid.ru">https://sitekid.ru</a>	1 ев 2022	Д	Интернет Плюс
67]	%		История робототехники. Происхождение первых роботов <a href="https://sitekid.ru">https://sitekid.ru</a>	7 юн 2022	И	Интернет Плюс
68]	%		Топ-10: лучшие фантастические книги <a href="https://toprating.in.ua">https://toprating.in.ua</a>	7 юн 2022	И	Интернет Плюс
69]	%	(обзор) – тема научной статьи по компьютерным и информационным наукам работы в	Методы планирования пути в среде с препятствиями читайте бесплатно текст научно-исследовательской электронной библиотеке КиберЛенинка <a href="https://cyberleninka.ru">https://cyberleninka.ru</a>	8 нв 2021	Ф	Интернет Плюс
70]	,34%	политика "Московские ведомости": автореферат дис. ... кандидата исторических наук :	Байбиков, Вячеслав Юрьевич Финансово-экономическая России в оценке консервативной печати в 1881-1903 гг. : по материалам журнала "Русский вестник" и газеты 07.00.02 Сар... <a href="http://dlib.rsl.ru">http://dlib.rsl.ru</a>	5 вг 2019	А	Сводная коллекция РГБ

[71]	РОБОТ-ПОМОЩНИК «ДОМОВЕНОК» НА ПЛАТЕ АРДУИНО <a href="http://www.digitallab.ru">http://www.digitallab.ru</a>	02 Июн 2022	Интернет Плюс	
[72]	Общая биология и микробиология: методические указания по организации лабораторной и самостоятельной работы <a href="http://www.fedpress.ru">http://www.fedpress.ru</a>	20 Янв 2020	Сводная коллекция ЭБС	
[73]	Прогнозирование поведения групп автономных интеллектуальных агентов на основе теории многоагентных систем   Контент-платформа Rarbia.ru <a href="http://www.rarbia.ru">http://www.rarbia.ru</a>	07 Июн 2022	Интернет Плюс	
[74]	Все, что вам нужно знать об ИИ - за несколько минут <a href="http://www.vopros.ru">http://www.vopros.ru</a>	21 Дек 2018	СМИ России и СНГ	
[75]	Военные роботы: технологии на службе армии   Robo-Sapiens.ru   Яндекс Дзен <a href="http://zen.yandex.ru">http://zen.yandex.ru</a>	07 Июн 2022	Интернет Плюс	
[76]	Учебный план освоения компьютерного зрения за 8 недель <a href="http://www.courses.ru">http://www.courses.ru</a>	07 Июн 2022	Интернет Плюс	
[77]	Осваиваем компьютерное зрение - 8 основных шагов <a href="http://www.vopros.ru">http://www.vopros.ru</a>	17 Авг 2019	СМИ России и СНГ	
[78]	СИСТЕМНАЯ КОНЦЕПЦИЯ САМОУПРАВЛЕНИЯ	14 Янв 2019	СМИ России и СНГ	Источник исключен. Причина: Маленький процент пересечения.
[79]	Создана система навигации для роботов, курсирующих в организме человека <a href="http://www.vopros.ru">http://www.vopros.ru</a>	09 Ноя 2021	Интернет Плюс	Источник исключен. Причина: Маленький процент пересечения.
[80]	Реферат/Курсовая: Зарубежный опыт и рекомендации по совершенствованию менеджмента в России <a href="http://www.vopros.ru">http://www.vopros.ru</a>	07 Июн 2022	Интернет Плюс	Источник исключен. Причина: Маленький процент пересечения.
[81]	АЛГОРИТМ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ ПО ВЫБОРУ СРЕДСТВ ОБРАБОТКИ БОЛЬШИХ МАССИВОВ ЕСТЕСТВЕННО-ЯЗЫКОВЫХ ДАННЫХ <a href="http://elibrary.ru">http://elibrary.ru</a>	12 Фев 2019	eLIBRARY.RU	Источник исключен. Причина: Маленький процент пересечения.
[82]	«Ростелеком» представил в Калуге умный комплекс для контроля дорожной инфраструктуры <a href="http://www.vopros.ru">http://www.vopros.ru</a>	27 Апр 2021	СМИ России и СНГ	Источник исключен. Причина: Маленький процент пересечения.

Виктор Васильев     
 Руководитель научной группы  
 Рук. соп. групп (Александр С. А.)  
 (Великов С. А.)