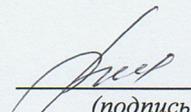


Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
«Комсомольский-на-Амуре государственный университет»

Работа выполнена в СКБ «Электроника и робототехника»

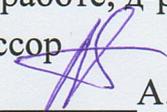
СОГЛАСОВАНО

Начальник отдела ОНиПКРС

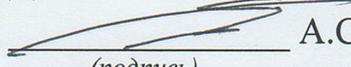
  
(подпись) Е.М. Димитриади  
« 16 » 06 2023 г.

УТВЕРЖДАЮ

Проректор по науке и инноваци-  
онной работе, д-р техн. наук.  
профессор

  
(подпись) А.В. Космынин  
« 16 » 06 2023 г.

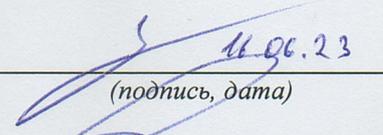
Декан ФЭУ

  
(подпись) А.С. Гудим  
« 16 » 06 2023 г.

«Разработка системы распознавания жестов для управления  
коллаборативным роботом в сложной промышленной среде»

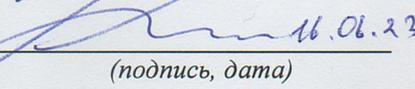
Комплект конструкторской документации

Руководитель СКБ

  
(подпись, дата)

В.В. Солецкий

Руководитель проекта

  
(подпись, дата)

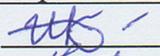
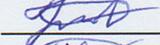
Ю.С. Иванов

Наставник проекта

  
(подпись, дата)

Д.М. Грабарь

### Карточка проекта

Тип проекта	техническое творчество (инициативный)		
Исполнители	Ученик		И.А. Косачов – 16 школа
	Ученик		А.Н. Панькин – 16 школа
	Ученик		И.Е. Жданов – 16 школа
Срок реализации	01.10.2022 – 01.05.2023		

### Использованные материалы и компоненты

Наименование	Количество, шт.
Камера Intel Realsense D455	1
Одноплатный компьютер Jetson Nano под управлением ОС Linux	1
Коллаборативный робот KUKA 7 iiwa R800	1

Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
«Комсомольский-на-Амуре государственный университет»

ЗАДАНИЕ  
на разработку

Название проекта: Разработка системы распознавания жестов для управления коллаборативным роботом в сложной промышленной среде

Назначение: упрощение работы с коллаборативным роботом в сложной промышленной среде за счёт использования системы распознавания жестов

Область использования: промышленное производство, научно-исследовательские лаборатории

Функциональное описание проекта: камера непрерывно передаёт изображение на вычислительное устройство (компьютер), на котором нейросетевая модель производит обработку видеопотока. В случае обнаружения жеста, находящегося в базе данных, управляющее вычислительное устройство посылает управляющий сигнал на контроллер робота. Полученный сигнал обрабатывается, происходит процесс взаимодействия с коллаборативным роботом

Техническое описание устройства: устройство представляет из себя любой вычислительный модуль (ноутбук, одноплатный компьютер, персональный компьютер) подключенный в локальную сеть с промышленным роботом. По локальной сети происходит обмен информацией между контроллером робота и вычислительным модулем. Вычислительный модуль оснащен внешней камерой

Требования: представленное решение не должно нарушать требования ISO/TS 15066:2016 («Роботы и робототехнические устройства — Коллаборативные роботы»), ISO 10218-1 и ISO 10218-2 («Требования безопасности для промышленных роботов»)

План работ:

Наименование работ	Срок
Разработка системы передачи управляющих сигналов на контроллер робота	10.2022
Разработка системы сбора данных для обучения нейросетевой модели	11.2022
Сбор и разметка данных	02.2023
Обучение нейросетевой модели	03.2023
Написание программного комплекса обработки жестов	04.2023
Объединение имеющихся наработок в единый программно-аппаратный комплекс	04.2023

Комментарии:

---

---

---

---

---

Перечень графического материала:

1. Внешний вид изделия;
2. Функциональная модель;
3. Нейросетевая архитектура;
4. Программная реализация.

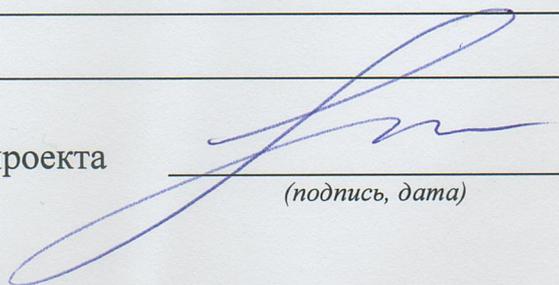
---

---

---

---

Руководитель проекта



---

(подпись, дата)

Ю.С. Иванов

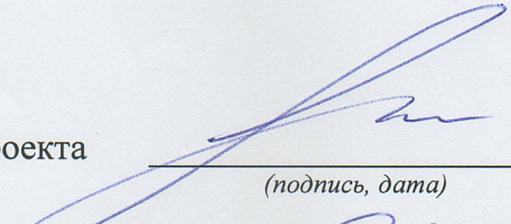
Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
«Комсомольский-на-Амуре государственный университет»

## ПАСПОРТ

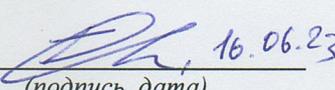
«Разработка системы распознавания жестов для управления  
коллаборативным роботом в сложной промышленной среде»

Руководитель проекта

  
\_\_\_\_\_  
(подпись, дата)

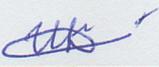
Ю.С. Иванов

Наставник проекта

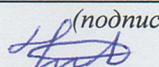
  
\_\_\_\_\_  
(подпись, дата)

Д.М. Грабарь

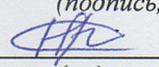
Исполнители проекта

  
\_\_\_\_\_  
(подпись, дата)

И.А. Косачов

  
\_\_\_\_\_  
(подпись, дата)

А.Н. Панькин

  
\_\_\_\_\_  
(подпись, дата)

И.Е. Жданов

Комсомольск-на-Амуре 2023

## Содержание

1	Общие положения .....	7
1.1	Наименование изделия .....	7
1.2	Наименования документов, на основании которых ведется проектирование изделия.....	7
1.3	Перечень организаций, участвующих в разработке изделия .....	7
1.4	Сведения об использованных при проектировании нормативно-технических документах .....	8
2	Назначение и принцип действия .....	9
2.1	Назначение изделия .....	9
2.2	Области использования изделия .....	9
2.3	Принцип действия изделия .....	9
3	Состав изделия и комплектность.....	10
4	Технические характеристики.....	11
4.1	Основные технические характеристики блока одноплатного компьютера Jetson Nano .....	11
4.2	Основные технические характеристики камеры Intel Realsense D455 .....	11
4.3	Основные технические характеристики коллаборативного робота KUKA 7 iiwa R800.....	12
5	Устройство и описание работы изделия .....	13
6	Условия эксплуатации .....	15
6.1	Правила и особенности размещения изделия .....	15
6.2	Меры безопасности.....	15
6.3	Правила хранения и транспортирования.....	16
	ПРИЛОЖЕНИЕ А .....	17
	ПРИЛОЖЕНИЕ Б.....	21

					<b>СКБФЭУ.1.ИП.00000000</b>	Лист
Изм.	Лист.	№ документа	Подп.	Дата.		6

## **1 Общие положения**

Настоящий паспорт является документом, предназначенным для ознакомления с основными техническими характеристиками, устройством, правилами установки и эксплуатации устройства «Разработка системы распознавания жестов для управления коллаборативным роботом в сложной промышленной среде» (далее «изделие»).

Паспорт входит в комплект поставки изделия. Прежде, чем пользоваться изделием, внимательно изучите правила обращения и порядок работы с ним. В связи с постоянной работой по усовершенствованию изделия, повышающей его надежность и улучшающей условия эксплуатации, в конструкцию могут быть внесены изменения, не отраженные в данном издании.

### **1.1 Наименование изделия**

Полное наименование изделия – «Разработка системы распознавания жестов для управления коллаборативным роботом в сложной промышленной среде».

### **1.2 Наименования документов, на основании которых ведется проектирование изделия**

Проектирование «Разработка системы распознавания жестов для управления коллаборативным роботом в сложной промышленной среде» осуществляется на основании требований и положений следующих документов:

- задание на разработку.

### **1.3 Перечень организаций, участвующих в разработке изделия**

Заказчиком проекта «Разработка системы распознавания жестов для управления коллаборативным роботом в сложной промышленной среде» является Федеральное государственное бюджетное образовательное учреждение высшего образования «Комсомольский-на-Амуре

					<b>СКБФЭУ.1.ИП.01000000</b>	<i>Лист</i>
<i>Изм.</i>	<i>Лист.</i>	<i>№ документа</i>	<i>Подп.</i>	<i>Дата.</i>		7

государственный университет» (далее заказчик), находящийся по адресу: 681013, Хабаровский край, г. Комсомольск-на-Амуре, Ленина пр-кт., д. 17.

Исполнителями проекта «Разработка системы распознавания жестов для управления коллаборативным роботом в сложной промышленной среде» являются Конструкторы студенческого конструкторского бюро «Электроника и робототехника» (далее СКБ), школьникам 16 школы, Панькину Александру Николаевичу, Косачову Ильи Александровичу, Жданову Ивану Евгеньевичу

#### **1.4 Сведения об использованных при проектировании нормативно-технических документах**

При проектировании использованы следующие нормативно-технические документы:

ГОСТ 2.001-2013. Единая система конструкторской документации. Общие положения.

ГОСТ 2.102-2013. Единая система конструкторской документации. Виды и комплектность конструкторских документов.

ГОСТ 2.105-95. Единая система конструкторской документации. Общие требования к текстовым документам.

ГОСТ 2.610-2006. Единая система конструкторской документации. Правила выполнения эксплуатационных документов.

ГОСТ 2.004-88. Единая система конструкторской документации. Общие требования к выполнению конструкторских технологических документов на печатающих и графических устройствах вывода ЭВМ.

ГОСТ 2.051-2006. Единая система конструкторской документации. Электронные документы. Общие положения.

ГОСТ 2.052-2006. Единая система конструкторской документации. Электронная модель изделия. Общие положения.

ГОСТ 2.601-2013. Единая система конструкторской документации. Эксплуатационные документы.

					<b>СКБФЭУ.1.ИП.01000000</b>	Лист
Изм.	Лист.	№ документа	Подп.	Дата.		8

## **2 Назначение и принцип действия**

### **2.1 Назначение изделия**

Разработка системы распознавания жестов для управления коллаборативным роботом в сложной промышленной среде необходимо для упрощения взаимодействия с коллаборативным роботом на промышленном производстве или в научно-исследовательских лабораториях, где робот является основным инструментом в достижении поставленных целей и задач при создании продукции.

В состав изделия входят: одноплатный компьютер Jetson Nano под управлением ОС Linux (Ubuntu), камера Intel RealSense D455, коллаборативный робот KUKA 7 iiwa R800, программная реализация.

### **2.2 Области использования изделия**

Изделие может применяться в промышленном производстве (судо-, самолето-, авто-строении), в научно-исследовательских лабораториях.

### **2.3 Принцип действия изделия**

Камера непрерывно передаёт изображение на вычислительное устройство (компьютер), на котором нейросетевая модель производит обработку видеопотока. В случае обнаружения жеста, находящегося в базе данных, управляющее вычислительное устройство посылает управляющий сигнал на контроллер робота. Полученный сигнал обрабатывается, происходит процесс взаимодействия с коллаборативным роботом. На данный момент обрабатывается три типа сигнала: управление захватом; движение робота за рукой оператора; движение робота по заранее нарисованной (с помощью руки) траектории.

					<b>СКБФЭУ.1.ИП.02000000</b>	<i>Лист</i>
<i>Изм.</i>	<i>Лист.</i>	<i>№ документа</i>	<i>Подп.</i>	<i>Дата.</i>		9

### 3 Состав изделия и комплектность

В комплект поставки входит:

- Камера Intel Realsense D455;
- Одноплатный компьютер Jetson Nano под управлением ОС Linux (Ubuntu);
- Коллаборативный робот KUKA 7 iiwa R800;
- Программная реализация;
- Паспорт.

					<b>СКБФЭУ.1.ИП.03000000</b>	<i>Лист</i>
<i>Изм.</i>	<i>Лист.</i>	<i>№ документа</i>	<i>Подп.</i>	<i>Дата.</i>		10



### 4.3 Основные технические характеристики коллаборативного робота KUKA 7 iiwa R800

Основные технические характеристики коллаборативного робота KUKA 7 iiwa R800 приведены в таблице 3.

Таблица 3 – Технические характеристики коллаборативного робота KUKA 7 iiwa R800

Наименование параметра	Значение
Тип кинематики	Шарнирная рука
Макс. рабочий радиус	800 мм.
Номинальная грузоподъемность	7 кг.
Стабильность повторяемости позиций (ISO 9283)	± 0,1 мм.
Количество осей	7
Монтажное положение	Пол, потолок, стена
Вес	23.9 кг.
Температура окружающей среды при эксплуатации	5°C - 45°C (278 К - 318 К)
Класс защиты (IEC 60529)	IP54
Система управления	KUKA Sunrise Cabinet
Диапазон перемещения A1	±170°
Диапазон перемещения A2	±120°
Диапазон перемещения A3	±170°
Диапазон перемещения A4	±120°
Диапазон перемещения A5	±170°
Диапазон перемещения A6	±120°
Диапазон перемещения A7	±175°
Скорость при номинальной грузоподъемности A1	98°/s
Скорость при номинальной грузоподъемности A2	98°/s
Скорость при номинальной грузоподъемности A3	100°/s
Скорость при номинальной грузоподъемности A4	130°/s
Скорость при номинальной грузоподъемности A5	140°/s
Скорость при номинальной грузоподъемности A6	180°/s
Скорость при номинальной грузоподъемности A7	180°/s

## 5 Устройство и описание работы изделия

Процесс распознавания ключевых точек на ладони человека с помощью нейронной сети начинается с получения видеопотока с камеры Intel Realsense D455. Затем, используя алгоритмы компьютерного зрения, изображение разбивается на отдельные кадры.

Далее, каждый кадр подается на вход нейронной сети, которая обучена распознавать ключевые точки на ладони человека. Нейронная сеть использует алгоритмы глубокого обучения, такие как сверточные нейронные сети, для извлечения признаков изображения и определения положения ключевых точек на ладони человека.

Ключевые точки на ладони человека могут включать в себя такие элементы, как пальцы, запястье и т.д. Каждая ключевая точка имеет свой уникальный идентификатор, который нейронная сеть определяет в соответствии с обучающими данными.

Процесс передачи информации между контроллером робота и вычислительным блоком на базе Jetson Nano:

1. Команда на перемещение коллаборативного робота формируется на вычислительном блоке на базе Jetson Nano. Это сделано с помощью разработанного программного обеспечения, которое позволяет задавать координаты перемещения, скорость и другие параметры движения.

2. Сформированная команда отправляется на контроллер робота. Для этого используется специальный протокол связи, который позволяет передавать данные между устройствами.

3. Контроллер робота принимает команду и обрабатывает ее.

4. После обработки команды контроллер робота отправляет сигналы на двигатели и другие устройства, которые отвечают за перемещение робота. Это позволяет роботу выполнить заданную команду на перемещение.

5. В процессе выполнения команды контроллер робота отправляет обратную связь на вычислительный блок на базе Jetson Nano. Это

					<b>СКБФЭУ.1.ИП.05000000</b>	<i>Лист</i>
<i>Изм.</i>	<i>Лист.</i>	<i>№ документа</i>	<i>Подп.</i>	<i>Дата.</i>		13

информация о текущем положении робота, скорости движения и другие параметры.

6. Вычислительный блок на базе Jetson Nano может использует эту информацию для корректировки команды на перемещение робота или для выполнения других задач.

В приложении А представлены основные результаты проекта. В приложении Б отображена программная реализация проекта.

					<b>СКБФЭУ.1.ИП.05000000</b>	<i>Лист</i>
<i>Изм.</i>	<i>Лист.</i>	<i>№ документа</i>	<i>Подп.</i>	<i>Дата.</i>		14

## **6 Условия эксплуатации**

Изделие выпускается в климатическом исполнении УХЛ 4.2 по ГОСТ 15150-69 и предназначен для использования в стационарных условиях в закрытых помещениях при соответствующих климатических условиях:

- интервал температур от +10 до +35 °С;
- относительная влажность воздуха до 80 % при температуре +25 °С;
- высота над уровнем моря не более 2000 м;
- атмосферное давление от 86,6 до 106 кПа (от 650 до 800 мм рт. ст.).

В помещении, где используется изделие не должно возникать условий для конденсации влаги (выпадения росы). Изделие является электронным прибором, требующим бережного обращения.

Для обеспечения безотказной работы, сохранения точности и его сбережения необходимо соблюдать следующие правила:

- изучить паспорт, прежде чем приступить к работе с изделием;
- предохранять изделие от ударов и повреждений;
- не допускать самостоятельную разборку изделия.

### **6.1 Правила и особенности размещения изделия**

Изделие должно быть расположено на расстоянии не менее 1 м от нагревательных приборов.

**ВНИМАНИЕ!** При эксплуатации изделия запрещается проводить самостоятельно какие-то либо работы по извлечению и установке внутренних компонентов изделия.

### **6.2 Меры безопасности**

Необходимо соблюдать требования техники безопасности и следующие меры предосторожности:

- после транспортировки в холодное время года изделие необходимо выдержать при комнатной температуре не менее двух часов;
- внутренние осмотры и ремонт изделия должны производиться только квалифицированными специалистами;

					<b>СКБФЭУ.1.ИП.06000000</b>	<i>Лист</i>
<i>Изм.</i>	<i>Лист.</i>	<i>№ документа</i>	<i>Подп.</i>	<i>Дата.</i>		15

- не устанавливайте изделие на неустойчивой подставке, стойке или ненадежном кронштейне.

### **6.3 Правила хранения и транспортирования**

Транспортирование изделия в упакованном виде может производиться железнодорожным, автомобильным (в закрытых транспортных средствах), воздушным, речным и морским видами транспорта в соответствии с правилами перевозок грузов, действующих на транспорт данного вида. Условия транспортирования изделия по части воздействия климатических факторов должны соответствовать группе 5 по ГОСТ 15150.

После транспортирования изделие должно быть выдержано не менее 2 часов в транспортной таре при температуре  $20 \pm 5$  °С и относительной влажности воздуха не более 80 %.

Распакованное изделие должно храниться в отапливаемом и вентилируемом чистом помещении при температуре от +5 до +40 °С и относительной влажности воздуха не более 60 %. При температуре ниже 25 °С допускается увеличение относительной влажности до 80 %. Воздух в помещении не должен содержать примесей, вызывающих коррозию металлов, налеты на поверхностях оптических деталей.

					<b>СКБФЭУ.1.ИП.06000000</b>	<i>Лист</i>
<i>Изм.</i>	<i>Лист.</i>	<i>№ документа</i>	<i>Подп.</i>	<i>Дата.</i>		16

# ПРИЛОЖЕНИЕ А

(обязательное)

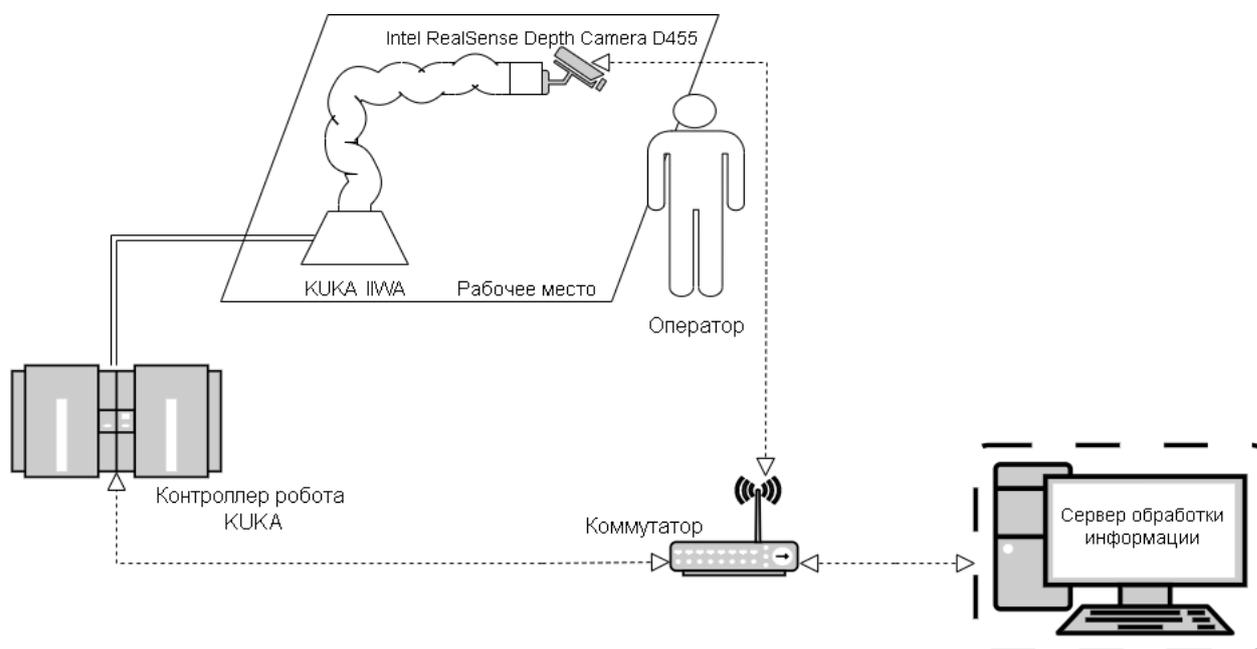


Рисунок А.1 – Функциональная схема проекта



Рисунок А.2 – Процесс разметки данных

					<b>СКБФЭУ.1.ИП.00000000</b>	Лист
Изм.	Лист.	№ документа	Подп.	Дата.		17



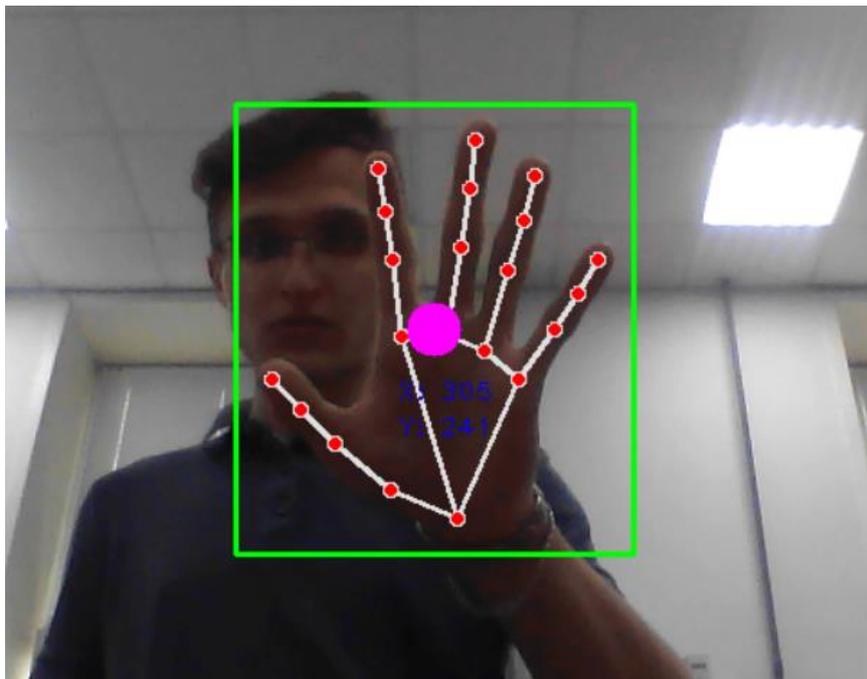


Рисунок А.6 – Перемещение коллаборативного робота за рукой оператора

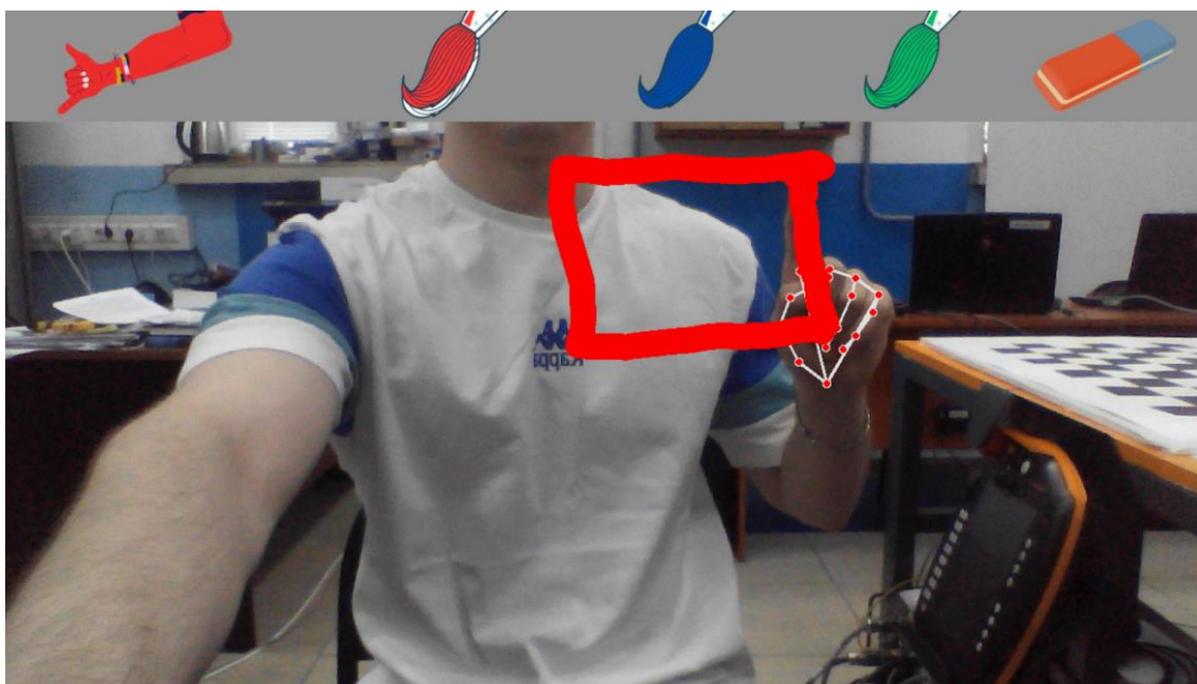


Рисунок А.7 – Перемещение коллаборативного робота по заранее нарисованной траектории

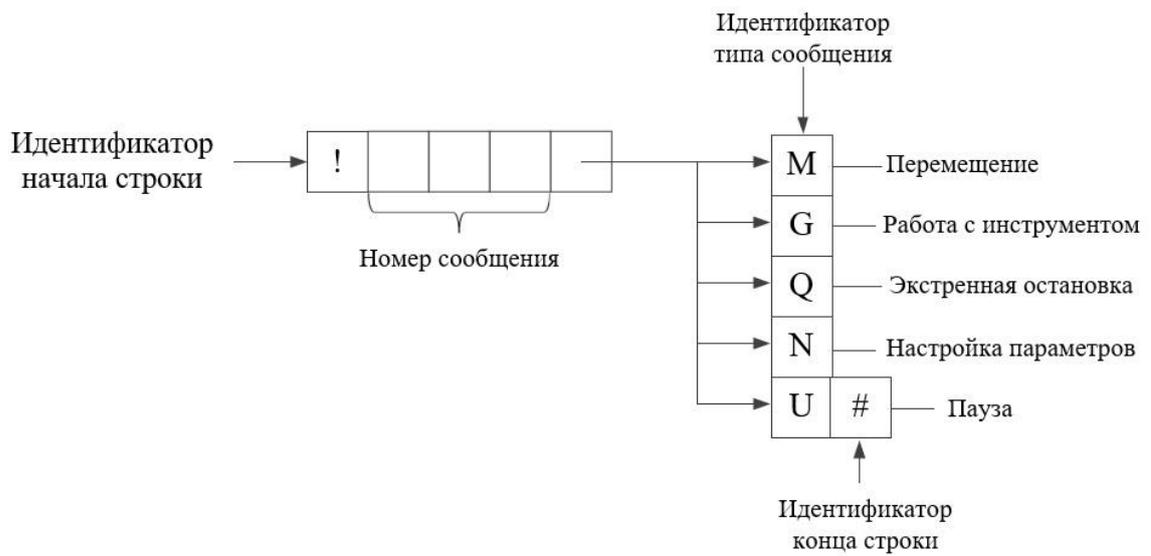


Рисунок А.8 – Формат отправки сигнала с вычислительного устройства на контроллер робота



```

66.         acceleration = Float.parseFloat(checker.replaceAll("a", ""));
67.     }
68.     else if (checker.startsWith("r"))
69.         rotation = Float.parseFloat(checker
70.             .replaceAll("r", ""));
71.     else if (checker.startsWith("V"))
72.         velocity = Float.parseFloat(checker
73.             .replaceAll("V", ""));
74.     else if (checker.startsWith("s"))
75.         status = Short.parseShort(checker
76.             .replaceAll("s", ""));
77.     else if (checker.startsWith("@"))
78.         smooth = Short.parseShort(checker
79.             .replaceAll("@", ""));
80.     else if (checker.startsWith("O"))
81.         orientation = Short.parseShort(checker
82.             .replaceAll("O", ""));
83.     else if (checker.startsWith("T"))
84.         tool = Short.parseShort(checker
85.             .replaceAll("T", ""));
86.     else if (checker.startsWith("B"))
87.         base = Short.parseShort(checker
88.             .replaceAll("B", ""));
89.     else
90.         cords.add(Float.parseFloat(checker
91.             .replaceAll("\\d{1}+", "")
92.             .replaceAll("\\d{1}[\\-]", "-")
93.             .replaceAll(";", "")));
94.     });
95. }
96. }
97.
98. HashMap<String, Object> move_setting = new HashMap<String, Object>();
99.
100. move_setting.put("move_mode", move_mode);
101. move_setting.put("status", status);
102. move_setting.put("angle_rotate", angle_rotate);
103. move_setting.put("rotation", rotation);
104. move_setting.put("velocity", velocity);
105. move_setting.put("acceleration", acceleration);
106. move_setting.put("smooth", smooth);
107. move_setting.put("orientation", orientation);
108. move_setting.put("tool", tool);
109. move_setting.put("base", base);
110. move_setting.put("cords", cords);
111.
112. return move_setting;
113. }
114.
115. private HashMap<String, Object> settingIdsParse(String request) {
116.     HashMap<String, Object> settings = new HashMap<String, Object>();
117.
118.     float velocity = 0f,
119.     acceleration = 0f,
120.     smooth = 0f,
121.     orientation = 0f,
122.     tool = 0f,
123.     base = 0f,
124.     override = 0f,
125.     end_connection = 0f;
126.
127.     if (request.contains("V"))
128.         velocity = Float.parseFloat(
129.             request.substring(request.indexOf('v')+1, request.indexOf('v')+5)
130.         );
131.     if (request.contains("a"))
132.         acceleration = Float.parseFloat(
133.             request.substring(request.indexOf('a')+1, request.indexOf('a')+4)
134.         );
135.     if (request.contains("@"))
136.         smooth = Float.parseFloat(

```

						<i>Лист</i>
						22
<i>Изм.</i>	<i>Лист.</i>	<i>№ документа</i>	<i>Подп.</i>	<i>Дата.</i>	<b>СКБФЭУ.1.ИП.00000000</b>	

```

137.         request.substring(request.indexOf('@')+1, request.indexOf('@')+4)
138.     );
139.     if (request.contains("O"))
140.         orientation = Float.parseFloat(
141.             request.substring(request.indexOf('o')+1, request.indexOf('o')+2)
142.         );
143.     if (request.contains("T"))
144.         tool = Float.parseFloat(
145.             request.substring(request.indexOf('t')+1, request.indexOf('t')+3)
146.         );
147.     if (request.contains("B"))
148.         base = Float.parseFloat(
149.             request.substring(request.indexOf('b')+1, request.indexOf('b')+3)
150.         );
151.     if (request.contains("W"))
152.         override = Float.parseFloat(
153.             request.substring(request.indexOf('w')+1, request.indexOf('w')+5)
154.         );
155.     if (request.contains("D"))
156.         end_connection = 1f;
157.
158.     settings.put("velocity", velocity);
159.     settings.put("acceleration", acceleration);
160.     settings.put("smooth", smooth);
161.     settings.put("orientation", orientation);
162.     settings.put("tool", tool);
163.     settings.put("base", base);
164.     settings.put("override", override);
165.     settings.put("end_connection", end_connection);
166.
167.     return settings;
168. }
169.
170. private HashMap<String, Object> toolsParse(String request) {
171.     HashMap<String, Object> tool = new HashMap<String, Object>();
172.
173.     int tool_number = Integer.parseInt(
174.         request.substring(request.indexOf('G')+1, request.indexOf('G')+3)
175.     );
176.     int mode_tool = 0;
177.
178.     if (request.contains(";"))
179.         mode_tool = Integer.parseInt(
180.             request.substring(request.indexOf(';')+1, request.indexOf(';') + 3)
181.         );
182.
183.     tool.put("tool", tool_number);
184.     tool.put("mode", mode_tool);
185.
186.     return tool;
187. }
188.
189. public HashMap<String, Object> parseString(String message, String regex) {
190.     HashMap<String, Object> result = new HashMap<String, Object>();
191.     result.put("message", "Error");
192.
193.     Pattern pattern = Pattern.compile(regex, Pattern.MULTILINE);
194.     String[] strings = pattern.split(message, 2);
195.
196.     if (strings.length == 2) {
197.
198.         String request = strings[1];
199.
200.         if ( !request.endsWith("#") ) {
201.             return result;
202.         }
203.
204.         switch (request.charAt(0)) {
205.             case messageIds.REQUEST:
206.                 result = requestIdsParse(request);
207.                 result.put("message", messageIds.REQUEST);

```

						<i>Лист</i>
					<b>СКБФЭУ.1.ИП.00000000</b>	23
<i>Изм.</i>	<i>Лист.</i>	<i>№ документа</i>	<i>Подп.</i>	<i>Дата.</i>		

```

208.         break;
209.     case messageIds.MOVING:
210.         // "P CμP BμPŸP,P BμP C♦P BμPŸB°P BμP P...P C°P Bμ"
211.         result = moveIdsParse(request);
212.         result.put("message", messageIds.MOVING);
213.         break;
214.     case messageIds.TOOLS:
215.         // P nīSP P...PŸPŸPŸB°PŸP,PŸC°P C♦P BμP P...PŸB°B°
216.         result = toolsParse(request);
217.         result.put("message", messageIds.TOOLS);
218.         break;
219.     case messageIds.PAUSE:
220.         // P CμP B°PŸC°P B°P B°
221.         result.put("message", messageIds.PAUSE);
222.         break;
223.     case messageIds.QUEUE_COMMANDS:
224.         // P PŸP B±PŸP,P C°PŸPŸP P C°PŸB°PŸP BμPŸP,P BμP T°P C°
P C°P C°P C♦P C♦P B°P P...P T°
225.         result.put("message", messageIds.QUEUE_COMMANDS);
226.         break;
227.     case messageIds.SETTING_PARAMETERS:
228.         // P CμP B°PŸPŸPŸB°PŸP,P C°P B,,-P C°P C° PŸPŸP C°PŸPŸPŸB°P BμP C♦PŸB°B°
229.         result = settingIdsParse(request);
230.         result.put("message", messageIds.SETTING_PARAMETERS);
231.         break;
232.     default:
233.         // P C°PŸP,P C°PŸB°P C°PŸB°PŸP BμPŸPŸP C°P B°PŸPŸP
P C°PŸPŸPŸB°P B°P P...P C°P P†P C°P B°
234.         result.put("message", messageIds.CRITICAL_PAUSE);
235.         break;
236.     }
237.
238. }
239.
240.     return result;
241. }
242.
243. }
244.

```

## Листинг Б.2 – Интерфейс сообщений

```

1. package SocketRobot;
2.
3. public interface messageIds {
4.     char REQUEST = 'Z';
5.     char MOVING = 'M';
6.     char TOOLS = 'G';
7.     char CRITICAL_PAUSE = 'Q';
8.     char QUEUE_COMMANDS = 'R';
9.     char SETTING_PARAMETERS = 'N';
10.    char PAUSE = 'U';
11. }
12.

```

## Листинг Б.3 – Интерфейс перемещений

```

1. package SocketRobot;
2.
3. public interface moveIds {
4.     char PTP = 'P';
5.     char PTP_ANGLE = 'A';
6.     char LIN = 'L';
7.     char CIRC = 'C';
8.     char SPLINE = 'S';
9.     char HOME = 'H';
10. }
11.

```





```

6.
7. import main as mn
8. import config as cfg
9.
10.
11. clients = []
12.
13. async def robot_func():
14.     text = "None\n"
15.
16.     try:
17.         state_queue = mn.queue_process.get(block=False)
18.
19.         if state_queue["mode"] == "gripper":
20.             text = f"!001G01;0{int(mn.state['gripper'])}#\n"
21.         elif state_queue["mode"] == 'move':
22.             cords_text =
f"{mn.state['cords'][0]};2{mn.state['cords'][1]};3{mn.state['cords'][2]};"
23.             angle_text =
f"4{mn.state['angle'][0]};5{mn.state['angle'][1]};6{mn.state['angle'][2]};"
24.             E1_text = f"{mn.state['E1']}"
25.             text = f"!001M0P{cords_text}{angle_text}{E1_text}s6r044V0.10a0.10#\n"
26.         elif state_queue["mode"] == 'simple':
27.             text = "!001U#\n"
28.
29.         mn.queue_process.task_done()
30.
31.     except queue.Empty:
32.         pass
33.
34.     return text
35.
36.
37. def camera_func():
38.     print('it`s camera func')
39.
40.
41. async def handle_client(client):
42.     loop = asyncio.get_event_loop()
43.     request = None
44.     response = "None"
45.     while request != 'exit':
46.
47.         request = (await loop.sock_recv(client, 1024)).decode('utf8')
48.         print("Request = ", request)
49.
50.         if client.getpeername()[0] == cfg.ID_DICT["kuka"]:
51.             response = await robot_func()
52.             print(response)
53.
54.         if request == "exit":
55.             await loop.sock_sendall(client, "!000M0HV0.50a0.50#".encode('utf8'))
56.             break
57.
58.         if response != "None" and request == "ready":
59.             print("Direct message")
60.             await loop.sock_sendall(client, response.encode('utf8'))
61.
62.     client.close()
63.
64.
65. async def writeClient(client, message):
66.     loop = asyncio.get_event_loop()
67.     await loop.sock_sendall(client, message.encode('utf8'))
68.
69.
70. async def run_server():
71.     server = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
72.     server.bind((cfg.HOST, cfg.PORT))
73.     server.listen(5)
74.     server.setblocking(0)

```

					<b>СКБФЭУ.1.ИП.00000000</b>	Лист
Изм.	Лист.	№ документа	Подп.	Дата.		27

```

75.     loop = asyncio.get_event_loop()
76.     while True:
77.         client, _ = await loop.sock_accept(server)
78.         if not client in clients:
79.             clients.append(client)
80.             loop.create_task(handle_client(client))
81.
82.
83. if __name__ == "__main__":
84.     opnecv_project = threading.Thread(target=mn.application, args=[], daemon=True)
85.     opnecv_project.start()
86.     asyncio.run(run_server())
87.

```

## Листинг Б.8 – Пример взаимодействия с коллаборативным роботом

```

1. package SocketRobot;
2.
3. import java.io.*;
4. import java.net.Socket;
5. import java.nio.charset.StandardCharsets;
6. import java.util.HashMap;
7.
8.
9. public class Main extends Thread {
10.
11.     private Socket socket = null;
12.     private BufferedReader reader = null;
13.     private BufferedWriter writer = null;
14.     private boolean stop = false;
15.     private String msg = "None";
16.
17.     public Main(String address, int port) {
18.         try {
19.             System.out.println("Connected to " + address + ":" + port);
20.             socket = new Socket(address, port);
21.             reader = new BufferedReader(new InputStreamReader(socket.getInputStream()));
22.             writer = new BufferedWriter(new OutputStreamWriter(socket.getOutputStream(),
23. //                 Charset.forName("UTF-8"),
24.                 StandardCharsets.UTF_8));
25.
26.         } catch (IOException e) {
27.             closeSocket();
28.             closeBuffer();
29.             e.printStackTrace();
30.         }
31.     }
32.
33.     private void closeBuffer() {
34.         try {
35.             System.out.println("Close buffer");
36.             reader.close();
37.             writer.close();
38.         } catch (IOException ex) {
39.             throw new RuntimeException(ex);
40.         }
41.     }
42.
43.     private void closeSocket() {
44.         try {
45.             socket.close();
46.         } catch (IOException e) {
47.             e.printStackTrace();
48.         }
49.     }
50.
51.     @Override
52.     public void run() {
53.

```

```

54.     while (!stop) {
55.         try {
56.             CommandParser parser = new CommandParser();
57.
58.             HashMap<String, Object> message = parser.parseString(msg, parser.REGEX_START);
59.             System.out.println("I get: " + msg);
60.
61.             wait(3000);
62.
63.             // if (!msg.equals("None")) {
64.                 writeToUser("ready");
65.             // }
66.
67.             msg = readToUser();
68.             if (msg.equals("exit")) {
69.                 stopSocket();
70.             }
71.
72.         } catch (IOException e) {
73.             closeSocket();
74.             closeBuffer();
75.             System.out.println(e);
76.             break;
77.         }
78.     }
79.     closeSocket();
80.     closeBuffer();
81. }
82.
83. public static void wait(int ms) {
84.     try {Thread.sleep(ms);}
85.     catch (InterruptedException ex) {
86.         Thread.currentThread().interrupt();
87.     }
88. }
89.
90. public void writeToUser(String message) throws IOException {
91.     writer.write(message);
92.     writer.flush();
93. }
94.
95. public String readToUser() throws IOException {
96.     return reader.readLine();
97. }
98.
99. public void stopSocket() throws IOException {
100.    System.out.println("Закрытия сокета");
101.    writeToUser("exit");
102.    stop = true;
103. }
104.
105. public static void main(String[] args) throws IOException {
106.    String address = "192.168.2.188";
107.    int port = 5000;
108.
109.    Main test = new Main(address, port);
110.
111.    test.writeToUser("Start process");
112.    test.start();
113. }
114. }
115. }
116.

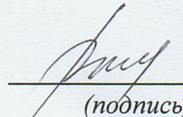
```

Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
«Комсомольский-на-Амуре государственный университет»

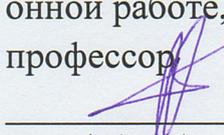
СОГЛАСОВАНО

Начальник отдела ОНиПКРС

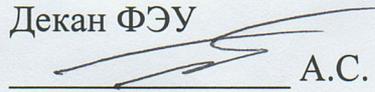
  
(подпись) Е.М. Димитриади  
« 16 » 06 20 23 г.

УТВЕРЖДАЮ

Проректор по науке и инноваци-  
онной работе, д-р техн. наук.  
профессор

  
(подпись) А.В. Космынин  
« 16 » 06 20 23 г.

Декан ФЭУ

  
(подпись) А.С. Гудим  
« 16 » 06 20 23 г.

АКТ

о приемке в эксплуатацию проекта  
«Разработка системы распознавания жестов для управления  
коллаборативным роботом в сложной промышленной среде»

г. Комсомольск-на-Амуре

« 16 » 06 20 23 г.

Комиссия в составе представителей:

со стороны заказчика

- В.В. Солецкий – руководитель СКБ,
- А.С. Гудим – декан ФЭУ

со стороны исполнителя

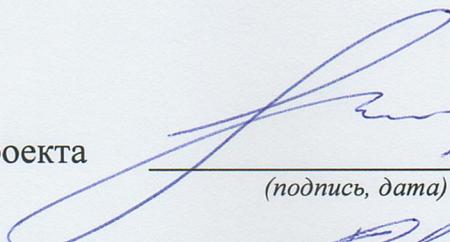
- Ю.С. Иванов – руководителя проекта,
- Д.М. Грабарь – наставник проекта,
- И.А. Косачов – ученик 16 школы,
- А.Н. Панькин – ученик 16 школы,
- И.Е. Жданов – ученик 16 школы.

составила акт о нижеследующем:

«Исполнитель» передает проект «Разработка системы распознавания жестов для управления коллаборативным роботом в сложной промышленной среде», в составе:

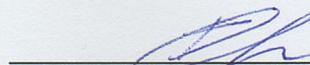
1. Камера Intel Realsense D455;
2. Одноплатный компьютер Jetson Nano под управлением ОС Linux (Ubuntu);
3. Коллаборативный робот KUKA 7 iiwa R800;
4. Программная реализация

Руководитель проекта

  
16.06.23  
(подпись, дата)

Ю.С. Иванов

Наставник проекта

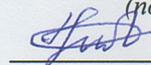
  
16.06.23  
(подпись, дата)

Д.М. Грабарь

Исполнители проекта

  
16.06.23  
(подпись, дата)

И.А. Косачов

  
16.06.23  
(подпись, дата)

А.Н. Панькин

  
16.06.23  
(подпись, дата)

И.Е. Жданов