

УДК 004.896, 62-519

## УЛУЧШЕНИЕ ПОЛЬЗОВАТЕЛЬСКОГО ГРАФИЧЕСКОГО ИНТЕРФЕЙСА ДЛЯ УПРАВЛЕНИЯ ГУСЕНИЧНЫМ РОБОТОМ СЕРВОСИЛА ИНЖЕНЕР

А. С. Доброквашина (*dobrokvashina@it.kfu.ru*)

Р. Н. Сафин (*safin.ramil@it.kfu.ru*)

Р. О. Лавренов (*lavrenov@it.kfu.ru*)

Лаборатория Интеллектуальных Робототехнических Систем,  
Институт информационных технологий и интеллектуальных  
систем, Казанский федеральный университет, Казань

Я. Бай (*yangbai@fc.ritsumei.ac.jp*)

Кафедра информационных наук и инженерии, Колледж  
информационных наук и инженерии, Университет  
Рицумейкан, Кусацу, Япония

**Аннотация.** Роботы широко используются во многих областях современной жизни, и дистанционное управление играет не последнюю роль в решении критически важных задач. Графический пользовательский интерфейс (GUI) является одним из самых популярных способов для ручного управления и навигации. В этой статье мы представляем улучшенный графический интерфейс для российского гусеничного робота Сервосила Инженер. Он предоставляет оператору изображения с четырех камер робота, представление текущей конфигурации с помощью 3D модели и возможность управления каждым сочленением робота. В дополнение к незначительным корректировкам и расширениям предыдущей версии графического интерфейса, особое внимание уделяется моделированию и управлению захватом робота.<sup>1</sup>

**Ключевые слова:** гусеничный робот, моделирование, Сервосила Инженер, графический интерфейс, 3D, видео, Qt, GUI

---

<sup>1</sup> Исследование выполнено при финансовой поддержке Российского Фонда Фундаментальных Исследований (проект № 19-58-70002). Четвертый автор выражает признательность за поддержку Японскому агентству по науке и технологиям, Программа стратегических международных совместных исследований JST, проект № 18065977.

## Введение

Сегодня роботов можно встретить повсеместно. Они умеют водить автомобили [Paolillo et al., 2018], убираются в наших домах, работают на производствах, помогают хирургам выполнять сложные операции [Isogaki et al., 2011], используя более простые и безопасные методы. Роботы зачастую используются во враждебных для человека условиях, таких как, например, опасные производственные объекты [Martines et al., 2015] или радиоактивные зоны. Точность и производительность роботов и манипуляторов представляет особую ценность. Ведь в отличие от человека, машина может работать круглые сутки семь дней в неделю, не теряя при этом в качестве и скорости.

Для каждой цели существует свой вид роботов, заточенный под конкретные спецификации [Pecka et al., 2017]. А в различных сценариях работы могут быть представлены разные уровни автоматизации [Shimchik et al., 2016]. Например, роботы-пылесосы показывают один из самых высоких уровней автономности, с другой же стороны некоторые крайне важные задачи на сегодняшний день не доверяют роботам. А потому телеоперация остается достаточно востребованной, особенно в таких случаях, как проведение хирургических операций или помощь в поисково-спасательных операциях [Wang et al., 2003]. Телеоперация предполагает стабильную связь с роботом, а также наличие удобного способа взаимодействия с ним [Suzuki et al., 2014]. Это означает простой доступ к данным с камер или сенсоров, а также понятную систему управления самим роботом.

Так, одними из самых используемых решений в вопросе взаимодействия с роботом остаются компьютерные и мобильные приложения с графическим пользовательским интерфейсом (GUI). Данное решение представляется одним из самых простых и универсальных.

Графические интерфейсы используются для различных задач. Одни дают лишь общую информацию о нынешней конфигурации робота, другие, обрабатывая данные с датчиков, могут предупредить оператора о возможном падении робота. Третьи не получают никакой информации с робота, а лишь дают возможность осуществлять движение сочленений. А есть и решения, совмещающие в себе все вышеперечисленное - благодаря ним оператор может не только двигать робота и знать его нынешнюю конфигурацию, но также видеть глазами робота: получать данные с камер и сенсоров в режиме реального времени.

В данной статье речь пойдет про улучшение графического пользовательского интерфейса для российского гусеничного робота Сервосила Инженер [Mavrin et al., 2018].



Рис. 2. Гусеничный робот Сервосила Инженер.

## 1. Гусеничный робот Сервосила Инженер

### 1.1 Общая информация

Мобильный робот гусеничного типа Инженер (рис. 2) был произведен российской компанией Сервосила [Servosila]. Он был разработан для эксплуатации в опасных для человека условиях. Благодаря изолированному и водонепроницаемому корпусу, данный робот может использоваться при различных погодных условиях, а также в экстремальных обстоятельствах, таких как наводнение, пожар или в условиях радиационного загрязнения. Благодаря встроенным сенсорам и моторам Инженер может использоваться как в автономном режиме, так и в режиме телеоперации. На борту Инженера представлены 4 камеры (стереопара, камера с зумом и одна камера заднего вида), инерциальный датчик и фонарик. Также робот оснащен лазерным дальномером [Larson et al., 2011] установленным на подвижной подставке. Это дает роботу возможность измерять не только положительные препятствия, но и отрицательные.

Модульная конструкция робота Сервосила Инженер делает из него замечательную образовательную платформу. С использованием установленных камер робот может быть использован при работе с задачами компьютерного зрения. Манипулятор с захватом дает возможность взаимодействовать с окружающей средой, а также на практике решать задачи прямой и обратной кинематики. А гусеницы и флипперы обеспечивают роботу хорошую проходимость на неровной местности [Magid et al., 2010].

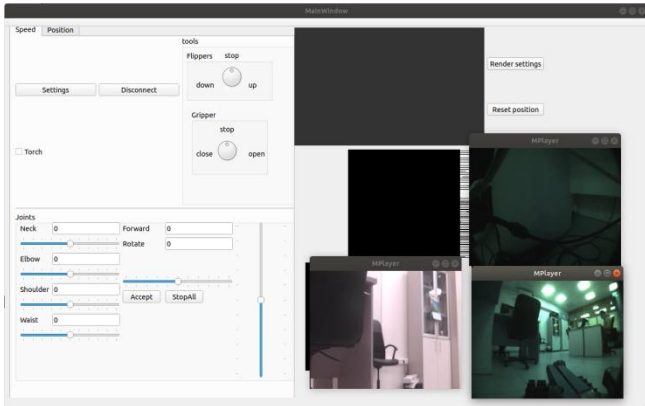


Рис. 3. Исходная версия графического интерфейса (GUI).

## 1.2. Предыдущие версии графического интерфейса

Как упоминалось ранее, эта статья продолжает тему создания графического пользовательского интерфейса для работы с роботом Сервосила Инженер [Mavrin et al., 2018]. На начальном этапе графический интерфейс уже имел возможность управления сочленениями робота, а также представление конфигурации робота с помощью 3D модели (рис. 3).

В этой статье разработка была продолжена, были улучшены некоторые элементы графического интерфейса, в частности дополнена и улучшена подвижность 3D модели, а также доработаны виджеты с изображениями видео, полученных с камер робота.

## 2. Улучшения графического интерфейса

### 2.1. Доработка трехмерной модели

Первая версия 3D модели, представленной в графическом интерфейсе для репрезентации конфигурации робота имеет некоторые отличия относительно реального робота. Этот недостаток может создать существенную практическую проблему при телеоперации без визуального контроля робота, а именно неверное определение рабочего пространства манипулятора или же несоответствие габаритного коридора. По этой причине изменение 3D модели стало одной из первых задач в процессе улучшения графического интерфейса.

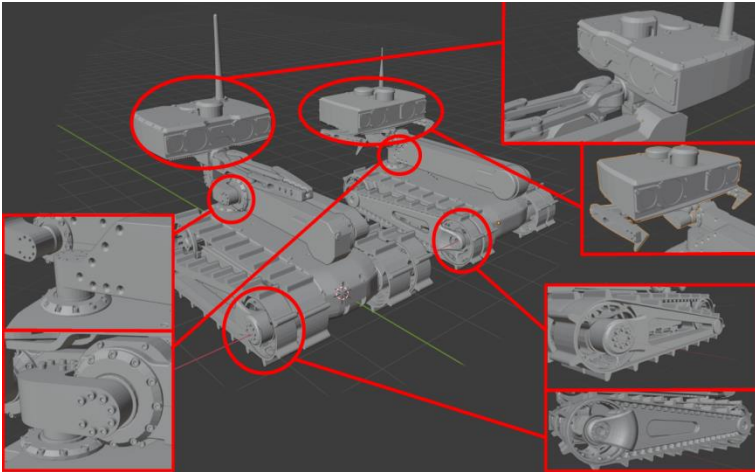


Рис. 4. Сравнение моделей для визуального отображения (слева - новая модель, справа - старая).

Новая 3D модель была создана с помощью элементов, использованных при создании симуляционной модели робота Сервосила Инженер в симуляторе Gazebo [Dobrokvashina et al., 2020]. Сравнение начальной и конечной версий модели представлены на рис. 4.

Другая проблема связана со способом представления модели в графическом интерфейсе. Первое представление трехмерной модели было создано в Qt [Lazar et al., 2016] (Ubuntu 16) с помощью самостоятельно написанного виджета с использованием OpenGL. Однако после обновления Ubuntu до 18 версии была обнаружена критическая проблема с отображением OpenGL. Как видно на рис. 3 3D модель не отображается. В качестве решения пришлось заново воссоздать окно с трехмерной сценой с помощью встроенных модулей Qt3D [Qt 3D Overview], которые предоставляют простое решение не только для добавления модели на сцену, но и дальнейшее ее движение [Harmer et al., 2016].

## 2.2. Движение захвата

Еще одной проблемой трехмерного представления стал захват робота. В старой версии модели захват был соединен с головой робота в единый элемент. Это лишало его возможности изменять положение в соответствии с реальным роботом.



Рис. 5. Схема движения захвата.

После разложения модели на составляющие было получено 7 частей: непосредственно сама голова робота, а также 6 элементов захвата (по три для каждой из двух сторон). Чтобы симитировать движение захвата на модели аналогично реальному при изменении положения необходимо двигать все шесть элементов захвата синхронно, а именно - для открытия или закрытия захвата на некоторый угол  $\beta$  необходимо повернуть каждый из элементов согласно схеме, представленной на рис. 5.

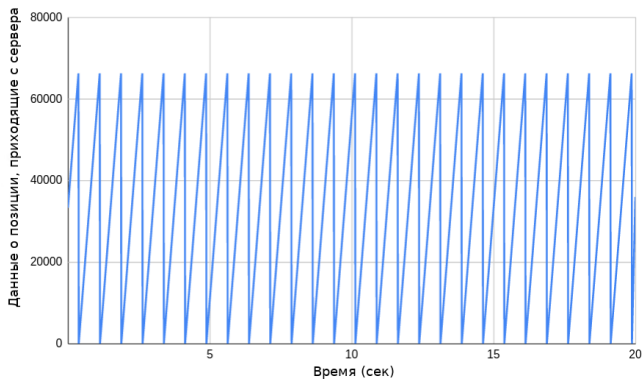


Рис. 6. График зависимости данных о позиции от времени.

Следующим этапом в работе над захватом стала синхронизация модели с реальным роботом. Для синхронизации прочих элементов робота использовались пакеты телеметрии, регулярно поступающие от робота. Они содержат информацию о сервоприводах в целом, а также о каждом из них в частности (Таблица 1). При первых запусках для синхронизации были использованы именно эти значения относительно захвата. Однако в ходе

экспериментов было выяснено, что сервер параметров робота выдает невалидные значения положения захвата робота. Причина - неверно выставленные граничные положения. Корреляция значения положения на сервере от времени движения из крайне открытого в крайне закрытое положение показана на рис. 6.

Таблица I. Структура пакета телеметрии.

Поле	Тип данных	Размер
Frame type ID	unit8	1 байт
Tick number	unit64	8 байт
Number of motors	unit8	1 байт
Motor data #0	struct	24 байта
...	...	...
Motor data #9	struct	24 байта
-	-	25 байт

Одним из возможных способов решения возникнувшей проблемы стал математический подсчет положения захвата на стороне оператора. Если значение положения захвата известно на каждой итерации, значит, начиная с некоторого нового нулевого значения, каждое следующее можно рассчитать, используя разницу между приходящими от сервера значениями. В случае же пересечения границы сервера, высчитывается разница предыдущего значения до пересеченной границы и добавляется разница от противоположной границы до нового значения.

На практике реализация данного метода оказалась невозможной. Частота отправки пакетов телеметрии оказалась недостаточно высокой, что привело к пропуску целых интервалов и, как следствие, потере синхронизации захвата модели с реальным роботом.

Другим возможным решением данной проблемы является использование временных меток. В данном случае происходит отказ от

данных, поступающих с сервера. Для расчета положения захвата используется время открытия\закрытия. Так, при запуске робота, захват автоматически движется в крайнее открытое положение, а далее основываясь на том, сколько времени нужно захвату для полного закрытия, и сколько времени он уже открывался или закрывался мы рассчитываем нынешнее его положение. Несмотря на спорность данного метода, его превосходство над предыдущим способом на практике оказалось очевидным. И на данный момент синхронизация робота с моделью происходит именно так.

Еще одним решением обозначенной проблемы может стать исправление границ на стороне сервера. Такое решение представляется достаточно сложным и объемным, однако оно может быть рассмотрено в следующих работах.

### **2.3. Потокное видео с камер**

Как было замечено ранее, робот Сервосила Инженер оснащен четырьмя встроенными камерами: стереопарой, камерой с оптическим увеличением, а также камерой заднего вида. Изображение с них отправляются с помощью видеосервера [Safin et al., 2018], использующего аппаратное кодирование [Safin et al., 2020]. Одной из возможностей представленного нами графического интерфейса является отображение данных с камер. Это позволит оператору знать больше о мире, окружающем робота

В предыдущей версии графического интерфейса были проблемы с подключением видео к основному экрану. Видео открывались только в отдельных окнах, как показано на рис. 3. Работа с видеопотоком осуществляется через приложение MPlayer [Mplayer], который широко используется для работы с аудио и видео [Dantas et al., 2018]. В нашем случае данная программа использует “slave mode”, при котором все аргументы и команды передаются через командную строку, а при запуске процессу присваивается идентификатор окна, в котором он должен быть запущен.

Проблему решила реструктуризация элементов. Теперь для каждого видеопотока появилось отдельное окно, что позволило процессам публиковать данные независимо друг от друга.

Таким образом окончательный вид созданного графического пользовательского интерфейса представлен на рис. 7. На видео [Demo] можно увидеть практическое применение созданного интерфейса в условиях одного из экспериментов.



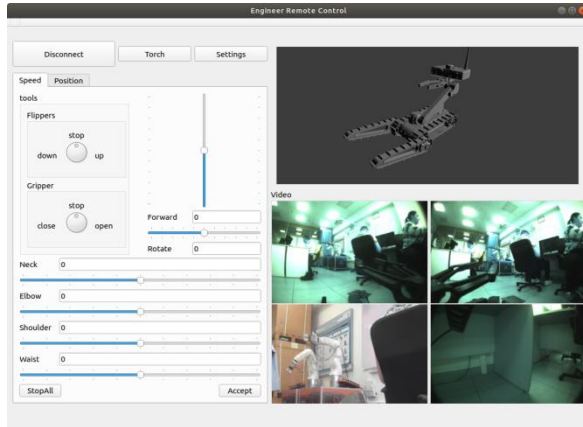


Рис. 7. Финальная версия графического интерфейса (GUI).

### 3. Вывод

Создание пользовательского графического интерфейса для работы с роботами является комплексной задачей, которая включает в себя разработку графической части на стороне оператора, обеспечение передачи данных от робота к оператору, и команд оператора роботу.

В этой статье мы представили улучшенный графический пользовательский интерфейс (GUI) для гусеничного робота Сервосила Инженер. В работе мы рассмотрели улучшение и доработку трехмерной модели робота, исправление графического отображения модели и данных с камер, а также, реализовали движение захвата на модели и его синхронизацию с реальным роботом.

### Список литературы

- [Dantas et al., 2018] Dantas R., Exton C., Le Gear A. Comparing network performance of mobile voip solutions // 6th IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (MobileCloud). 2018.
- [Dobrokvashina et al., 2020] Dobrokvashina A., Lavrenov R., Martinez-Garcia E. A., & Bai Y. Improving model of crawler robot Servosila "Engineer" for simulation in ROS/Gazebo // 13th International Conference on Developments in eSystems Engineering (DeSE). 2020.
- [Demo] Автономное возвращение робота "Сервосила Инженер". [https://www.youtube.com/watch?v=GnOrh\\_uNWYc](https://www.youtube.com/watch?v=GnOrh_uNWYc)
- [Harmer et al., 2016] Harmer S., Qt 3D: a data-driven renderer for mortals // ACM SIGGRAPH 2016 Talks. 2016.

- [**Isogaki et al., 2011**] Isogaki J., Haruta S., Man-i M., Suda K., Kawamura Y., Yoshimura F., Taniguchi K. Robot-assisted surgery for gastric cancer: experience at our institute // *Pathobiology*. 2011. 78(6).
- [**Magid et al., 2010**] Magid E., Tsubouchi T. Static Balance for Rescue Robot Navigation: Translation Motion Discretization Issue within Random Step Environment // *International Conference on Informatics in Control, Automation and Robotics (ICINCO)*. Portugal. 2010.
- [**Mavrin et al., 2018**] Mavrin, R. Lavrenov, M. Svinin, S. Sorokin, and E. Magid, Remote control library and GUI development for Russian crawler robot Servosila Engineer, *MATEC Web of Conferences*, vol. 161. 2018.
- [**Martinec et al., 2015**] Martinec T., Mlýnek J., Petrů M. Calculation of the robot trajectory for the optimum directional orientation of fibre placement in the manufacture of composite profile frames // *Robotics and Computer-Integrated Manufacturing* 35. 2015.
- [**Mplayer**] M. Team, Mplayer – the movie player. <https://mplayerhq.hu>
- [**Larson et al., 2011**] Larson J., Trivedi M. Lidar based off-road negative obstacle detection and analysis // *14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*. 2011.
- [**Lazar et al., 2016**] Lazar G., Penea R. *Mastering Qt 5* // Packt Publishing Ltd. 2016.
- [**Paolillo et al., 2018**] Paolillo A., Gergondet P., Cherubini A., Vendittelli M., Kheddar A. Autonomous car driving by a humanoid robot // *Journal of Field Robotics*. 2018. 35(2).
- [**Pecka et al., 2017**] Pecka, M., Zimmermann, K., Svoboda, T. Fast simulation of vehicles with non-deformable tracks // *International Conference on Intelligent Robots and Systems (IROS)*. 2017.
- [**Qt 3D Overview**] Qt 3D Overview. <https://doc.qt.io/qt-5/qt3d-overview.html>
- [**Safin et al., 2018**] Safin R., Lavrenov R., Tsoy T., Svinin M., Magid E. Real-Time Video Server Implementation for a Mobile Robot // *11th International Conference on Developments in eSystems Engineering (DeSE)*. 2018.
- [**Safin et al., 2020**] Safin R., Garipova E., Lavrenov R., Li H., Svinin M., Magid E. Hardware and software video encoding comparison // *59th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE)*. 2020.
- [**Servosila**] Servosila official site. <https://www.servosila.com/en/index.html>
- [**Shimchik et al., 2016**] Shimchik, I., Sagitov, A., Afanasyev, I., Matsuno, F., Magid, E. Golf cart prototype development and navigation simulation using ROS and Gazebo // *In MATEC Web of Conferences*. 2016. Vol. 75.
- [**Suzuki et al., 2014**] Suzuki S., Hasegawa S., Okugawa M. Remote control system of disaster response robot with passive sub-crawlers considering falling down avoidance // *Robomech J*. 2014. 1 (20).
- [**Wang et al., 2003**] Wang J., Lewis M., Gennari J. USAR: A game-based simulation for teleoperation // *In Proceedings of the Human Factors and Ergonomics Society Annual Meeting*. 2003. Vol. 47 No. 3.