

Module MarkerBot.GUImain

В данном модуле реализована gui на пк для задания рисунков роботу.

► [EXPAND SOURCE CODE](#)

Functions

```
def averaging()
```

Функция, которая усредняет траекторию (убирает лишние точки чрезмерную кучность точек)

► [EXPAND SOURCE CODE](#)

```
def clear_traj()
```

Функция, которая очищает массивы с координатами

► [EXPAND SOURCE CODE](#)

Classes

```
class Paint
```

Класс создания доски для рисования

Функция создания параметров доски :param self: экземпляр класса

► [EXPAND SOURCE CODE](#)

Class variables

```
var default_brush_size
```

```
var default_colour
```

Methods

```
def activate_button(self, some_button, eraser_mode=False)
```

Функция активирования кнопок :param self: экземпляр класса

► [EXPAND SOURCE CODE](#)

```
def clean_board(self)
```

Функция очистки доски :param self: экземпляр класса

► [EXPAND SOURCE CODE](#)

```
def paint(self, event)
```

Функция построения отпечатка кисти :param self: экземпляр класса

► [EXPAND SOURCE CODE](#)

```
def reset(self)
```

Функция обнуления предыдущих координат :param self: экземпляр класса

► [EXPAND SOURCE CODE](#)

```
def sent_trajectory(self)
```

Функция отправки траектории в ноду :param self: экземпляр класса

► [EXPAND SOURCE CODE](#)

```
def setup(self)
```

Функция создания параметров доски :param self: экземпляр класса

► [EXPAND SOURCE CODE](#)

```
def use_brush(self)
```

Функция переключения на кисть :param self: экземпляр класса

► [EXPAND SOURCE CODE](#)

```
def use_eraser(self)
```

Функция переключения на стирательную резинку :param self: экземпляр класса

► [EXPAND SOURCE CODE](#)

Project MarkerBot

Super-module

MarkerBot

Functions

averaging
clear_traj

Classes

Paint
activate_button
clean_board
default_brush_size
default_colour
paint
reset
sent_trajectory
setup
use_brush
use_eraser

Module MarkerBot.ROS.cros_ws.src.hrwwros_we ek1.scripts.GuiPublisher

► [EXPAND SOURCE CODE](#)

Functions

```
def PointsPublisher()
```

Функция, которая объявляет ноду и связь с топиками

► [EXPAND SOURCE CODE](#)

```
def averaging()
```

Функция, которая усредняет траекторию (убирает лишние точки чрезмерную кучность точек)

► [EXPAND SOURCE CODE](#)

```
def clear_traj()
```

Функция, которая очищает массивы с координатами

► [EXPAND SOURCE CODE](#)

Classes

```
class Paint
```

Класс создания доски для рисования

Функция создания параметров доски :param self: экземпляр класса

► [EXPAND SOURCE CODE](#)

Class variables

```
var default_brush_size
```

```
var default_colour
```

Methods

```
def activate_button(self, some_button, eraser_mode=False)
```

Функция активирования кнопок :param self: экземпляр класса

► [EXPAND SOURCE CODE](#)

```
def clean_board(self)
```

Функция очистки доски :param self: экземпляр класса

► [EXPAND SOURCE CODE](#)

```
def paint(self, event)
```

Функция построения отпечатка кисти :param self: экземпляр класса

► [EXPAND SOURCE CODE](#)

```
def reset(self, event)
```

Функция обнуления предыдущих координат :param self: экземпляр класса

► [EXPAND SOURCE CODE](#)

```
def sent_trajectory(self)
```

Функция отправки траектории в ноду :param self: экземпляр класса

► [EXPAND SOURCE CODE](#)

```
def setup(self)
```

Функция создания параметров доски :param self: экземпляр класса

► [EXPAND SOURCE CODE](#)

```
def use_brush(self)
```

Функция переключения на кисть :param self: экземпляр класса

► [EXPAND SOURCE CODE](#)

```
def use_eraser(self)
```

Функция переключения на стирательную резинку :param self: экземпляр класса

► [EXPAND SOURCE CODE](#)

Project MarkerBot

Super-module

[MarkerBot.ROS.cros_ws.src.hrwwros_week1.scripts](#)

Functions

[PointsPublisher](#)
[averaging](#)
[clear_traj](#)

Classes

Paint
[activate_button](#)
[clean_board](#)
[default_brush_size](#)

default_colour
paint
reset
sent_trajectory
setup
use_brush
use_eraser

Namespace MarkerBot

MarkerBot - это проект робота, который может создавать рисунки на маркерной доске на основе отправленного ему изображения. Вам просто нужно указать изображение на вашем компьютере, которое вы хотите распечатать, и робот будет использовать маркеры, чтобы нарисовать его.

Sub-modules

`MarkerBot.GUImain`

`MarkerBot.R0S`

`MarkerBot.camera_vision`

`MarkerBot.robot_worker`

Project MarkerBot

Sub-modules

`MarkerBot.GUImain`

`MarkerBot.R0S`

`MarkerBot.camera_vision`

`MarkerBot.robot_worker`

Namespace MarkerBot.camera_vision

Модуль работы с камерой и техническим зрением.

Sub-modules

`MarkerBot.camera_vision.Client`

`MarkerBot.camera_vision.board_detection`

`MarkerBot.camera_vision.robot_detection`

Project MarkerBot

Super-module

`MarkerBot`

Sub-modules

`MarkerBot.camera_vision.Client`

`MarkerBot.camera_vision.board_detection`

`MarkerBot.camera_vision.robot_detection`

Namespace MarkerBot.robot_worker

Модуль работы с ev3 роботом.

Sub-modules

[MarkerBot.robot_worker.Server](#)

[MarkerBot.robot_worker.main](#)

Project MarkerBot

Super-module

[MarkerBot](#)

Sub-modules

[MarkerBot.robot_worker.Server](#)

[MarkerBot.robot_worker.main](#)

Namespace MarkerBot.R0S

Модули работы с ROS.

Sub-modules

[MarkerBot.R0S.cros_ws](#)

Project MarkerBot

Super-module

[MarkerBot](#)

Sub-modules

[MarkerBot.R0S.cros_ws](#)

Namespace MarkerBot.R0S.cros_ws

Модули работы с ROS.

Sub-modules

[MarkerBot.R0S.cros_ws.src](#)

Project MarkerBot

Super-module

[MarkerBot.R0S](#)

Sub-modules

[MarkerBot.R0S.cros_ws.src](#)

Namespace MarkerBot.ROS.cros_ws.src.hrwwros_week1.scripts

Модули работы Ros.

Sub-modules

MarkerBot.ROS.cros_ws.src.hrwwros_week1.scripts.GuiPublisher

MarkerBot.ROS.cros_ws.src.hrwwros_week1.scripts.coordinates_server

MarkerBot.ROS.cros_ws.src.hrwwros_week1.scripts.detector

Project MarkerBot

Super-module

MarkerBot.ROS.cros_ws.src.hrwwros_week1

Sub-modules

MarkerBot.ROS.cros_ws.src.hrwwros_week1.scripts.GuiPublisher

MarkerBot.ROS.cros_ws.src.hrwwros_week1.scripts.coordinates_server

MarkerBot.ROS.cros_ws.src.hrwwros_week1.scripts.detector

Module MarkerBot.robot_worker.main

Модуль движения ev3.

► [EXPAND SOURCE CODE](#)

Functions

```
def filter(sp)
```

Функция для фильтрации значений напряжения

:param sp: напряжение подаваемое на мотор :return: отфильтрованное напряжение

► [EXPAND SOURCE CODE](#)

```
def getmL()
```

Функция возвращающая позицию левого мотора

:return:позиция левого мотора в радианах

► [EXPAND SOURCE CODE](#)

```
def getmR()
```

Функция возвращающая позицию правого мотора

:return:позиция правого мотора в радианах

► [EXPAND SOURCE CODE](#)

```
def main(Rad, Base, mL, mR, mM, fh, btn)
```

Основная функция, осуществляющая движения робота по заданному рисунку.

:param getX: координаты X по которым движется робот

:param getY: координаты Y по которым движется робот

:param getM: флаги поднятия и опускания маркера

:param Rad: радиус колес

:param Base: расстояние между колесами

:param mL: левый мотор

:param mR: правый мотор

:param mM: мотор маркера

:param fh: файл записи логов

:param btn: кнопка включения робота

► [EXPAND SOURCE CODE](#)

```
def move_marker(getM, flag_down, iterations, mM)
```

Функция осуществляющая поднятие и опускание маркера в зависимости от рисунка

:param getM: входной массив с метками рисовать или нет

:param flag_down: флаг состояния маркера: опущен или поднят

:param iterations: итерации цикла

:param mM: мотор прикрепленный к маркеру

Project MarkerBot

Super-module

`MarkerBot.robot_worker`

Functions

`filter`
`getmL`
`getmR`
`main`
`move_marker`

Module MarkerBot.camera_vision.robot_detection

Модуль для детекции робота на изображении.

► [EXPAND SOURCE CODE](#)

Functions

```
def detect_label(img, h_min, h_max)
```

Детектирование метки на изображении вычисление по порогу цвета и основным моментам изображения.

:param img: np.array image
:param h_min: (h, s, v) кортеж мин. HSV трешхолдов
:param h_max: (h, s, v) кортеж макс. HSV трешхолдов

:return: (x, y) - Координаты центра метки

► [EXPAND SOURCE CODE](#)

```
def detect_robot_coords(img, previous_values,  
resize=False)
```

Детектирование координат робота по фотографии и меткам.

:param img: np.array текущее изображение для детектирования
:param previous_values: Результат предыдущего удачного детектирования
:param resize: Изменение изображения в более удобный вид (для теста)

:return: (x, y) координаты (x, y) центра , img детекции,

► [EXPAND SOURCE CODE](#)

```
def nothing(x)
```

Функция для теста

:param x: - :return: -

► [EXPAND SOURCE CODE](#)

```
def tuning_color_filter(test_img)
```

Вспомогательная функция для настройки порога цвета метки.

:param test_img: np.array - image
:return:

► [EXPAND SOURCE CODE](#)

Project MarkerBot

Super-module

`MarkerBot.camera_vision`

Functions

`detect_label`
`detect_robot_coords`
`nothing`
`tuning_color_filter`

Module MarkerBot.robot_worker.Server

Модуль связи с сервером.

► [EXPAND SOURCE CODE](#)

Functions

```
def closeConnection(conn)
```

Функция для закрытия соединения
:param conn: подключение для принятия данных
:return:

► [EXPAND SOURCE CODE](#)

```
def createSocket()
```

Функция, создающая socket для подключения
:return: conn - подключение для принятия данных

► [EXPAND SOURCE CODE](#)

```
def receiveData(conn)
```

Функция, которая принимает данные о координатах с клиента
:param conn: подключение для принятия данных
:return: startPositionXY - arraylist начальных координат
x - arraylist координат по X
y - arraylist координат по Y
flags - arraylist поднятия и опускания маркера

► [EXPAND SOURCE CODE](#)

Project MarkerBot

Super-module

[MarkerBot.robot_worker](#)

Functions

[closeConnection](#)
[createSocket](#)
[receiveData](#)

Module MarkerBot.camera_vision.board_detection

Модуль для детекции доски на изображении.

► [EXPAND SOURCE CODE](#)

Functions

```
def find_board(img: , test=False, lenght=883, min_size=527132, max_size=1500000, centroid=True, record=False)
```

Функция для поиска доски на изображении

:param img: RGB изображение np.array
:param test: bool настройка параметров
:param lenght: int Минимальная длина линии доски
:param min_size: int Минимальный размер определившейся доски
:param max_size: int Максимальный размер определившейся доски
:param centroid: bool Центрирована ли доска на frame
:param record: запись проходных изображений

:return: img, flag - img - перспективное преобразование доски (кроп доски)
flag - True - преобразование выполнено; False - преобразование не выполнено, img - оригинальная картинка

► [EXPAND SOURCE CODE](#)

```
def nothing(x)
```

Функция для теста

:param x: - :return: -

► [EXPAND SOURCE CODE](#)

Project MarkerBot

Super-module

[MarkerBot.camera_vision](#)

Functions

[find_board](#)
[nothing](#)

Module MarkerBot.camera_vision.Client

Модуль связи с ev3.

► [EXPAND SOURCE CODE](#)

Functions

```
def sendData(x, y, flags)
```

Функция для отправки всех имеющихся данных (начальных позиций, координат x и y рисунка) на Brick EV3

:param x: arraylist координат по x

:param y: arraylist координат по y

:param flags: arraylist поднятия и опускания маркера

► [EXPAND SOURCE CODE](#)

Project MarkerBot

Super-module

[MarkerBot.camera_vision](#)

Functions

[sendData](#)

Generated by [pdoc](#)

0.9.2.

Module MarkerBot.ROS.cros_ws.src.hrwros_week1.scripts.coordinates_server

► [EXPAND SOURCE CODE](#)

Functions

```
def stringListener()
```

Listener нода сервера.

:return:

► [EXPAND SOURCE CODE](#)

```
def stringListenerCallback(data)
```

Log - функция

:param data: Координаты робота из ноды.

:return:

► [EXPAND SOURCE CODE](#)

Project MarkerBot

Super-module

[MarkerBot.ROS.cros_ws.src.hrwros_week1.scripts](#)

Functions

[stringListener](#)

[stringListenerCallback](#)

Module

MarkerBot.camera_vision.detector

► [EXPAND SOURCE CODE](#)

Functions

```
def detect()
```

► [EXPAND SOURCE CODE](#)

```
def test(img)
```

► [EXPAND SOURCE CODE](#)

Index

Super-module

[MarkerBot.camera_vision](#)

Functions

[detect](#)

[test](#)

Module MarkerBot.ROS.cros_ws.src.hrwros_week1.scripts.detector

► [EXPAND SOURCE CODE](#)

Functions

```
def Detector()
```

Нода детекции робота. Передает в string топик значения координат робота.

► [EXPAND SOURCE CODE](#)

Project MarkerBot

Super-module

[MarkerBot.ROS.cros_ws.src.hrwros_week1.scripts](#)

Functions

[Detector](#)