

Санкт-Петербургский политехнический университет Петра Великого
Институт машиностроения, материалов и транспорта
Высшая школа автоматизации и робототехники

Отчёт

по лабораторной работе №1

Дисциплина: Техническое зрение

Тема: Моделирование движения робота с использованием библиотеки OpenCV

Студент гр. 3331506/70401

Ляховский М.В.

Преподаватель

Варлашин В.В.

« » _____ 2020 г.

Санкт-Петербург

2020

Задание

С помощью методов *OpenCV* реализовать движение робота по заданному полю, его поворот на месте, а также ограничение на выезд за пределы.

Ход работы

Общий алгоритм выполнения программы:

- 1) Ожидание нажатия клавиши управления;
- 2) Вычисление новых координат точек робота, в зависимости от нажатой клавиши;
- 3) Проверка на пересечение границ заданного поля и точек робота;
- 4) Вывод нового изображения робота на экран.

Весь процесс находится в цикле *while()*, пока не будет нажата клавиша «*esc*».

Чтобы избежать отображения прошлых изображений робота, создан отдельный объект класса *Mat* – общий фон. Перед выводом нового изображения старое затирается общим фоном, на который помещается робот. Он состоит из 4 линий, заданных 4 точками. Координаты этих точек изменяются по формуле:

$$\begin{pmatrix} X \\ Y \\ 1 \end{pmatrix} = \begin{pmatrix} \cos(\varphi) & -\sin(\varphi) & t_x \\ \sin(\varphi) & \cos(\varphi) & t_y \\ 0 & 0 & 1 \end{pmatrix} \times \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

Вывод всех линий на экран производится с помощью встроенной в *OpenCV* функции *cv::line()*.

Класс *MyRobot*

Для робота создан класс *MyRobot* с параметрами и методами, показанными на рисунках 1 и 2.

```
void move(float Vx, float Vy, float Wz, float Wz_gun);  
  
void draw(cv::Mat* image_address);  
  
bool checkBorder();
```

Рисунок 1 – Методы класса

```
private:
    //world coordinates of rectangle center
    cv::Point2f m_center;
    cv::Point2f m_center_old;

    //local coordinates of points
    cv::Point2f m_array_const_points[8];

    //image coordinates of points
    cv::Point2f m_array_points[8];

    //image coordinates of old points
    cv::Point2f m_array_points_old[8];

    //current robot angle
    float m_angle;
    float m_gun_angle;

    //size of robot's body
    float m_width;
    float m_height;

    //size of robot's gun
    float m_gun_width;
    float m_gun_height;
    int m_gun_radius;

    //size of wheel
    float m_wheel_width;
    float m_wheel_diameter;

    int m_obstacle_size;
    cv::Point2f m_obstacle[2];

    //flag for area limitation
    bool is_border_reached;
```

Рисунок 2 – Параметры класса

Все параметры робота: размеры, скорости задаются в конструкторе класса.

Основные функции

Метод *move()* считает новые координаты точек робота исходя из 4 входных параметров: скорости по горизонтальной и вертикальной оси, скорости вращения корпуса и башни. Проверяет, заходят ли новые координаты за границы изображения. Если заходят – в новые координаты записываются старые и изображение не отличается от прошлого, если не заходят – в старые координаты записываются новые и рисуется новое изображение.

Метод *draw()* клонирует фон изображения, затирая тем самым старое, и рисует робота в новом положении.

Вывод

В ходе работы проведено успешное моделирование движения робота по заданной области. Успешно изучены и использованы необходимые алгоритмы *OpenCV*.