МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

НИЖЕГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ

УНИВЕРСИТЕТ им. Р.Е.АЛЕКСЕЕВА

Институт радиоэлектроники и информационных технологий

Кафедра «Вычислительные системы и технологии»

Отчет по лабораторной работе №3

по дисциплине

«Аппаратное и программное обеспечение робототехнических систем»

РУКОВОДИТЕЛЬ:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Гай В.Е

СТУДЕНТ:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Куприхин Д.А.

ГРУППА:

19-В-1

Работа защищена «\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_

С оценкой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Нижний Новгород

2022г.

Задание: Управление автомобилем

Разработать контроллер для автомобиля, который позволит переместиться автомобилю как можно дальше за одну минуту времени без столкновений с другими транспортными средствами или статическими объектами.

Контроллер

import os

import sys

from controller import Camera

from controller import CameraRecognitionObject

from controller import Compass

*# операции с путями для правильного нахождения автомобильного модуля*

try:

libraryPath = os.path.join(os.environ.get('WEBOTS\_HOME'), 'projects', 'vehicles', 'libraries', 'python')

libraryPath = libraryPath.replace('/', os.sep)

sys.path.append(libraryPath)

from automobile import Driver

except ImportError:

sys.stderr.write("Warning: 'automobile' module not found.\n")

sys.exit(0)

*# название доступных датчиков расстояния*

sensorsNames = [

'front',

'front right 0',

'front right 1',

'front right 2',

'front left 0',

'front left 1',

'front left 2',

'rear',

'rear left',

'rear right',

'right',

'left']

sensors = {}

maxSpeed = 80

driver = Driver()

driver.setSteeringAngle(0.0) # go straight

turn=0

*# получить и включить датчики расстояния*

for name in sensorsNames:

sensors[name] = driver.getDistanceSensor('distance sensor ' + name)

sensors[name].enable(10)

# получить и включить GPS

gps = driver.getGPS('gps')

gps.enable(10)

# получить камеру

camera = driver.getCamera('camera')

# uncomment those lines to enable the camera

camera.enable(50)

camera.recognitionEnable(50)

#cameraobject= driver.CameraRecognitionObject('camera')

orientation=[]

position=[]

orientation2=[]

position2=[]

print(gps.getCoordinateSystem())

while driver.step() != -1:

# adjust the speed according to the value returned by the front distance sensor

frontDistance = sensors['front'].getValue()

frontRange = sensors['front'].getMaxValue()

leftDistance = sensors['left'].getValue()

leftRange = sensors['left'].getMaxValue()

rightDistance = sensors['right'].getValue()

rightRange = sensors['right'].getMaxValue()

frontl0Distance = sensors['front left 0'].getValue()

frontl0Range = sensors['front left 0'].getMaxValue()

frontl1Distance = sensors['front left 1'].getValue()

frontl1Range = sensors['front left 1'].getMaxValue()

frontl2Distance = sensors['front left 2'].getValue()

frontl2Range = sensors['front left 2'].getMaxValue()

frontr0Distance = sensors['front right 0'].getValue()

frontr0Range = sensors['front right 0'].getMaxValue()

frontr1Distance = sensors['front right 1'].getValue()

frontr1Range = sensors['front right 1'].getMaxValue()

frontr2Distance = sensors['front right 2'].getValue()

frontr2Range = sensors['front right 2'].getMaxValue()

speed = maxSpeed \* (frontDistance / frontRange)

driver.setCruisingSpeed(speed)

# brake if we need to reduce the speed

speedDiff = driver.getCurrentSpeed() - speed

gpscord = gps.getValues()

if camera.getRecognitionNumberOfObjects()>=1:

Objects = camera.getRecognitionObjects()

found = 0

counter = 0

while (found!=2 and counter<(camera.getRecognitionNumberOfObjects()-1)):

if Objects[counter].get\_id() == 24 :#id of right fece

orientation = Objects[counter].get\_orientation()

position = Objects[counter].get\_position()

counterid=counter

found =found+1

elif Objects[counter].get\_id() == 58 :#id of left fece

orientation2 = Objects[counter].get\_orientation()

position2 = Objects[counter].get\_position()

counterid2=counter

found =found+1

counter+=1

if speedDiff > 0:

driver.setBrakeIntensity(min(speedDiff / speed, 1))

if frontl0Distance>2 and leftDistance>5 and turn==0:

driver.setIndicator(2)

driver.setCruisingSpeed(60)

driver.setSteeringAngle(-0.0245)

for i in range(0,182):

driver.step()

driver.setSteeringAngle(0.025)

for i in range(0,185):

driver.step()

for i in range(0,7):

driver.setSteeringAngle(-0.013+i\*0.001)

driver.step()

driver.setSteeringAngle(0.0)

driver.setIndicator(0)

turn=1

else:

driver.setBrakeIntensity(min(speedDiff / speed, 1))

if len(orientation2)>1:

if abs(orientation2[1])<0.9999945:

if orientation2[1]<0:

driver.setSteeringAngle(-0.015)

else:

driver.setSteeringAngle(0.015)

else:

driver.setSteeringAngle(0.0)

elif len(orientation)>1:

#print(orientation[1])

if abs(orientation[1])<0.9999945:

if orientation[1]<0:

driver.setSteeringAngle(-0.015)

else:

driver.setSteeringAngle(0.015)

else:

driver.setSteeringAngle(0.0)

driver.setBrakeIntensity(0)

**Результат работы:**

