

گزارش تمرین ۲

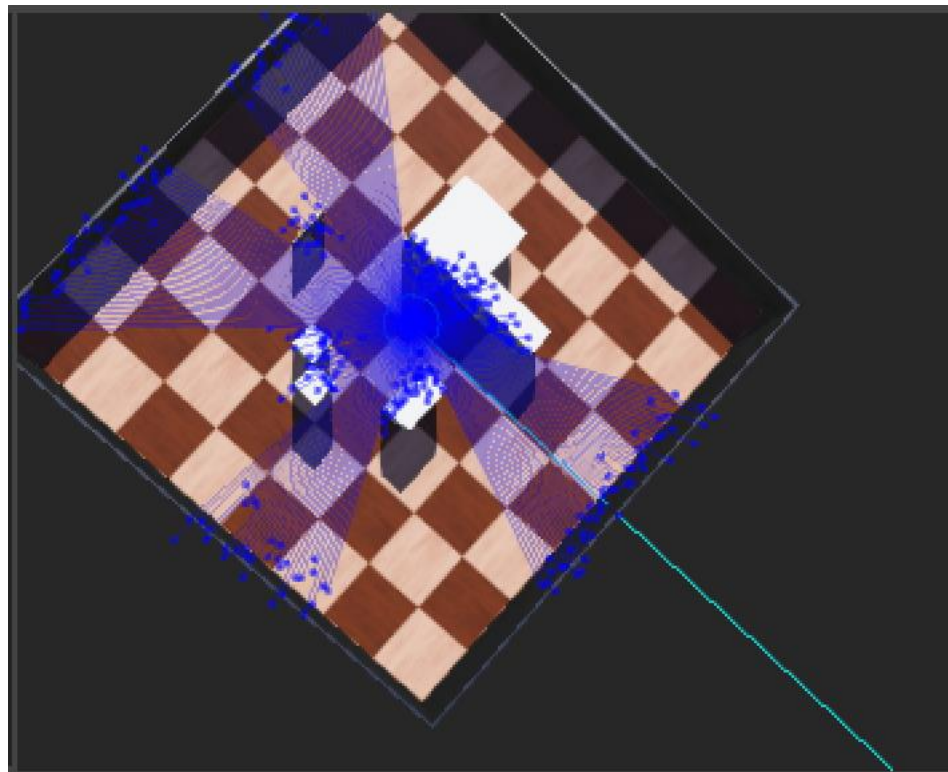
سید آرمین جلالی منفرد - ۹۷۲۴۳۱۱۳

سید محمدرضا حسینی - ۹۷۲۴۳۱۲۹

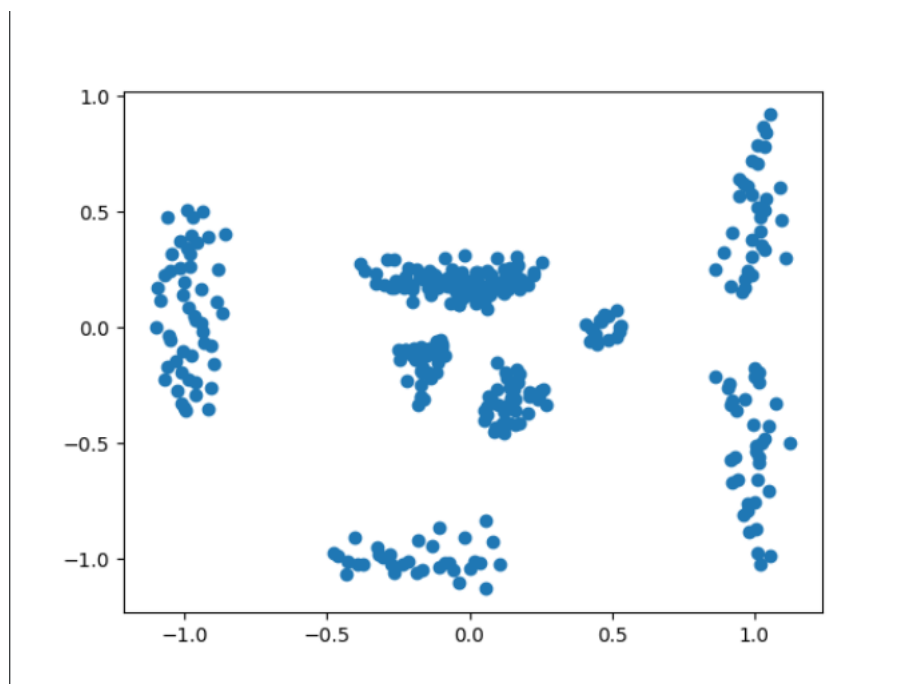
سید عباس میرقاسمی - ۹۷۲۴۳۰۶۸

سوال ۱

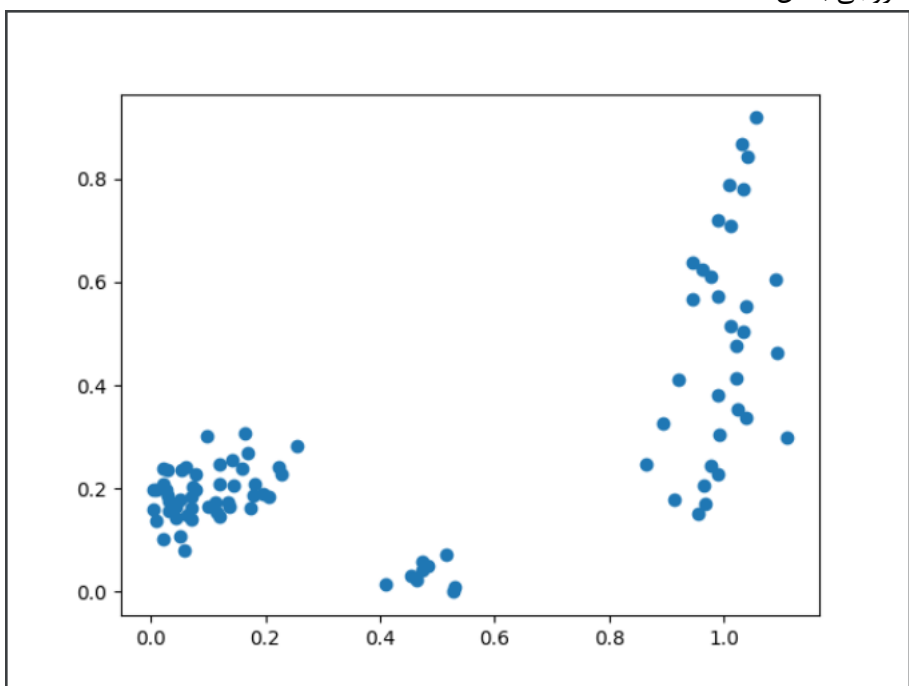
در این سوال ابتدا داخل ویباتس برای ربات epuck در بخش torretslot آمدیم lidar را از داخل basen node ها به رباتمان اضافه کردیم سپس تنظیماتش را در همان بخش انجام دادیم و noise را برابر ۰.۰۵ قرار دادیم بر روی lidar کلیک کردیم و از خود ربات سرش را تنظیم کردیم و رادار را آوردیم بالا تا خود ربات را به عنوان شی نشناسد.



حال تا زمانی که inf نگیریم از رادار اطلاعات میگیریم و به محض گرفتن یک داده مناسب، از آن خارج میشیم و آن داده را در فایل a.txt ذخیره میکنیم.
خروجی ۳۶۰ درجه :

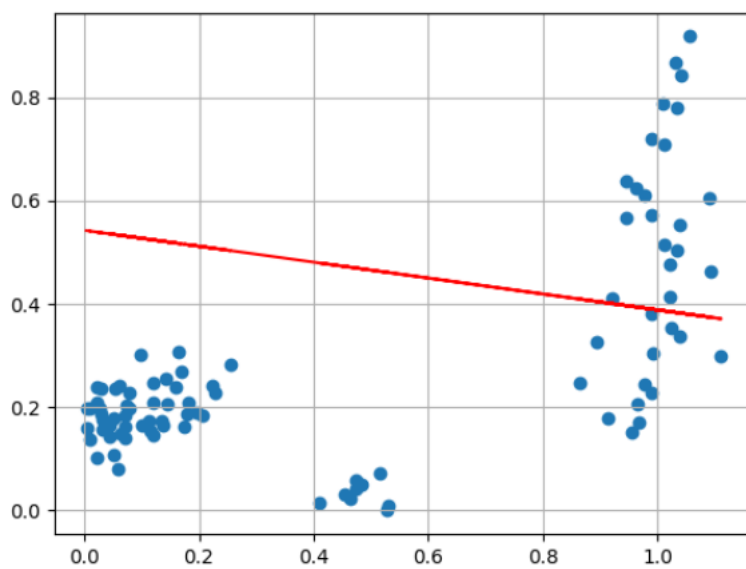


خروجی بخش الف:

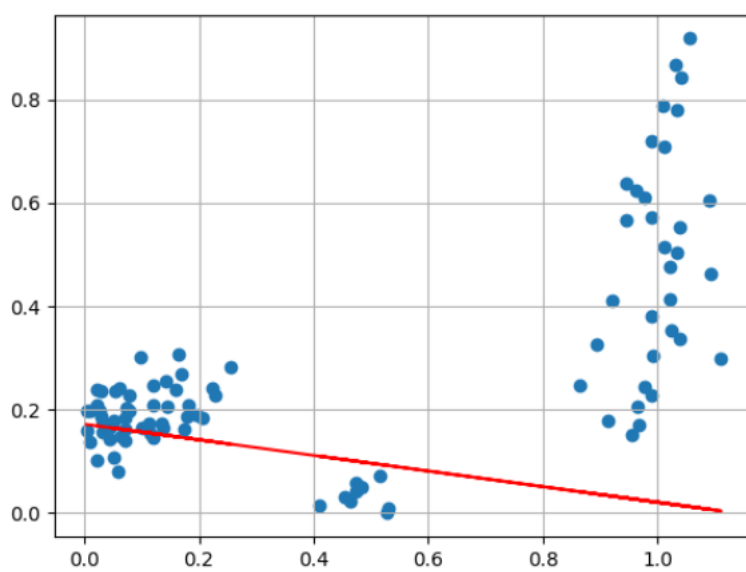


در حالت ۹۰ درجه

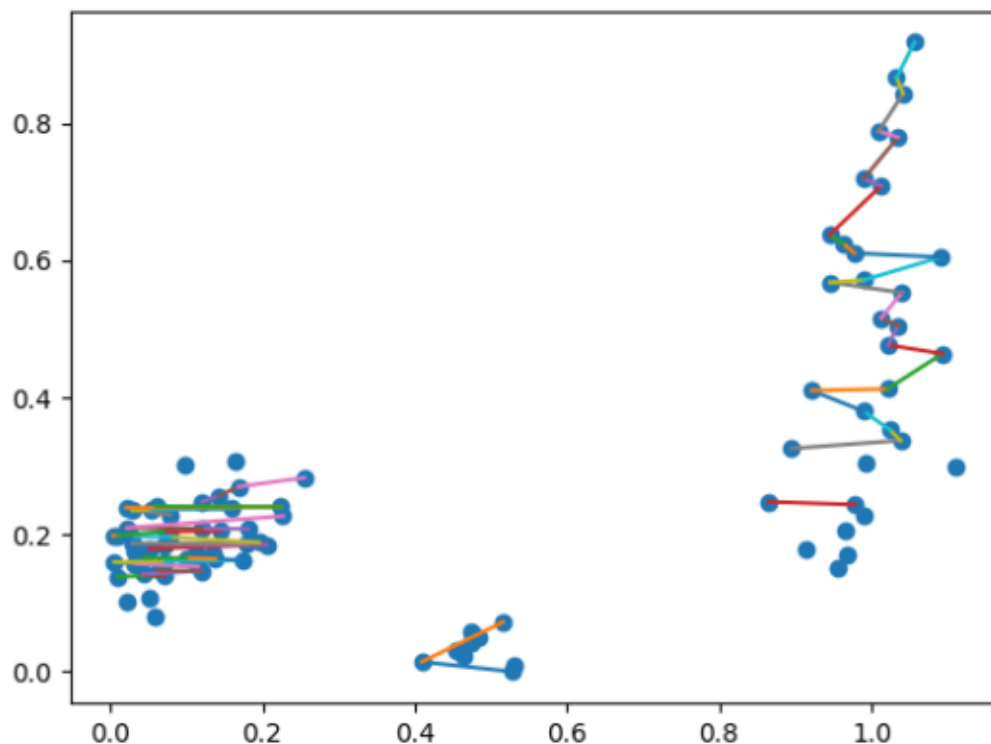
خروجی سوال ب:



خروجی سوال ج:



خروجی سوال د :



سوال ۲

در این سوال ابتدا داخل ویباتس برای ربات epuck در بخش torretslot آمدیم camera را از داخل basen node ها به رباتمان اضافه کردیم سپس تنظیماتش را در همان بخش انجام دادیم و noise را برابر 0.2 قرار دادیم بر روی camera کلیک کردیم و از خود ربات سرش را تنظیم کردیم و دوربین را آوردیم بالا تر تا تابلوی عکس را پوشش دهد. همچنین کنترلی را با نام cam_controller ساختیم و در آن ربات را به حرکت می آوریم، Camera را در آن import می کنیم و همچنین camera ای که تعریف کردیم یعنی 'cam' را با آن ست می کنیم سپس camera مان را enable کردیم و داخل while هم عکس می گیریم و برای save کردن هم image.jpg زدیم که در پوشه ی controller مان

عکس را ذخیره می کند.

```
cam_controller.py Open with Visual Studio Code
from controller import Robot, Camera
TIME_STEP = 64
robot = Robot()
max_speed = 6.28

left_motor = robot.getDevice('left wheel motor')
right_motor = robot.getDevice('right wheel motor')
left_motor.setPosition(float('inf'))
right_motor.setPosition(float('inf'))

left_motor.setVelocity(1)
right_motor.setVelocity(1)

camera = robot.getDevice('cam')
camera.enable(TIME_STEP)

while robot.step(TIME_STEP) != -1:
    camera.getImage()
    camera.saveImage('image.jpg', 100)
```

در ادامه داخل متلب آمدیم و این عکس را رویش فیلتر gaussian و prewitt را اعمال کردیم.

برای فیلتر gaussian آمدیم convolution نوشتیم و کرنل مربوط به gaussian که مطابق شکل زیر است را به آن اعمال

1	4	7	4	1
4	16	26	16	4
7	26	41	26	7
4	16	26	16	4
1	4	7	4	1

$\frac{1}{273}$

کردیم.

برای فیلتر **prewitt** هم آمدیم کرنل آن که به صورت افقی و عمودی است را به آن اعمال کردیم که به صورت زیر است

(a)

-1	0	1
-1	0	1
-1	0	1

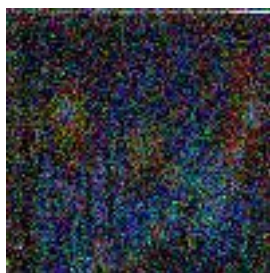
Horizontal (f_x)

(b)

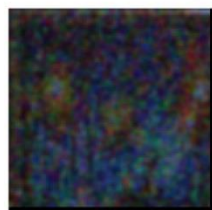
-1	-1	-1
0	0	0
1	1	1

Vertical (f_y)

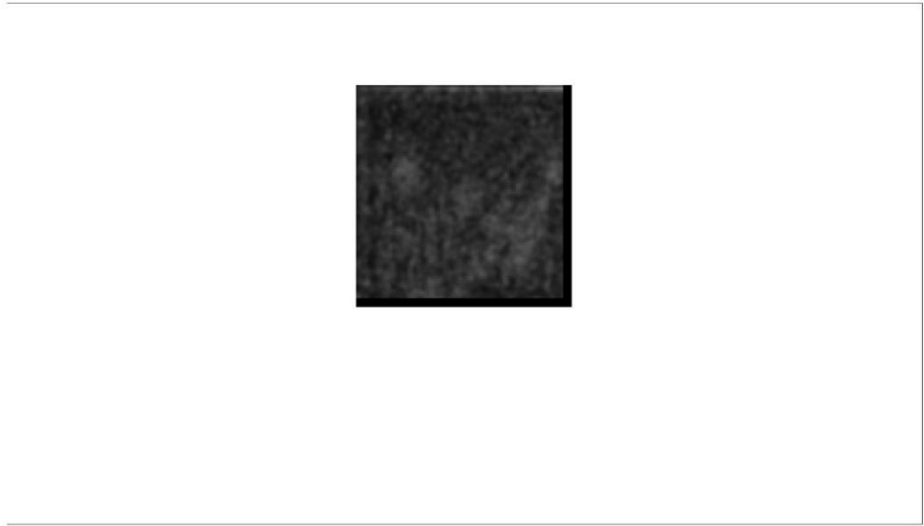
عکسی که ربات گرفت و **save** کردیم در پوشه ی **cam_controller** ذخیره شد :

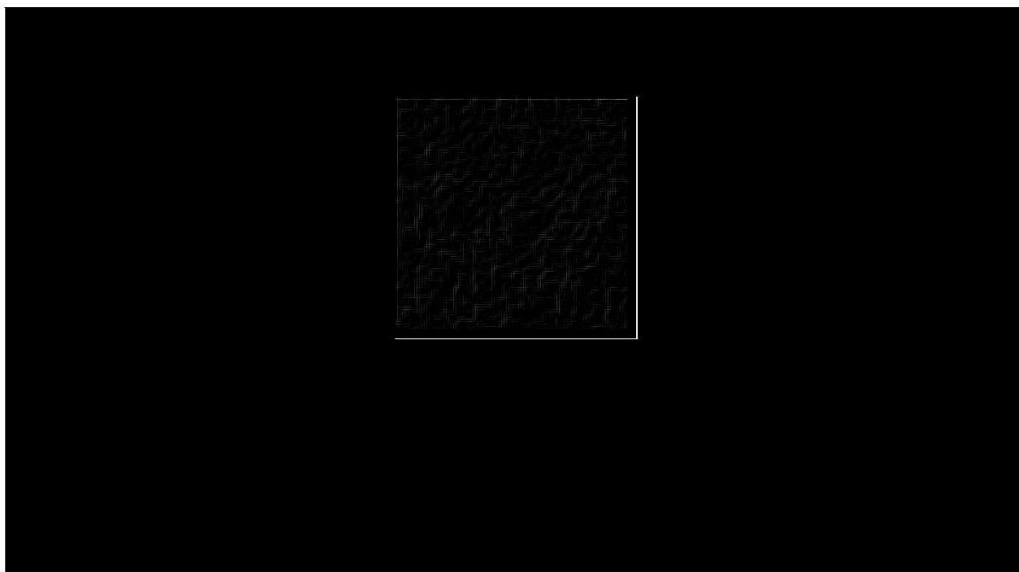


اعمال فیلتر **gaussian** :



اعمال فیلتر **prewitt** پس از هموار شدن عکس توسط فیلتر **gaussian** :





که به ترتیب نتیجه ی اعمال ارجی بی تو گری و کرنل عمودی و افقی و هر دوی افقی و عمودی کرنل ها همزمان در تابع پرویت می باشد.

عکس ها را به صورت جدا در پوشه ی فیلتر ها با نام مناسب که مشخص باشد قرار دادیم.