CS 3630 Project 4 Report

Name: Anthony Wong
GT email: awong307@gatech.edu
GT username: awong307

1.1) What is the shape of the feature map after an image from the kMNIST dataset is passed through the first convolution layer of SimpleNet?
(Answer as [# Channels, Height, Width])

[5, 10, 10]

1.2) Count the Number of Parameters:

- Relu Layer: 0
- AvgPool2D: 0
- The last Linear Layer (self.linear1): 490

1.3) What will be the size of the output when we apply the nn.AvgPool2d function on an input image of dimension [3x466x700]?
(kernel size: 4x4, stride: 1, padding: 0)

[3x463x697]

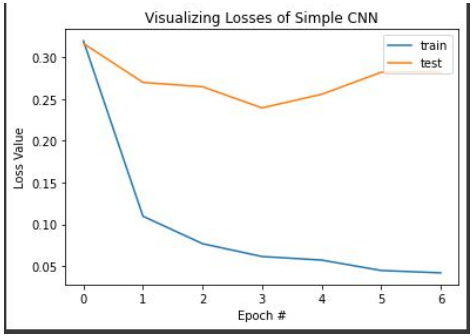
1.4) In one line, explain how average pooling is different from max pooling.

Instead of taking the max of a group of values, we take the average of the group, resulting in the ability to smoothly extract features rather than just the most prominent ones.

1.5) Try different combinations for the set of hyperparamters:
- batch size = {32, 64, 128}
- learning rate = {0.01, 0.05, 0.005, 0.001}
- You may also increase the number of epochs
Which set of parameters worked best for you? Explain your findings.

The set of parameters that worked the best was batch size 64, learning rate, 0.005, and 7 epochs. We are constantly finding the balance between training enough to converge towards high accuracy and overtraining. 0.005 was a good learning rate for convergence and a batch size of 64 and 7 epochs was enough data to train good accuracy but not overtrain.

1.6) Paste a screenshot showing the loss vs epoch of your LeNet. State the train and test accuracies.



Model Predicted 59539 correctly out of 60000 from training dataset, Accuracy : 99.23

Model Predicted 9408 correctly out of 10000 from testing dataset, Accuracy : 94.08

Points for >= XX% test accuracy

2.1) Read through the code in the Setup section. Answer the questions below.

1. How many obstacles are in the environment? Exclude the walls along the corners of the environment.

   10

2. What does the integer ‘0’ in the images_list (ex. Image([0,3], ‘e’, **0**) represent?

   The location within the environment where the image is located.

2.2) When randomly sampling points for RRT, what is the purpose of returning the goal position with some probability?

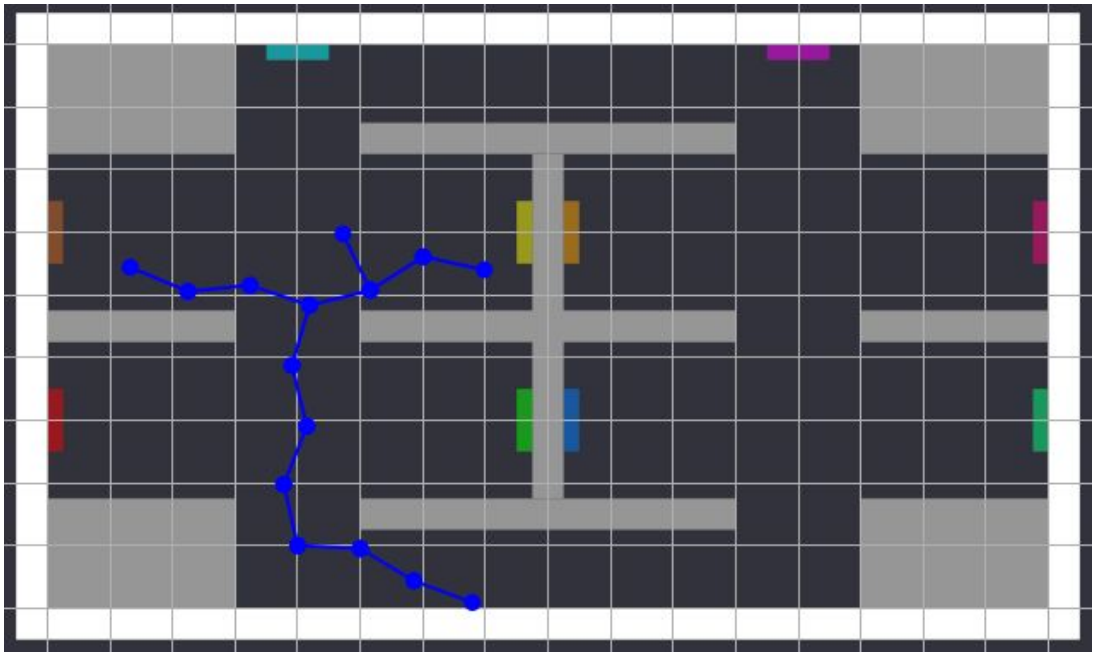
This ensures that we are not wandering around the environment completely randomly. We always have a chance of navigating towards our goal position, meaning we find a path to the goal quicker.

2.3) Write the high-level pseudocode for the RRT algorithm here. Add an explanation of each step beside the pseudocode.

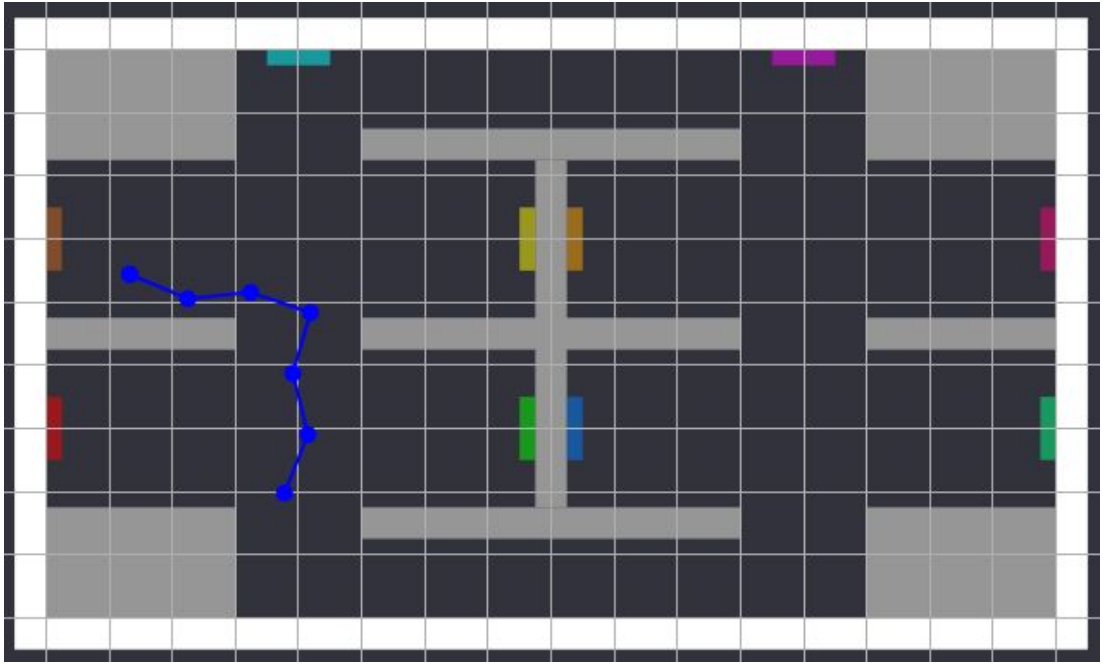
```
for i = 1 to N:
        x_rand = RandomSample()
        x_near = Nearest(x_rand)
        x_new = Steer(x_near, x_rand)
        if ObstacleFree(x_near, x_new):
                add vertex x_new
                add edge (x_new, x_near)
```

We first sample a random point within our environment to serve as the next point in our tree or the direction of the next point. It's parent is the closest node in our RRT tree so we then need to find that. Because this is a slowly expanding tree, we have a max step length such that if a random sample is too far away, we only need to choose a node in that direction, which is what Steer does. If the new node is not in an obstacle nor does its path to it collides with an obstacle, we can safely add it to our vertex and edge set.

2.4) Attach a screenshot of the rapidly-exploring random tree (RRT) generated to find the path from [4,1] to goal position [0,6].

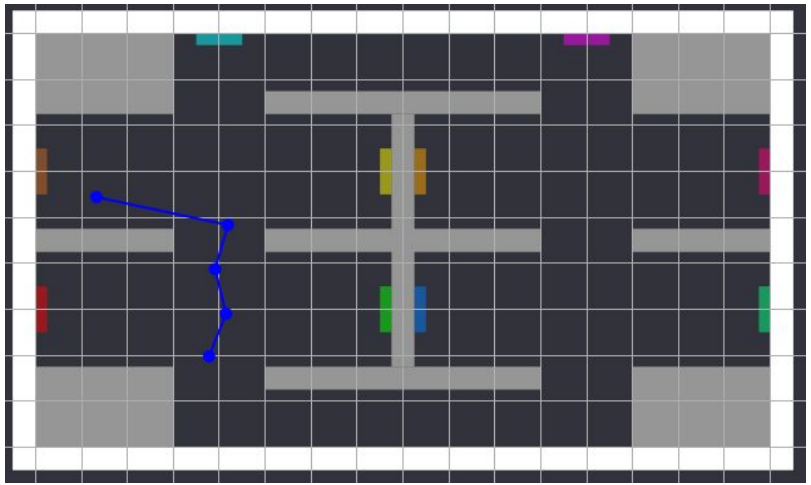


2.5) Attach a screenshot of the path found from the tree in 2.4 without path smoothing.

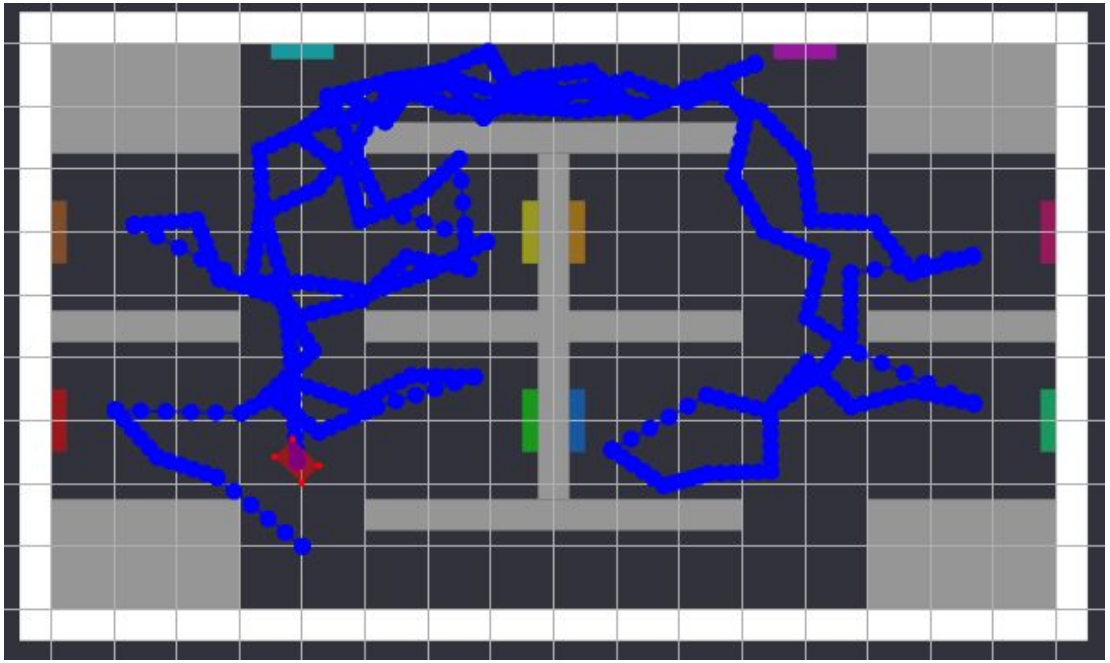


2.6) Attach a screenshot of the path found from the tree in 2.4 with path smoothing. Explain the differences with and without smoothing. If you haven’t implemented smoothing, just explain the expected differences in the path when smoothing is applied.

Smoothing removes non-integral points in the path by bypassing them and directly connecting two other dots. This also can decrease the distance of the path.



3.1) Attach a screenshot of the paths generated by the robot to look at all ten images in the museum.



3.2) The RRT algorithm implemented in this lab plans paths composed of straight-line segments. How would you edit the differential drive section to plan smooth curved paths?

The sections currently only is comprised of a pure rotation and pure translation. A combination of these two functions are needed such that you can call ddr_ik() with a translational and rotational velocity.

3. Feedback

Please provide feedback on the coding portion of the project. How did it help your understanding of the material? Is there anything that you think could have been made more clear?

I think the coding portion was very good in clarifying the general concepts taught in class