CBAGAN-RRT: Convolutional Block Attention Generative Adversarial Network for Sampling-Based Path Planning

Abhinav Sagar
119116371

Sai Teja Gilukara
119369623

Link to video of presentation: link

Introduction

1. A common problem among RRT based algorithms is that the initial path generated is not optimal and the convergence is too slow to be used in real world applications.

1. We propose a novel image based learning algorithm (CBAGAN-RRT) using a Convolutional Block Attention Generative Adversarial Network with a combination of spatial and channel attention, and a novel loss function to design the heuristics, find a better optimal path, and to improve the convergence of the algorithm both with respect to time and speed.

1. We demonstrate that our algorithm outperforms previous state of the art algorithms using both the image quality generation metrics like IOU Score, Dice Score, FID score and path planning metrics like time cost and the number of nodes.

1. Our approach can avoid the complicated preprocessing in the state space, can be generalized to complicated environments like those containing turns and narrow passages without loss of accuracy, and can be easily integrated with other sampling based path planning algorithms.

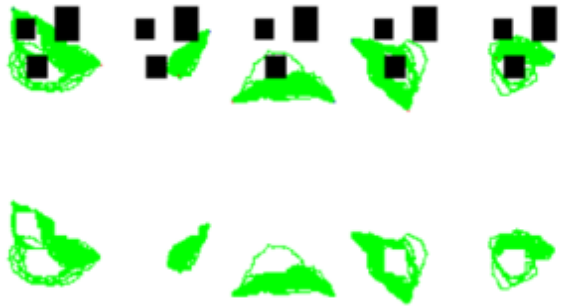
Data

1. The dataset was generated by randomly placing different obstacles on the map and randomly sampling the start and goal nodes which are denoted by red and blue dots in the map respectively.

1. The RRT algorithm was run to generate the feasible path which is shown in green color or the ground truth.

1. The dimensions of all the images are (3x64x64) where the height and the width of the images are 64 and the number of channels is 3.

1. We use 8000 images for training and 2000 images for testing.



Data Augmentation

The following data augmentation methods were used to increase the quality and quantity of the dataset:

1. Rescaling: We rescale the pixels values by rescaling factor 1/255.
2. Rotation: Random rotations with the setup degree range between [0, 360] were used.
3. Height and Width Shift: Shifting the input to the left or right and up or down was performed.
4. Shearing Intensity: It refers to shear angle (unit in degrees) in counter-clockwise direction.
5. Brightness: It uses a brightness shift value from the setup range.

Network Architecture

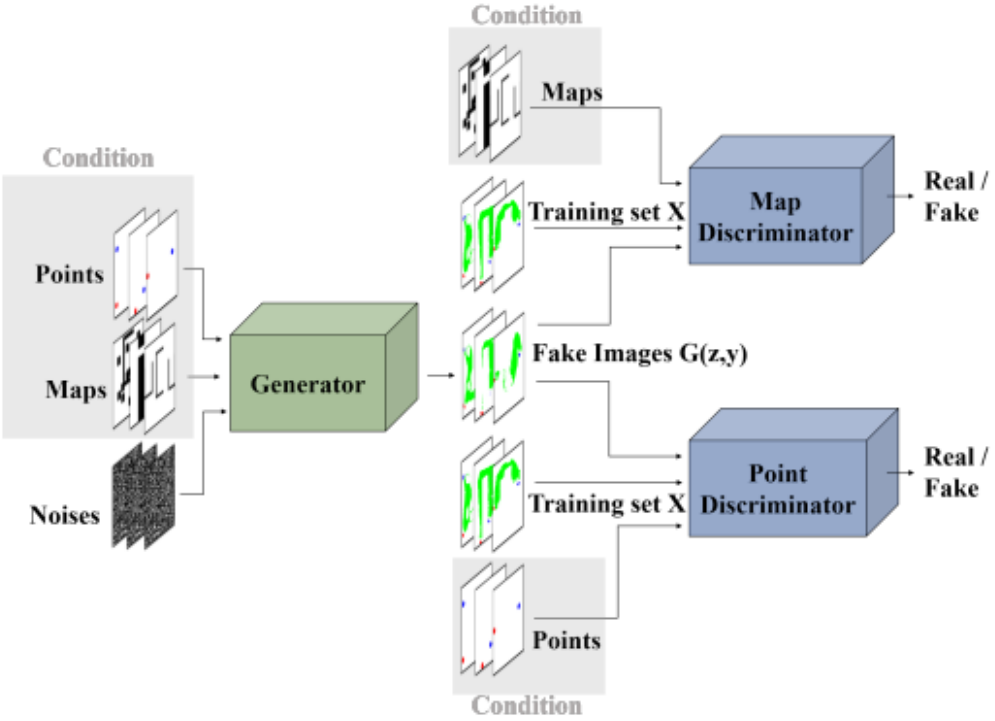
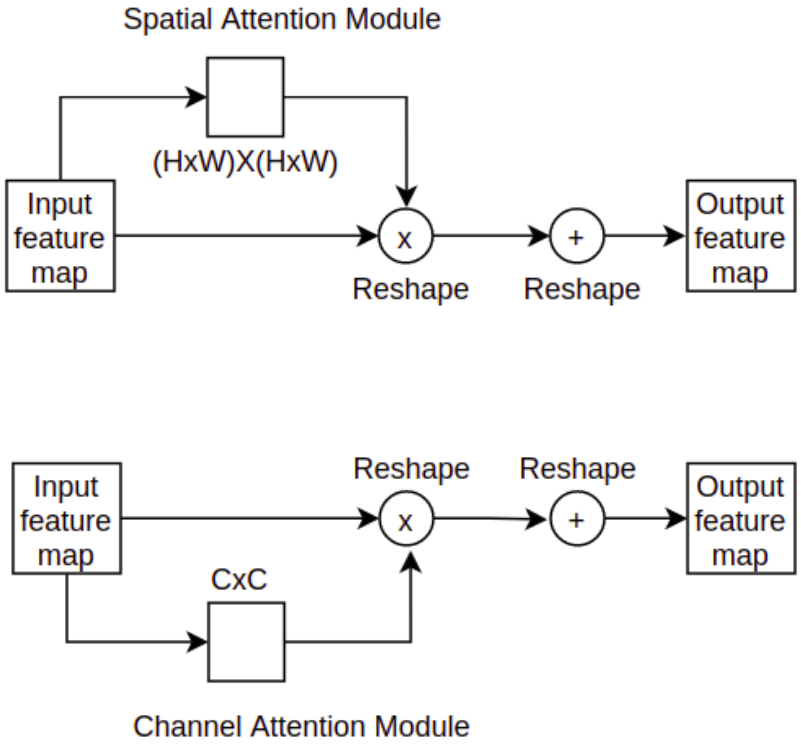


Fig. 3. The overall structure of GAN model for promising region generation.

Network Architecture



Loss Functions

Cross Entropy Loss

$$\mathcal{L}_{CE} = -\sum_{i,j} y_{i,j} \log \hat{y}_{i,j} + (1 - y_{i,j}) \log (1 - \hat{y}_{i,j}) \quad (18)$$

Binary Cross Entropy Loss X Direction

$$L_{bce}^{x}\left(\mathcal{P}^{x}, \hat{\mathcal{P}}^{x}\right) = \sum_{i=1}^{H}\sum_{j=1}^{W} \mathcal{P}_{i,j}^{x} \log\left(\hat{\mathcal{P}}_{i,j}^{x}\right) + \left(1 - \mathcal{P}_{i,j}^{x}\right) \log\left(1 - \hat{\mathcal{P}}_{i,j}^{x}\right) \quad (19)$$

Binary Cross Entropy Loss X+Y Direction

$$L_{bce}^{xy}(\mathcal{P}, \hat{\mathcal{P}}) = L_{bce}^{x}\left(\mathcal{P}^{x}, \hat{\mathcal{P}}^{x}\right) + L_{bce}^{y}\left(\mathcal{P}^{y}, \hat{\mathcal{P}}^{y}\right) \quad (20)$$

Dice Loss

$$L_{dice}^{xy}(\mathcal{P}, \hat{\mathcal{P}}) = 1 - \frac{2\left|\mathcal{P}^{x} \cap \hat{\mathcal{P}}^{x}\right| + 2\left|\mathcal{P}^{y} \cap \hat{\mathcal{P}}^{y}\right|}{|\mathcal{P}^{x2}| + \left|\hat{\mathcal{P}}^{x2}\right| + |\mathcal{P}^{y2}| + \left|\hat{\mathcal{P}}^{y2}\right|} \quad (21)$$

Total Loss

$$L(\mathcal{P}, \hat{\mathcal{P}}) = L_{bce}^{xy}(\mathcal{P}, \hat{\mathcal{P}}) + L_{dice}^{xy}(\mathcal{P}, \hat{\mathcal{P}}) \quad (22)$$

Map Discriminator Loss

$$\mathcal{L}_{D_{map}} = L(\mathcal{P}, \hat{\mathcal{P}}) + \mathbb{E}\left[\log D_{\text{map}}(s, m)\right] + \mathbb{E}_{[}\log\left(1 - D_{\text{map}}(G(z, m, p), m)\right)\Big] \quad (23)$$

Point Discriminator Loss

$$\mathcal{L}_{D_{point}} = L(\mathcal{P}, \hat{\mathcal{P}}) + \mathbb{E}\left[\log D_{\text{point}}(s, p)\right] + \mathbb{E}_{[}\log\left(1 - D_{\text{point}}(G(z, m, p), p)\right)\Big] \quad (24)$$

Pixel Wise MSE Loss

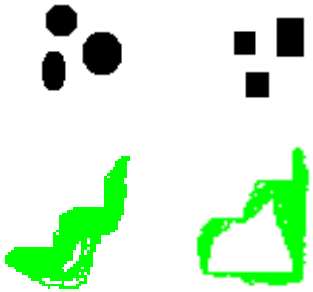
$$\mathcal{L}_{mse}(X, Y) = \ell_{mse}(G(X), Y) = \|G(X) - Y\|^{2} \quad (25)$$

Generator Loss

$$\mathcal{L}_{G} = \alpha_1 \mathbb{E}\left[\log D_{\text{map}}(G(z, m, p), m)\right] + \alpha_2 \mathbb{E}\left[\log D_{\text{point}}(G(z, m, p), p)\right] + \mathcal{L}_{mse}(X, Y) \quad (26)$$

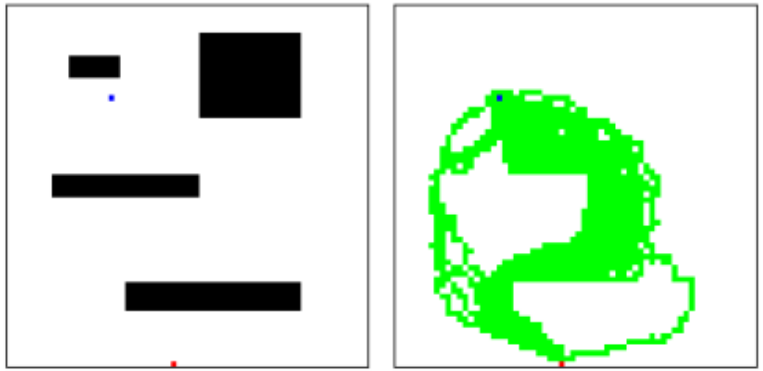
GAN Based Learning

1. Train on 8000 Maps with respective start and goal by GAN based learning. And test on 2000 images to see how well the model has learn



2. Now when given a new image, The model outputs an image like below.

3. An example to show how ROI look on a map



Path Planning

1. The algorithm inputs the map with generated ROI and start planning path.
2. The map is converted into binary where black represents obstacle
3. The algorithm attempts to find the nearest vertex to the new sample and connect them and the new vertex is added to the vertex set V and the edge is added to the edge set E if no obstacles are present in the connection.

Advantages

1. Search Space reduced drastically
2. Since search space is smaller, the path generated is optimal than the one generated by only using RRT.

RRT algorithm

Algorithm 1: RRT Code

```
RRT(Map, start , goal)
    qs = [start]
    qs_parent = [0]
    while True
    {
        q_rand = random_vertex_generated()
        near_index ,q_near = nearest_vertex_check(q_rand)
        q_new = new_point_generate(q_near , q_rand , index_near)
        if connection_check(q_new)
        {
            break
        }
    }
figure_generate()
```

Evaluation Metrics

1. We use Dice Score (Dice) and Intersection over union (IoU) as image quality metrics to evaluate the performance of our network.

1. We use Fréchet Inception Distance (FID) as image quality generation metric to evaluate the quality of the generated promising regions

1. We use the time cost which represents the time taken for the algorithm to find a solution and the number of nodes which denotes the number of nodes explored on the way to reach the solution as the path planning metrics.

Hyperparameters

Table 2: Hyperparameters

Parameter	Value
Batch Size	8
Epochs	20
Generator Learning Rate	0.0001
Map Discriminator Learning Rate	0.00005
Point Discriminator Learning Rate	0.00005
Optimzer	Adam ($\beta_1 = 0.5$, $\beta_2 = 0.999$)

Loss Function Parameters

Table 3. Loss Function Parameters

Parameter	Value
α in Dice loss	10
The smooth parameter in Dice loss	1
α in IOU loss	10
The smooth parameter in IOU loss	1
α in Pixel Wise MSE loss	20
α in Generator loss	100
α in CBAM Generator loss	1
β in CBAM Generator loss	1
α in Adaptive CBAM Generator loss	3

RRT Algorithm Parameters

Table 4. RRT Algorithm Parameters

Parameter	Value
Step Length	0.2
Maximum Iterations	5000
Resolution of path	1

Comparison of Results

Figure 5: Illustration of the comparison of results obtained. The first column shows the obstacle map with the white region as the free space and the black region as the obstacle space. The second column shows the promising region of the ground truth maps which goes into the training. The third column shows the promising region generated using the Self-Attention Generative Adversarial Network by (Zhang et al., 2021). The fourth column shows the promising region generated using our CBAGAN. The red and the black nodes in all the maps denote the start and goal nodes respectively.

Comparison of Results

Table 5: Comparison of our results vs state of the art networks using IOU, Dice, FID, and the number of parameters as the evaluation metrics.

Metrics	IOU	Dice	FID	Number of Parameters
SAGAN (Zhang et al., 2021)	88.45	93.50	66.47	1.24 Million
Our Network	**89.22**	**94.16**	**62.04**	**0.88 Million**

Comparison of Results

Table 6: Comparison of our results versus RRT and SAGAN-RRT on 3 different maps using time cost and the number of nodes as the path planning evaluation metrics.

Map	Algorithm	Time Cost(s)	Number of Nodes
Map1	RRT	2.12	202
Map1	SAGAN (Zhang et al., 2021)	1.37	157
Map1	Our Network	**1.26**	**143**
Map2	RRT	3.65	351
Map2	SAGAN (Zhang et al., 2021)	2.40	246
Map2	Our Network	**2.08**	**218**
Map3	RRT	1.85	184
Map3	SAGAN (Zhang et al., 2021)	1.28	129
Map3	Our Network	**1.06**	**110**

Comparison of Results

RRT

```
C:\Users\Abhinav\Desktop\github\test\pathgan\models\rrt>python rrt1.py
Time taken to reach goal node=0.833s
Number of nodes explored=81
[[15.0, 53.0], [14, 55], [16, 53], [13, 56], [19, 55], [21, 55], [14, 50], [27, 56], [14, 55], [21, 55], [31, 55], [11,
49], [9, 45], [31, 56], [11, 43], [31, 56], [13, 41], [33, 56], [34, 56], [34, 56], [11, 57], [21, 55], [12, 40], [7, 46
], [34, 57], [14, 41], [14, 40], [10, 49], [8, 57], [16, 41], [18, 56], [36, 55], [38, 55], [40, 55], [27, 57], [14, 52]
, [14, 49], [6, 57], [11, 58], [11, 42], [27, 58], [31, 56], [30, 56], [27, 58], [6, 46], [40, 56], [6, 45], [21, 55], [
11, 43], [26, 58], [42, 57], [5, 58], [45, 57], [33, 56], [47, 55], [25, 59], [4, 57], [49, 56], [16, 40], [14, 48], [24
, 59], [17, 41], [35, 55], [50, 57], [49, 58], [3, 57], [4, 43], [13, 52], [14, 47], [11, 43], [30, 56], [52, 57], [17,
41], [5, 46], [11, 39], [53, 52], [53, 51], [45, 57], [17, 56], [54, 52], [7, 46], [51, 44], [24.0, 18.0]]
```

CBAGAN-RRT

```
C:\Users\Abhinav\Desktop\github\test\pathgan\models\rrt>python rrt2.py
Time taken to reach goal node=0.189s
Number of nodes explored=18
[[15.0, 53.0], [15, 52], [15, 53], [21, 49], [15, 52], [24, 46], [26, 46], [27, 46], [30, 42], [32, 41], [33, 37], [35,
37], [33, 37], [36, 38], [37, 37], [32, 37], [33, 42], [20, 48], [38, 34], [24.0, 18.0]]
```

Comparison of Results

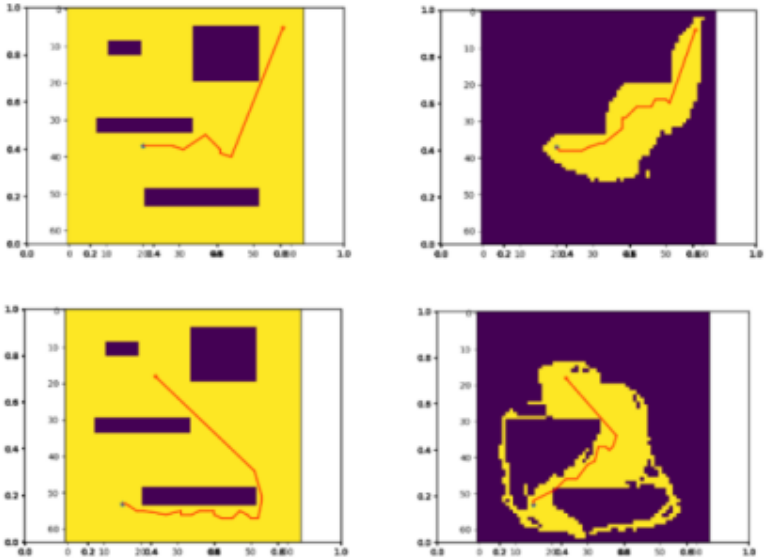


Figure 7: Illustration of our results using CBAGAN-RRT versus the original RRT in terms of the path generated. The left image depicts the path generated on the original obstacle map where the obstacles are shown in the dark color and the light color denotes the free space. The right image depicts the path generated using only the region of interest generated from our GAN model where the lighter color denotes the area inside the region of interest while the darker color denotes the region outside the region of interest. Only the region of interest is used for finding the feasible path and because the area is much smaller than that in the other image, we are able to improve the speed of convergence of the algorithm. The start and the goal nodes are shown in color blue and red respectively while the red line denotes the path taken.

Ablation Study

Table 7: Comparison of our results with various attention modules using IOU, Dice, and FID as the image quality metrics and the time cost and number of nodes as the path planning metrics.

Attention Module	IOU	Dice	FID	Time Cost	Number of Nodes
CBAM (Woo et al., 2018)	**85.22**	**95.86**	**65.47**	**1.08**	**127**
DAModule (Fu et al., 2019)	84.54	94.29	68.45	1.23	146
ECAAttention (Wang et al., 2020)	84.81	94.05	72.41	1.34	150
ShuffleAttention (Zhang and Yang, 2021)	85.06	95.42	70.56	1.16	133
TripletAttention (Misra et al., 2021)	83.60	92.87	77.15	1.40	164
SKAttention (Li et al., 2019b)	84.58	93.70	69.95	1.28	150
SpatialGroupEnhance (Li et al., 2019a)	83.82	94.99	72.63	1.42	142
SEAttention (Hu et al., 2018)	83.19	93.15	71.75	1.27	139
CoTAttention (Li et al., 2022)	84.14	94.06	69.25	1.36	145

Ablation Study

Table 8: Comparison of our results with a combination of spatial and channel attention modules using IOU, Dice, and FID as the image quality evaluation metrics and time cost and the number of nodes as the path planning metrics.

Spatial Attention	Channel Attention	IOU	Dice	FID	Time Cost	Number of Nodes
✓	✓	**85.22**	**95.86**	**65.47**	**1.08**	**127**
✓	×	83.56	92.69	75.74	2.02	193
×	✓	84.73	94.06	66.49	1.75	166
×	×	83.08	92.10	82.53	2.56	235

Conclusions

1. In this paper, we proposed a novel image based approach using a convolutional block attention based Generative Adversarial Network network CBAGAN-RRT to design the heuristics, find better optimal solutions, and improve the convergence thus reducing the computational requirements for sampling based path planning algorithms.

1. We use the predicted probability distribution of paths generated from our model to guide the sampling process of the RRT algorithm.

1. We demonstrate that our approach performs better than the state of the art approach both in terms of the quality of feasible paths generated from an image perspective using IOU, Dice, and FID as the evaluation metrics as well as metrics like time cost and number of nodes, from path planning perspective.

1. Our algorithm CBAGAN-RRT differs only in the sampling strategy from the RRT algorithm hence it can be easily integrated with other sampling based path planning algorithm.

Challenges Faced

1. We had to often transition between coding on the cloud versus coding on our personal laptop. We found it difficult to manage the different versions of code, data, and the weights together.

1. We were not able to run some of the script from the open source github repository that we were working on top of like the evaluation metrics and the path planning component so we had to write them ourselves.

1. We spent a lot of time reading research paper and brainstorming together with the aim to come up with a novel solution. We found it difficult in the beginning to understand some of the details mentioned in the papers.

Contributions

Abhinav Sagar - Came up with the idea, did literature review, trained and tested the model, made the presentation, wrote the report.

Sai Teja Gilukara - Integrated path planning algorithm on top of the GAN heuristic model, conducted experiments and got the results, made the presentation, wrote the report.

Future Work

1. A future idea to build upon this work could be to integrate the algorithm in a ROS-Gazebo environment while simulating and benchmarking the results with the RRT algorithm in 3 dimensions.

1. Another future idea could be to solve the problem by mapping it only as a image segmentation and not image generation that is segmenting the feasible region using a state of the art image segmentation algorithm and feeding that as heuristics for the path planning to improve the results.

1. Finally, our algorithm can be integrated with other sampling based path planning algorithms to study, test, and validate the effectiveness of our approach.

THANK YOU

Link to video of presentation: link