2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

Tube-GAN: A Novel Virtual Tube Generation Method for Unmanned Aerial Swarms Based on Generative Adversarial Network

Shixun Zhai, Kaige Zhang, Bo Nan, Yanwen Sun, and Qianyi Fu

***Abstract*—Virtual tube is a two-dimensional or three-dimensional strip or tubular area similar to RSFC (Relative Safe Flight Corridor), which can provide smooth, feasible, and safe space for UAV swarm in environments with dense obstacles. In order to address the problem that current virtual tube planning methods are mainly based on complex heuristic algorithm with consuming time complexity, we modify the model architecture by introducing generative adversarial network (GAN), and propose a Tube-GAN model. Tube-GAN takes the key point prompt image and obstacle environment image as inputs, and outputs the image of the virtual tube, which transforms the optimization problem into an image generation problem, leveraging the performance of computational efficiency for the construction of virtual tube. The experimental results demonstrate that the proposed Tube-GAN model can quickly generate virtual tube in random environments (less than 25ms), providing a new direction for the construction of virtual tube in real-time.**

INTRODUCTION

Unmanned aerial swarm systems have attracted more and more attentions from industry and military applications because of the advantages in autonomy, collaboration, robustness and scalability. The unmanned swarms can perform more complex and diverse tasks with higher efficiency and lower cost comparing with using single robotic machine in many fields, especially under harsh environmental conditions [1],[2]. However, the rapid and collision-free traversal for unmanned aerial swarm systems in complex environments such as forests, valleys, and cities is a challenging problem that has not been well solved. Reference [3] uses a distributed architecture to generate trajectory for each UAV following some priority order, and broadcasts the generated trajectory to other UAVs as obstacle information for trajectory planning. Such methods can achieve high-speed control for each UAV, but they can easily lead to deadlock problems, especially when the number of UAV individuals is large. Reference [4] designed a position and velocity consistency control law between the leader UAV and the follower UAVs to maintain the formation of the aerial swarm where artificial potential field method was introduced. Such formation control-based method can avoid deadlocks between UAVs, but the traffic efficiency is poor. In addition, by mimicking bird swarm behaviors, Reynolds et al. [5] proposed the Boyd model, followed by a number of biomimicry swarm motion models, such as Vicsek model, Couzin model, and Cucker-Smale model. These methods achieved some success in obstacle avoidance with large-scale unmanned aerial swarms, but the traffic efficiency problem is still not resolved.

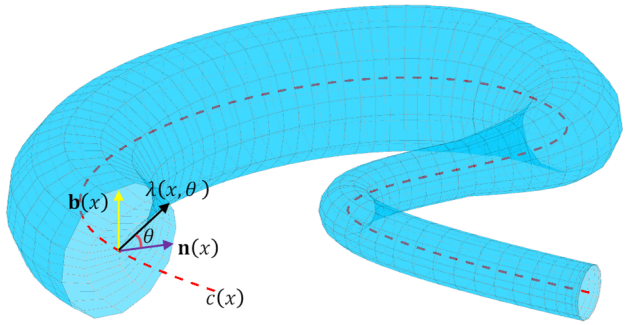


Fig. 1 Virtual tube in three-dimensional spaces

Virtual tube was proposed to address the contradiction between the increasing use of UAVs and limited airspace [6]. For vehicle motion planning, an Air Traffic Management System (ATM) is constructed to confine UAVs within planned virtual tube and to ensure that newly added UAVs could avoid collisions with other existing UAVs. Combining trajectory planning and swarm control algorithms, reference [7] proposed a swarm flight control approach with virtual tube. The virtual tube is constructed from the traditional trajectory planning prospect, and swarm control methods such as the flocking model are used to achieve collaborative control of the unmanned aerial swarms within the virtual tube. However, the construction of virtual tube is based on traditional heuristic algorithm with strict mathematical constraints, and that is time consuming and cannot meet the requirements of UAV swarm systems.

In this work, we propose a novel virtual tube planning method, named Tube-GAN, which transforms the virtual tube construction problem into an image generation problem by introducing the generative adversarial network technology. Tube-GAN takes the key point prompt image and obstacle environment image as inputs, outputs the environmental image embedded with well-planned virtual tubes, resulting in a significant performance improvement of computational efficiency, and providing a new solution for swarm trajectory planning based on virtual tube generation. The main contributions of this work are:

1) Propose a novel virtual tube generation method based on GAN, which for the first time transforms the virtual tube construction problem into an image generation problem;

2) Design, implement, and evaluate the Tube-GAN model for swarm control in the environment with dense obstacles;

3) Validate the effectiveness and efficiency of the proposed method by simulating the full process of UAV

Shixun Zhai, Kaige Zhang (corresponding author), Bo Nan and Yanwen Sun are with the North Automatic Control Institute, Taiyuan, Shanxi 030000 China (e-mail: zhaishixun@126.com; zkgusu@gmail.com).

Qianyi Fu is with School of Aeronautics and Astronautics, Zhejiang University, Hangzhou 310014 China (e-mail: scfq@leeds.ac.uk).

swarm flighting which utilizes flocking method to control the swarm individuals to fly within the generated virtual tube, providing a feasible solution of the virtual tube method.

RELATED WORK

A. Virtual Tube

The virtual tube method addresses two problems: virtual tube planning and virtual tube passing-through control. There are many studies on virtual tube, which generally consists of two steps. Firstly, an optimization-based method is used to plan a trajectory, and then the obtained trajectory is expanded to generate a virtual tube. Afterwards, some control method such as the flocking algorithm is employed to design a distributed swarm controller.

Mao and Quan [8] first mathematically modeled virtual tube and proposed the regular virtual tube. They defined and proved four criteria for regular virtual tube: the tube surface is regular, the tube shrinkage rate is low, the cross section of the tube is simply connected, and the tube has no self-intersection. Then they validated the proposed method in two-dimensional plane. In order to facilitate the free flight of unmanned aerial swarms in chaotic environments, Gao et al. [9] proposed a connected quadrangle virtual tube method, which achieved the construction of virtual tube by connecting multiple single trapezoid virtual tubes. Then, based on the gradient vector field method without local minima, a distributed group controller was designed to solve the control problem of passing through a single trapezoidal virtual tube. Recently, they generalized the definition of virtual tubes and proposed the curved virtual tube [10], and made an optimization [11].

In this work, we mainly discuss the planning problem of curved virtual tube, and then use the flocking method for flight control.

B. GAN

GAN has quickly become a hot research topic since it was proposed by Ian Goodfellow in 2014 [12]. Due to its many advantages, GAN is widely used in various fields such as image to image translation and image generation [13].

Qureshi et al. [14] designed a neural network-based motion planning network (MPnet) that encodes a given workspace from point cloud measurements and generates end-to-end collision free paths. Wang et al. [15] trained a CNN model to generate irregular sampling by using the path points obtained from the A* algorithm, and then guided the RRT* algorithm to perform higher quality sampling and thereby accelerate the convergence. Soboleva [16] performed a preliminary study on using the generator of GAN as a path finder. Fu and Lee [17] proposed a Progressive Route Planning GAN (ProgRPGAN), which generates paths by first planning routes in a low-resolution grid map and then refining them in a high-resolution grid map. Zhang et al. [18] used GAN to generate a promising region and performed point sampling within the region to accelerate the convergence RRT* algorithm.

Inspired by the above methods, we design the Tube-GAN model to generate virtual tube directly, accelerate the construction of virtual tube in random environments, and reduce computational time.

METHODS

In this section, we first introduce the formulation of virtual tube and then describe the proposed Tube-GAN model in detail, which takes the key point prompt images and obstacle environment images as inputs, the model output is the image contains the generated virtual tube.

A. Virtual Tube Formulation

The defined virtual tube is shown in Fig 1. For two-dimensional or three-dimensional space, the optimal trajectory connecting the starting point x_{start} and the target point x_{target} is defined as $c(x) \in \mathbb{R}^n, (n = 2 \text{ or } 3)$, where $x \in [x_{start}, x_{target}]$ represents a point on the trajectory, $\mathbf{n}(x) = \frac{\|\dot{c}(x)\|^2 \ddot{c}(x)}{\|\ddot{c}(x) \times \dot{c}(x)\|} \in \mathbb{R}^n$ is the principal normal vector at point x, and $\mathbf{b}(x) = \frac{\dot{c}(x)}{|\dot{c}(x)|} \times \mathbf{n}(x) \in \mathbb{R}^n$ is the binormal vector at point x, $\lambda(x,\theta) \in \mathbb{R}$ is the radius along the direction of the angle θ with the normal vector at point x, then the subset $\mathcal{T} \subset \mathbb{R}^n$ can be used to represent the virtual tube

$$\mathcal{T}(x,\theta,\rho) = c(x) + \rho\lambda(x,\theta)[\mathbf{n}(x)\cos\theta + \mathbf{b}(x)\sin\theta] \quad (1)$$

where $\rho \in [\rho_{min}(x), 1]$ is a constant coefficient. When $\mathcal{T} \subset \mathbb{R}^2$, $\mathbf{b}(x) = 0, \theta = 0$ or π, the Eq. 1 can be refined as:

$$\mathcal{T}(x,\theta,\rho) = c(x) + \rho\lambda(x,\theta)\mathbf{n}(x)\cos\theta \quad (2)$$

Virtual tube generation is performed on the optimal trajectory planned in a two-dimensional plane, and the radius of the virtual tube can be expressed as:

$$\frac{1}{\lambda(x,\theta)} = \begin{cases} \sum_{i=0}^{2s-1} p_{1,i}(\theta)x^i & x_0 \le x < x_1 \\ \sum_{i=0}^{2s-1} p_{2,i}(\theta)x^i & x_1 \le x < x_2 \\ \vdots & \vdots \\ \sum_{i=0}^{2s-1} p_{m,i}(\theta)x^i & x_{m-1} \le x < x_m \end{cases} \quad (3)$$

Among them, $p_{m,i}(\theta) = \sum_{k=0}^{2s-1} b_{k,m}\theta^k$. We expect the area of the virtual tube to be as large as possible, that is, the larger the radius, the better, while ensuring that the surface of the virtual tube is sufficiently smooth. Therefore, it is necessary to limit the second derivative of the radius. For practical considerations, although a larger radius is better, an upper limit needs to be set λ_{max}, which is determined by the size, detection radius, and range of movement of each drone (considering instability). After obtaining the optimal trajectory, it is necessary to calculate the distance between each small segment and the nearest obstacle along the trajectory $\lambda_{min,i}$, parameter $x(\lambda_{min,i})$ and $\theta(\lambda_{min,i})$. Comparing $\lambda_{min,i}$ and λ_{max}, if $\lambda_{min,i} > \lambda_{max}$, the radius of this trajectory segment is define as λ_{max}, otherwise it is defined as $\lambda_{min,i}$. According to [8], the cost function is designed as follows

$$\min_{b_{k,m}} \iint_{\mathcal{D}_x \times \mathcal{D}_\theta} \left\| \frac{1}{\lambda(x,\theta)} \right\|^2 + \left\| \frac{\partial^2 \lambda}{\partial x^2} \right\|^2 + \left\| \frac{\partial^2 \lambda}{\partial \theta^2} \right\|^2 + \left\| \frac{\partial^2 \lambda}{\partial x \partial \theta} \right\|^2 \mathrm{d}x\mathrm{d}\theta \quad (4)$$

The work mainly studies the problem of virtual tube construction in a two-dimensional environment. We define an RGB image with $n \times m$ pixels as set I_{RGB}, and the pixels are represented as Eq. 5.

$$P(x_{RGB}, y_{RGB}) \in I_{RGB}, (x_{RGB} \in [1,n], y_{RGB} \in [1,m]) \quad (5)$$

Then Eqs. 2-4 can be transformed from mathematical constraints to pixel constraints. On this basis, we define the RGB virtual tube model as

$$VT = \{P(x_{RGB}, y_{RGB}) \in \mathfrak{T}, x_{RGB} = 1,2,\ldots n, y_{RGB} = 1,2,\ldots m\} \quad (6)$$

Where VT is the connected area that connects the x_{start} and x_{target}, constrained by the two boundaries of the virtual tube. Let O represents the set of obstacle areas and F represents the set of free spaces, then the Eq. 5 can be refined as Eq. 7.

$$P(x_{RGB}, y_{RGB}) = \begin{cases} (0,0,255) & (x_{RGB}, y_{RGB}) \in VT \\ (0,0,0) & (x_{RGB}, y_{RGB}) \in O \\ (255,255,255) & (x_{RGB}, y_{RGB}) \in F \end{cases} \quad (7)$$

Now, we have transformed the virtual tube planning problem into an image generation problem.

B. Model Structure

The overall framework of Tube-GAN is shown in Figs. 2-4, which mainly includes two parts: generator G and discriminator D. Generator G is set up following the conditional GAN [19], it takes conditions y and noise image z as input, and outputs image $G(z \mid y)$; z follows the normal distribution $P_z(z)$. Discriminator D inputs image $G(z \mid y)$ and image x alternatively, and outputs true or false; x follows the distribution of the ground truth images, noted $P_{truth}(x)$. There is an adversarial learning between generator G and discriminator D, where G generates images that are as close to the training data as possible, so that discriminator D cannot distinguish whether the image comes from the dataset or is generated by generator G. The objective function can be defined as

$$\min_G \max_D (\mathbb{E}_{x \sim P_{truth}(x)}[\log D(x \mid y)] + \mathbb{E}_{z \sim P_z(z)}[\log(1 - D(G(z \mid y) \mid y))]) \quad (8)$$

In this work, we use the generator G to generate the virtual tube images with the same distribution as the ground truth, by inputting key point prompt images which contain starting and target point, as well as the images with obstacle environment. Assuming the ground truth images $i \in I$, and the noise z is a two-dimensional image sampled from normal distribution. The key point prompt images $p \in \mathcal{P}$, the obstacle environment images $e \in \mathcal{E}$; the input images p and e are two conditions for G. To ensure that the virtual tube images obtained by generator G is associated with the starting and target points, we use two discriminators D_{point} and D_{env} to determine whether the generated virtual tube matches the starting point, target point and obstacle environment.

Fig. 2 shows the detailed architecture of the generator in the Tube-GAN. In generator G, input a noise image z of 256×256×1, an obstacle environment image e of 256×256×3, a key point prompt image p of 256×256×3, and output a virtual tube image of 256×256×3.

In our model, the generator follows the U-shaped structure as shown in Fig. 2. The input images of p, e and z are firstly merged by a convolutional operation into 64-channel. We define block1 to block4 as the down-sampling processes. Each block contains two ResNet cells and a Convolution-Batch Normalization-ReLU layer. The features from block1 are stored for subsequent operations. We further define block5 to block7 as the up-sampling processes. Each block contains two Resnet cells and a Deconvolution-Batch Normalization-ReLU layer. Finally, we use block8 to fuse the contextual features and improve the quality of generated virtual tube image.

The two discriminators have similar structures except for the number of the layers, as demonstrated in Fig. 3 and Fig. 4. The discriminators are mainly constructed by Convolution with Batch-Normalization and ReLU blocks. Inspired by the self-attention mechanism [18], we add self-attention blocks at the first layer and the third layer of the discriminators to better include global information.

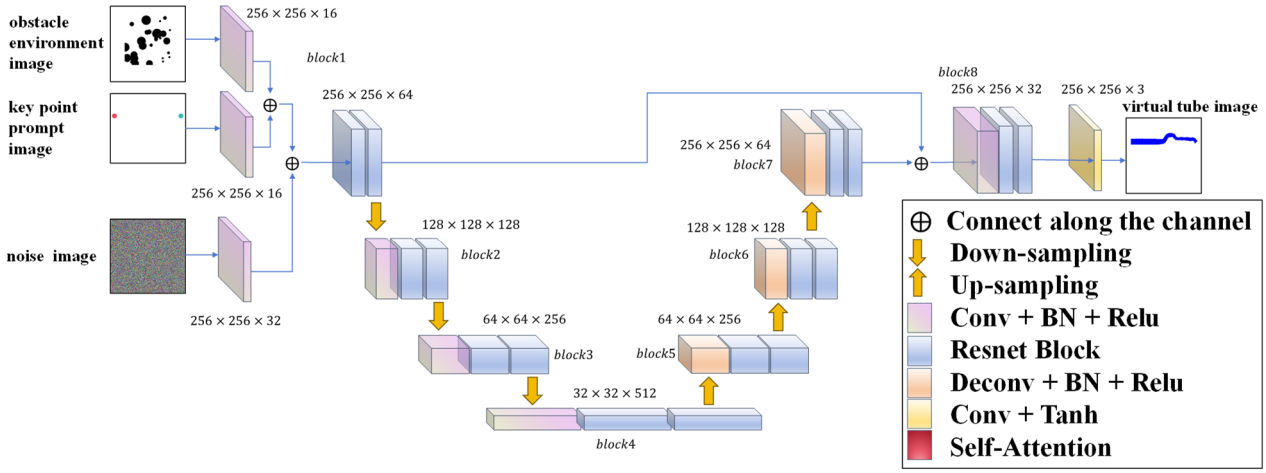


Fig. 2 Detailed structure of the generator G

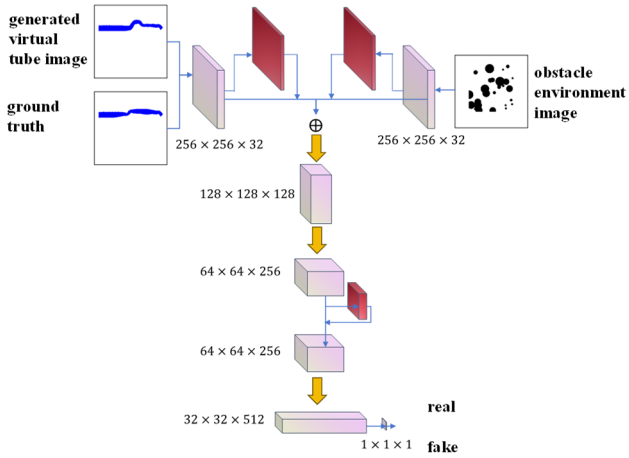


Fig. 3 The detailed structure of discriminator D_{env}

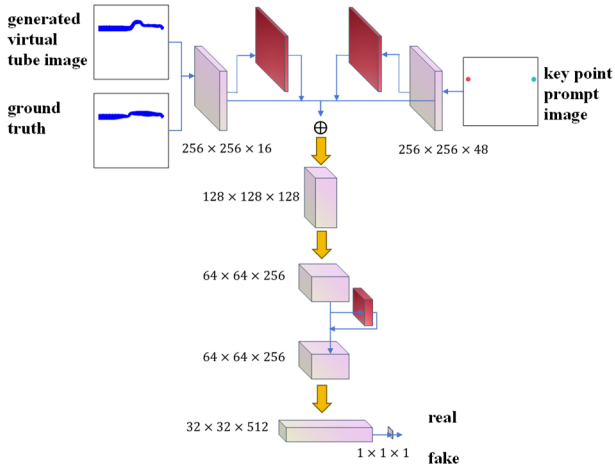


Fig. 4 The detailed structure of discriminator D_{point}

C. Loss Function

During the training process, we first fix the parameters of generator G to train two discriminators D_{point} and D_{env}, and the loss functions are defined as follows:

$$L_{D_{point}} = \mathbb{E}[\log D_{point}(i,p)] + \mathbb{E}[\log(1 - D_{point}(G(e,p,z),p))] \quad (9)$$

$$L_{D_{env}} = \mathbb{E}[\log D_{env}(i,e)] + \mathbb{E}[\log(1 - D_{env}(G(e,p,z),e))] \quad (10)$$

where $i, e, p \sim P_{I,\mathcal{E},\mathcal{P}}(i,e,p)$.

Then, we fix the parameters of D to train G. Here, the loss function of generator G are composed of two parts, L_{prompt} and L_{tube}:

$$L_G = \alpha_{prompt} L_{prompt} + \alpha_{tube} L_{tube} \quad (11)$$

where

$$L_{prompt} = \alpha_{point} L_{point} + \alpha_{env} L_{env} \quad (12)$$

and

$$L_{point} = \mathbb{E}[\log(D_{point}(G(e,p,z),p))] \quad (13)$$

$$L_{env} = \mathbb{E}[\log(D_{env}(G(e,p,z),e))] \quad (14)$$

Due to the Tube-GAN mainly focus on the generation of virtual tube, we assign a higher initial weight to the latter term L_{env} and follow the method [18] to adjust the loss function by designing hyperparameter h, as shown in Eq. 15. In our model, we set h to 8.

$$L_{prompt} = \frac{L_{D_{env}}}{hL_{D_{point}} + L_{D_{env}}} L_{point} + \frac{hL_{D_{point}}}{hL_{D_{point}} + L_{D_{env}}} L_{env} \quad (15)$$

We also introduce a virtual tube loss L_{tube}, as in Eq. 16. L_{tube} consists of two parts: pixel loss L_{pixel}, and the collision loss $L_{collision}$.

$$L_{tube} = L_{pixel} + L_{collision} \quad (16)$$

where

$$L_{pixel} = \frac{(i - G(z))^2}{N} \quad (17)$$

where i is the ground truth image, $N = 256 \times 256$ is the total number of pixels in each image, and $G(z)$ is the generated image.

While the obstacle environment image is a binary image, $L_{collision}$ is achieved by dot multiplying the generated virtual tube image with the obstacle environment image e. The collision loss $L_{collision}$ is

$$L_{collision} = G(z) \bullet e \quad (18)$$

L_{pixel} and $L_{collision}$ are made to help the model generate the collision-free tube image. In the experiments, α_{prompt} is set to 0.43, and α_{tube} is set to 0.57, which are determined by grid search.

EXPERIMENTS

A. Dataset and Training Details

We generate environment maps and regular virtual tube with smooth boundaries for obstacle avoidance based on MATLAB [8]. The generated images in the dataset are demonstrated in Fig. 5. Each set of data includes environment image, key point prompt image, and corresponding regular virtual tube image. On each environment map, we adopt a randomly generated method, where the white areas are free space and the black areas are obstacles space. The starting and target points are represented by red and green dots respectively. In order to enhance the robustness of the generator, the X - axis coordinates of the starting and target

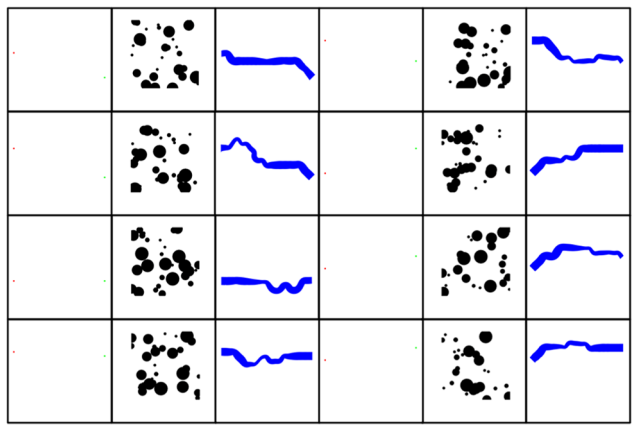
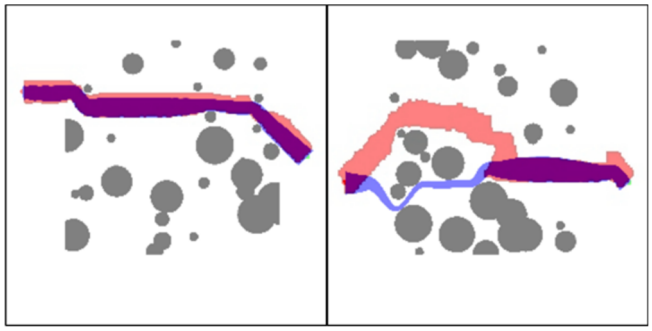
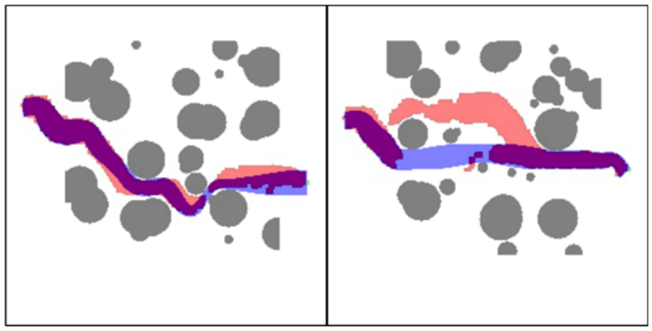


Fig. 5 Dataset images. The red dot represents the starting point, the green dot represents the target point, the black area represents obstacles, and the blue area represents virtual tube.



(a) Connected



(b) Disconnected

Fig. 6 Connected and disconnected images. The red area represents the virtual tube generated by Tube-GAN, the blue area represents the ground truth, and the purple area represents the overlapping part.

points are randomly generated on the basis of fixed Y-axis coordinates. At the same time, in order to prevent the Tube-GAN model from ignoring this information due to the small starting and target points, the discriminator inputs both the environment image and the key point prompt image. And the blue area represents the regular virtual tube generated based on the key point prompt image and environment image.

The experimental dataset contains 23512 sets of data, 12345 sets of data were randomly selected as the training set and 3344 sets as the testing set. In order to ensure that the boundaries of the virtual tube in the training set are smooth enough and the training proceeds normally, the image size is set to 256×256 pixels. For each image, we change the starting and target points of each map and randomly generate obstacles. We conducted 50 rounds of training using Tesla V100 SMX2 16GB, set the learning rate of the generator to 0.0001, the learning rate of the discriminator to 0.00005, and use Adam optimizer with parameters $\beta = (0.5, 0.999)$.

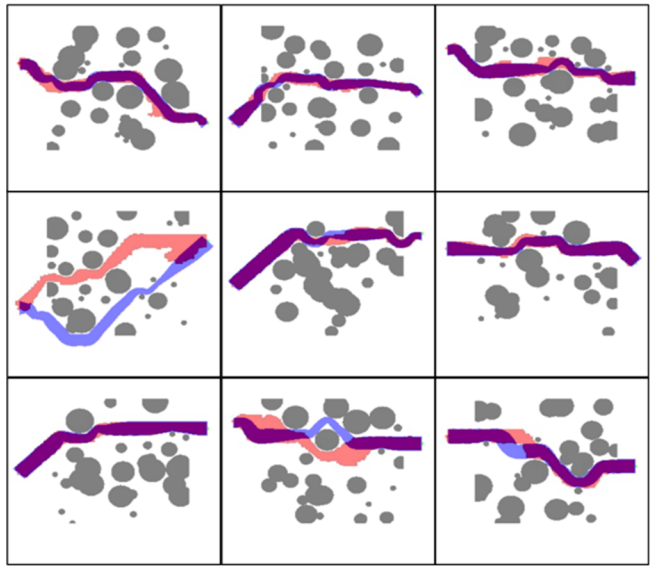


Fig. 7 Generated virtual tube and ground truth. It is obvious that the generated virtual tube overlaps a lot with the ground truth. Although partially generated virtual tube differ significantly from ground, they all meet the requirements of connectivity and collision avoidance.

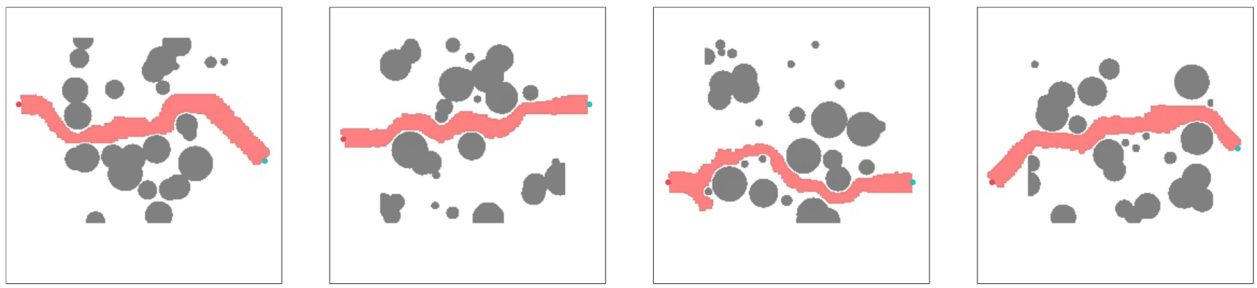
B. Evaluation

Tube-GAN is evaluated from three aspects: connectivity, obstacle avoidance ability and success rate, with the expectation of generating virtual tube under different conditions that are sufficiently similar in shape and features to the ground truth.

1) Connectivity

For the virtual tube generation task, the expected virtual tube is a continuous area image without interruption, connecting the starting and ending points, and avoiding obstacles as much as possible. Therefore, traditional image similarity evaluation indicators such as SSIM, L1 loss, and histogram method were not selected for this image generation task.

RRT* algorithm is used to test the connectivity of the generated virtual tube. Use the virtual tube area as the free space and sampling space, and the area outside the virtual tube as the obstacle space, and then use the RRT* algorithm to find the path connecting the starting point to the target point. If a path can be found, it is considered that the generated virtual tube meets the connectivity requirements, as demonstrated in Fig. 6.



(a) Scene1 (b) Scene2 (c) Scene3 (d) Scene4

Fig. 8 Four experimental scenes

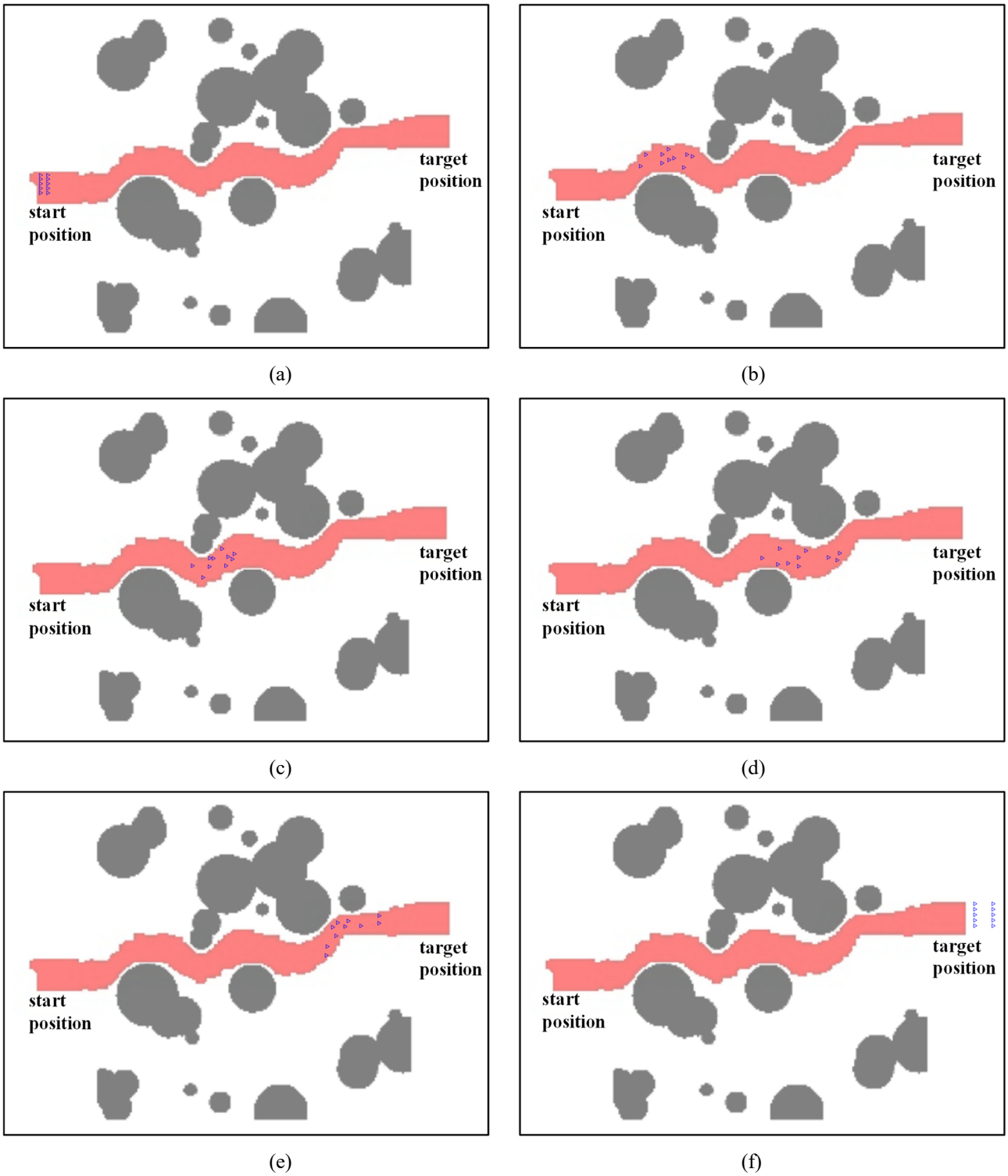


Fig. 9 The simulation of swarm flying. The red area represents the virtual tube generated by Tube-GAN, the black area represents messy obstacles, and the ten blue triangles represent the UAVs. Figure (a) indicates the initial state of the UAV swarm, all located at the starting position. Figure (f) indicates the final state of the UAV swarm, all located at the target position. Figures (a) to (f) represent the flight process of the UAV swarm. Since the two boundaries of the generated virtual tube are not smooth enough, to prevent deadlocks, we set up virtual drones and a safe radius at the boundaries. And the UAV swarm will try to stay away from the rough boundary as much as possible to avoid getting stuck in grooves and not being able to reach the target position. The experimental results indicate that our generated virtual tube is practical and can meet the requirements of collision free collaborative flight of UAV swarm, with a probability of reaching the target position of 100%

2) Obstacle Avoidance Ability

Compared with traditional heuristic methods with strict mathematical constraints, image generation-based method may generate the tube overlapping with obstacles that will lead to collision which is not expected. Therefore, Obstacle Avoidance Ability (OAA) is an important criterion to achieve collision free flight of UAV swarm in obstacle environments.

We use the method similar to Intersection over Union (IoU) to define OAA. The obstacle avoidance ability of each virtual tube is represented as follows:

$$\mathrm{OAA} = \frac{L_{collision}}{Aera(G(z))} \tag{19}$$

where $L_{collision}$ is represented as Eq. 18, $Aera(G(z))$ represents the area of the virtual tube in image $G(z)$ generated by Tube-GAN.

3) Success Rate

The success rate is an important indicator to ensure that the proposed Tube-GAN model can generate a high-quality virtual tube that satisfies the collaborative flight of unmanned aerial swarms. When the generated virtual tube satisfies both connectivity and complete obstacle avoidance, we think that the virtual tube is generated successfully.

C. Experiment Results

1) Connectivity

The generated virtual tube images and ground truth are shown in Fig. 7. After testing with the RRT* algorithm, the proposed GAN based virtual tube generation method has a connectivity rate of 92.5% and can generate virtual tube images that meet the requirements in a random environment.

2) Obstacle Avoidance Ability

As shown in Fig. 7, our model exhibits excellent performance in obstacle avoidance ability. The experimental result value $\mathrm{OAA}_{average} = 0$, with 100% of virtual tube completely avoiding obstacles. Moreover, due to the boundary safety distance set in the flocking control method, even if the virtual tube slightly overlaps with the obstacles, it will not affect the collaborative flight of the UAV swarm. The results fully demonstrate that the proposed Tube-GAN solution can effectively complete the virtual tube generation task, and the generated tube can completely avoid obstacles in the map.

3) Success Rate

We choose the virtual tube that satisfies the two conditions, connectivity requirement and $\mathrm{OAA} = 0$, as the successfully generated virtual tube. After verification, the proposed Tube-GAN model has a success rate of 92.5% in generating virtual tube in random environments, demonstrating good adaptability and robustness.

4) Time Cost

In order to verify the superiority of the proposed method in terms of computational efficiency, the proposed scheme is compared with the state-of-the-art method, and the experiments are carried out in the same configuration. We used [8] and the proposed method to generate virtual tube for 100 sets of obstacle environment maps, and key point prompt images using the Tesla V100. The experimental results are shown in Table I.

The state-of-the-art method requires two steps to complete the whole virtual tube generation task, in which the virtual tube planning process takes a lot of time, up to 863.5ms. While the Tube-GAN proposed in this paper only takes 25ms in the virtual tube planning process, and there is no need for generate curve planning. The state-of-the-art approach takes total 34.6 times longer than the proposed Tube-GAN solution. Comparing to the traditional method, Tube-GAN improves the computational efficiency and shortens the average generation time by tens of times.

TABLE I TIME COSTS COMPARISON BETWEEN TUBE-GAN AND TRADITIONAL METHOD

Method	**Average Time Cost**		
	Generator curve planning	***Virtual tube planning***	***Total***
[8]	1.5ms	863.5ms	865ms
Proposed		**25ms**	**25ms**

TABLE II PERFORMANCE OF UAV SWARM FLIGHT IN DIFFERENT SCENARIOS

Experimental scene	**Evaluating Indicator**			
	Average velocity	***Average acceleration***	***Average arrival time***	***Arrival rate***
Scene1	30.37m/s	2.35m/s^2	51.5s	100%
Scene2	23.15m/s	2.28m/s^2	60.0s	100%
Scene3	29.78m/s	2.62m/s^2	57.0s	100%
Scene4	24.58m/s	2.57m/s^2	64.3s	100%

D. Simulation For Swarm Flying

In order to verify the practicality of the virtual tube generated by the proposed Tube-GAN model, we set up 10 drones to fly in the generated virtual tube with 1280m×1280m environment using flocking algorithm. Since there are no obstacles inside the virtual tube, the drones can reach the endpoint at a faster speed. We conducted multiple experiments in different obstacle environments and starting and target points. The virtual tube and obstacle environment are shown in Fig. 8. We demonstrate the typical experimental processes as shown in Fig. 9. Table II shows the experimental results of UAV swarm flying in virtual tube in different scenarios. The results demonstrate that Tube-GAN can quickly generate high-quality virtual tube that enables the collision-free flight of UAV swarm in obstacle environments.

CONCLUSIONS AND FUTURE WORK

This work proposes a novel virtual tube generation method, Tube-GAN. The proposed Tube-GAN transforms the virtual tube construction problem from an originally complex and consuming heuristic search problem with strict mathematical constraints to a simple and straightforward image generation problem, which improves the computational efficiency significantly and provides a solution for the rapid construction of virtual tube. The quality of the generated virtual tube is evaluated concerning connectivity, robustness, and obstacle avoidance ability, and the effectiveness and efficiency of the method was verified by the flight simulation experiments. The experimental results demonstrate that Tube-GAN model is with high success rate and strong adaptability for virtual tube

generation, and can quickly generate virtual tube suitable for collaborative flight of UAV swarm system.

We have explored and verified the Tube-GAN model in two-dimensional spaces, and the method could be extended to three-dimensional spaces by employing three-view drawing and vector decomposition methodology.

REFERENCES

[1] M. Tortonesi, C. Stefanelli, E. Benvegnu, K. Ford, N. Suri and M. Linderman, "Multiple-UAV coordination and communications in tactical edge networks," in *IEEE Communications Magazine*, vol. 50, no. 10, pp. 48-55, 2012.

[2] J. Yang, J. Qian, and H. Gao, "Forest wildfire monitoring and communication UAV system based on particle swarm optimization," in *Journal of Physics: Conference Series*, vol. 1982, no. 1, pp. 012068, 2021.

[3] X. Zhou, J. Zhu, H. Zhou, C. Xu and F. Gao, "Ego-swarm: A fully autonomous and decentralized quadrotor swarm system in cluttered environments," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 4101-4107.

[4] X. Fu, J. Pan, H. Wang, and X. Gao, "A formation maintenance and reconstruction method of UAV swarm based on distributed control," in *Aerospace Science and Technology*, vol. 104, pp. 105981, 2020.

[5] F. Cucker and S. Smale, "Emergent Behavior in Flocks," in *IEEE Transactions on Automatic Control*, vol. 52, no. 5, pp. 852-862, 2007.

[6] Q. Quan, R. Fu, M. Li, D. Wei, Y. Gao and K. Cai, "Practical distributed control for VTOL UAVs to pass a virtual tube," in *IEEE Transactions on Intelligent Vehicles*, vol. 7, no. 2, pp. 342-353, 2022.

[7] Q. Quan, Y. Gao, and C. Bai, "Distributed control for a robotic swarm to pass through a curve virtual tube," in *Robotics and Autonomous Systems*, vol. 16, pp. 104368, 2023.

[8] P. Mao and Q. Quan, "Making robotics swarm flow more smoothly: A regular virtual tube model," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 4498-4504.

[9] Y. Gao, C. Bai and Q. Quan, "Distributed control for a multiagent system to pass through a connected quadrangle virtual tube," in *IEEE Transactions on Control of Network Systems*, vol. 10, no. 2, pp. 693-705, 2023.

[10] Y. Gao, C. Bai and Q. Quan. "Robust distributed control within a curve virtual tube for a robotic swarm under self-localization drift and precise relative navigation," in *International Journal of Robust and Nonlinear Control*, vol. 33, no. 16, pp. 9489-9513, 2023.

[11] P. Mao, R. Fu and Q. Quan. "Optimal virtual tube planning and control for swarm robotics," *arXiv preprint arXiv*: *2304.11407*, 2023.

[12] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative Adversarial Nets," in *Advances in neural information processing systems*, vol. 27, 2014.

[13] T. Zhou, Q. Li, H. Lu, Q. Cheng and X. Zhang, "GAN review: Models and medical image fusion applications," in *Information Fusion*, vol. 91, pp. 134-148, 2023.

[14] A. H. Qureshi, A. Simeonov, M. J. Bency and M. C. Yip, "Motion planning networks," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 2118-2124.

[15] J. Wang, W. Chi, C. Li, C. Wang and M. Q. -H. Meng, "Neural RRT*: Learning-based optimal path planning," in *IEEE Transactions on Automation Science and Engineering*, vol. 17, no. 4, pp. 1748-1758, 2020.

[16] N. Soboleva, and K. Yakovlev, "GAN path finder: Preliminary results," in *KI 2019: Advances in Artificial Intelligence: 42nd German Conference on AI, Kassel, Germany, September 23–26, 2019, Proceedings 42*. vol. 11793, pp. 316-324, 2019.

[17] T. Fu, and W.C. Lee, "ProgRPGAN: Progressive GAN for route planning," in *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 2021, pp. 393-403.

[18] T. Zhang, J. Wang and M. Q. -H. Meng, "Generative Adversarial Network Based Heuristics for Sampling-Based Path Planning," in *IEEE/CAA Journal of Automatica Sinica*, vol. 9, no. 1, pp. 64-74, 2022.

[19] M. Mirza, and S. Osindero, "Conditional generative adversarial nets," *arXiv preprint arXiv: 1411.1784*, 2014.