International Journal for Housing Science and Its Applications
Publish August 4, 2025. Volume 46, Issue 3 Pages 1997-2009

https://doi.org/10.70517/ijhsa463158

Multi-source Sensor Data Fusion and Deep Reinforcement Learning Anomaly Detection Method for Power UAV Inspection

Zhe Sun[1,*], Minru Kong[2] and Guiqi Zhu[1]
[1] State Grid Shaanxi Electric Power Co., Ltd. Training Center, Xi'an, Shaanxi, 710000, China
[2] State Grid Xi'an Electric Power Supply Company, Xi'an, Shaanxi, 710048, China
Corresponding authors: (e-mail: 13152091430@163.com).

Abstract Aiming at the problems of low accuracy and missing information when UAVs utilize a single sensor for obstacle avoidance, this paper designs and proposes an autonomous UAV obstacle avoidance method based on multi-sensor fusion. The improved Bayesian fusion algorithm contributes to the multi-sensor fusion, considers the use of multiple UAVs to perform power system inspection tasks collaboratively, and utilizes deep reinforcement learning for multi-UAV inspection path optimization. On the basis of the AnoGAN anomaly detection algorithm, the performance enhancement optimization of the anomaly detection technology is carried out, and a SE-f-AnoGAN model for anomaly detection of UAV power inspection images is designed. The model draws on the idea of attention mechanism, and introduces a compressed activation network based on channel attention into the encoder of f-AnoGAN, which captures the information of each channel from the global field of view category, so as to improve the accuracy of anomaly detection. Deep reinforcement learning multi-drone optimization path and multi-drone inspection image anomaly detection techniques are performed for model training and performance analysis, respectively.The DQN algorithm is designed to enable mobile drones to complete collision-free inspection path planning, and can continuously shorten the inspection path through training and learning to save inspection time.The SE-f-AnoGAN model has a high accuracy and precision rate in different dataset categories.

Index Terms multi-sensor fusion, Bayesian fusion algorithm, UAV inspection path, attention mechanism, anomaly detection

I. Introduction

In today's society, with the continuous advancement of industrialization and urbanization, the power system, as one of the infrastructures of the modern society, has a direct impact on the stable transmission and distribution of electric energy [1], [2]. However, due to long-term exposure to variable and harsh outdoor climatic conditions, the power system often faces a variety of abnormal conditions, posing a serious threat to the safe operation of the power grid [3], [4]. With the increasing demand for electricity, the traditional power system inspection methods have been difficult to meet the increasing task demands [5]. In the traditional inspection method, manual inspection is one of the main ways, which not only faces very high security risks, but also has many problems including high inspection cost, slow inspection speed, high task difficulty, etc [6].

With the rapid development of UAV technology, the use of UAVs for power line inspection has become an efficient and fast alternative [7]. UAVs have a strong flexibility and mobility can meet including mountainous areas, forests, waters, and areas that are difficult for humans to reach for filming and inspection, as well as the economic benefits of this portable equipment and real-time monitoring is widely used by modern society [8]-[11]. With the support of image processing, machine learning and other technologies, compared with traditional methods, the power drone inspection method has a higher degree of automation and recognition accuracy, which can effectively improve the efficiency and safety of power line inspection [12]-[15]. Therefore, the proposed intelligent inspection technology based on multi-source sensor data fusion and deep reinforcement learning can provide a new technical way for the intelligent operation and maintenance of the power system, which has important theoretical and practical significance.

This paper designs the UAV transmission line inspection system composed of UAV control system, data acquisition system, data transmission system, ground station control platform and so on. The 2D LiDAR and depth camera are synchronized in time and space under the mechanism of the ROS platform's helping role is completed. Introduce A* algorithm and propose UAV obstacle avoidance path search based on A* algorithm. Planning the UAV swarm power system inspection model, using deep reinforcement learning model for UAV swarm inspection path optimization. From generative adversarial network, further propose SE-f-AnoGAN model for UAV power inspection image anomaly detection. Analyze the UAV inspection path method optimized for the deep reinforcement learning model, and analyze the anomaly detection results of the SE-f-AnoGAN model.

II. Multi-UAV inspection collaboration on deep reinforcement learning

II. A. UAV transmission line inspection system

The transmission line inspection system based on UAV is shown in Fig. 1, which is mainly composed of UAV control system, data acquisition system, data transmission system and ground station control platform. The automatic online detection of abnormal state is realized in the inspection data intelligent analysis system. In this paper, we mainly process the images collected by the UAV, extract effective information, and realize the abnormal detection of transmission lines [16].

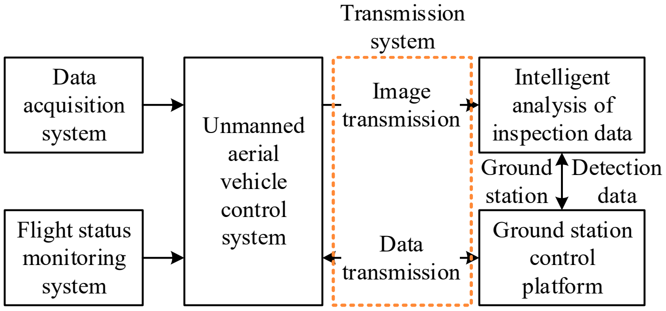
Transmission system
Data acquisition system
Flight status monitoring system
Unmanned aerial vehicle control system
Image transmission
Data transmission
Intelligent analysis of inspection data
Ground station
Detection data
Ground station control platform

Figure 1: Uav based transmission line inspection system

II. B. Autonomous UAV obstacle avoidance method based on multi-sensor fusion

In view of the limitations of single sensor, an autonomous UAV obstacle avoidance method based on multi-sensor fusion is proposed and implemented under the Robot Operating System (ROS) platform. The indoor environment information is collected using a 2D LiDAR and a depth camera, and the fused point cloud is obtained after multi-sensor fusion processing. The fused point cloud data can be used to generate an octree map to support UAV trajectory replanning to realize autonomous obstacle avoidance for UAVs in unknown environments. The flowchart of the autonomous UAV obstacle avoidance method based on multi-sensor fusion is shown in Fig. 2.

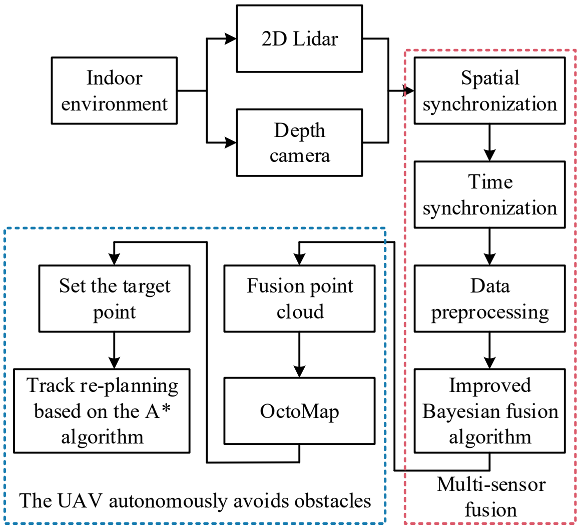

Figure 2: The flow diagram of unmanned uav based on multi-sensor fusion

II. B. 1) Multi-sensor fusion

(1) Time synchronization and space synchronization

Since the 2D LiDAR and the depth camera are mounted on different positions of the UAV, it is necessary to transform the sensors on the UAV to the same coordinate system before performing the multi-sensor fusion.

Joint calibration of the 2D LiDAR and depth camera can be performed to obtain the transformation relationship between them. Using 2D LiDAR and depth camera to observe the same position P in the space at the same time, let the coordinates of point P be (x_C, y_C, z_C) under the coordinate system of the depth camera, and the coordinates of point P be (x_L, y_L, z_L) under the coordinate system of the 2D LiDAR. The conversion relation between depth camera coordinates and 2D LiDAR coordinates can be derived as follows:

$$\begin{bmatrix} x_C \\ y_C \\ z_C \end{bmatrix} = \begin{bmatrix} R & T \\ 0^T & 1 \end{bmatrix} \begin{bmatrix} x_L \\ y_L \\ z_L \end{bmatrix} \tag{1}$$

where R denotes the transformation rotation matrix from the depth camera coordinate system to the 2D LiDAR coordinate system. T denotes the transformation translation matrix from the depth camera coordinate system to the 2D LiDAR coordinate system. The R, T matrix is solved by the least squares method, and then the rotation and translation relationship between the 2D Lidar coordinate system and the depth camera coordinate system can be determined to complete the joint calibration. Under the ROS platform, the static coordinate transformation is carried out using the tf2_ros software tool to transform the 2D Lidar and depth camera to the same coordinate system, thus realizing spatial synchronization.

When a sensor acquires data, a timestamp of the current time is obtained. And compared with the timestamp of another sensor data, it is necessary to select the sensor data whose difference between the two timestamps is less than a predetermined threshold as a single data set for fusion processing. The message_filters function package provided by ROS is utilized for time synchronization processing of multiple sensors.

(2) Improved Bayesian fusion algorithm

Bayesian estimation is a multi-sensor fusion algorithm based on Bayes' theorem [17]. It uses the observation vector Z to infer the unknown state vector X. Suppose that in a state space, the known observation $Z_k =| z_1 \cdots, z_k |$, and the probability of the state vector X at moment k is x_k, and its posterior distribution is as follows:

$$p(x_k \mid z_k) = \frac{p(z_k \mid x_k) p(x_k \mid z_{k-1})}{p(z_k \mid z_{k-1})} \tag{2}$$

where $p(z_k \mid x_k)$ denotes the likelihood function of the observed model, $p(x_k \mid z_{k-1})$ denotes the prior distribution function of the transformed system model, and $p(z_k \mid z_{k-1})$ denotes the normalized probability density function.

When updating the point cloud data using Bayesian estimation, O denotes that the point cloud is observed to exist, $\bar{O}$ denotes that the point cloud is observed to be absent, E denotes that the point cloud is real, and $\bar{E}$ denotes that the point cloud does not exist. Then the posterior probabilities are as follows:

$$P(E \mid O) = \frac{P(O \mid E)P(E)}{P(O \mid E)P(E) + P(O \mid \bar{E})P(\bar{E})} \tag{3}$$

$$P(E \mid \bar{O}) = \frac{P(\bar{O} \mid E)P(E)}{P(\bar{O} \mid E)P(E) + P(\bar{O} \mid \bar{E})P(\bar{E})} \tag{4}$$

2D LIDAR can work stably under various light conditions, providing highly accurate distance and angle measurements in indoor environments. However, it can only scan in the horizontal direction and cannot acquire information in the vertical direction. In contrast, depth cameras are able to detect three-dimensional information in the environment, but their ranging accuracy is greatly affected by environmental factors. Therefore, based on the point cloud information of 2D LiDAR, the point cloud information of depth camera is fused. Thereby, the 2D LiDAR detection range problem is compensated and the redundant information can be handled effectively. The probability formula for updating the fused point cloud is as follows:

$$P = \frac{P_s P_m}{P_s P_m + (1 - P_s)(1 - P_m)} \tag{5}$$

where P denotes the probability value of the fused point cloud update, P_m and $1 - P_m$ denote the a priori probability of the existence of the point cloud in the indoor environment with the size of 0.5, and P, denotes the likelihood probability of the presence of the point cloud detected by the depth camera, respectively.

Starting from the indoor environment and sensor characteristics, the fused point cloud is updated with the point cloud information of the 2D LIDAR, and the depth camera point cloud information is utilized to compensate for the regions not detected by the 2D LIDAR. Let a point cloud coordinate (x_i, y_i, z_i) of LiDAR and a point cloud coordinate (x_c, y_c, z_c) of depth camera, then the distance x_{dist} between the two point clouds is:

$$x_{dist} = \sqrt{(x_l - x_c)^2 + (y_l - y_c)^2 + (z_l - z_c)^2} \tag{6}$$

The likelihood function of the improved Bayesian fusion algorithm is transformed from the sigmoid function as:

$$P_s = \begin{cases} 1 x_{dist} \geq x_{\max} \\ \dfrac{2}{1 + e^{\frac{-10 s_{dist}}{s_{\max}}}} - 1 x_{dist} < x_{\max} \end{cases} \tag{7}$$

where $x_{\max}$ denotes the maximum distance value with a size of 0.Im. When the distance between two point clouds is greater than the maximum distance, the fused point cloud is updated with the point cloud information from the depth camera.

II. B. 2) Autonomous UAV obstacle avoidance

(1) OctoMap

OctoMap utilizes an octree data structure to partition the environment into voxels of different sizes and binarizes each voxel to represent its occupancy status. The octree map stored in this way can visually represent the occupied or idle state in the environment, which is conducive to the realization of UAV trajectory planning. Under the ROS platform, the fused point cloud can be converted into an octree map by calling the octomap_server function package to facilitate autonomous obstacle avoidance of the UAV.

(2) UAV obstacle avoidance path search based on A* algorithm

In the UAV obstacle avoidance path and route planning, it is necessary to find the completion of the obstacle avoidance path search in order to realize the subsequent route planning. To address this problem, the study introduces the A* algorithm, which is a fusion algorithm based on a depth-first algorithm and a breadth-first algorithm [18]. The algorithm creates a cost function mainly based on the heuristic function, which considers not only the cost of the new node's distance from the starting point, but also its distance from the goal point. The cost function of the A* algorithm is shown in Eq:

$$f(n) = g(n) + h(n) \tag{8}$$

where n denotes the point to be searched $f(n)$ denotes the path cost of the point to be searched, $g(n)$ denotes the path cost from the current node to the starting point, and $h(n)$ represents the estimated path cost from the current node to the target point. For the calculation of $h(n)$, the study mainly uses the Euclidean distance to calculate, compared with the Manhattan distance, the method can better reflect the distance of the spatial points, and its specific calculation is as follows:

$$d = sqrt((x_i + x_j)^2 + (y_i - y_j)^2) \tag{9}$$

The implementation process of the A-name algorithm in the 2D plane is the same as that in 3D space, except for the different coordinate dimensions of the computed points.

In the intelligent obstacle avoidance and route planning of UAVs, how to convert the route points searched by the A$*$ algorithm into obstacle avoidance Dubins paths is a key step. Among them, the Dubins path represents the shortest path between two positional points. The Dubins path is obtained by choosing one of the tangents to the two circles. The start and end points of this path are on the arcs of the circles. Thus, the path planning problem for UAVs can be ultimately converted into finding the common tangent of two circular arcs. At a particular bit attitude point, the UAV has the ability to turn left and right. Given a particular bit attitude point, its direction vector is represented as:

$$\vec{v} = \begin{bmatrix} \cos\theta & \sin\theta \end{bmatrix} \tag{10}$$

where θ denotes the angle between the orientation vector of the attitude point and the x-positive axis.

The centroid coordinates of the turning circle are calculated as:

$$p(x, y) = (x_c + r \cdot \cos(\varphi), y_c + r \cdot \sin(\varphi)) \tag{11}$$

where (x_c, y_c) denotes the coordinates of the center of the circle, r is the radius, and φ is the angle of the point on the circle. There exist two ways of turning the point. Thus, there correspond to two tangent circles. If the arc segment C2 of the left turn is labeled L. It is computed as:

$$L(x, y) = (x_c + r \cdot \cos(\theta + \pi / 2 + v), y_c + r \cdot \sin(\theta + \pi / 2 + v)) \tag{12}$$

where θ denotes the amount of angular change.

The arc segment C1 in the right turn is labeled R and is calculated as:

$$R(x, y) = (x_c + r \cdot \cos(\theta + \pi / 2 - v), y_c + r \cdot \sin(\theta + \pi / 2 - v)) \tag{13}$$

Four Dubins paths can be generated for a single bit position point.

II. C. Deep Reinforcement Learning Based Multi-UAV Inspection Path Planning

II. C. 1) Planning Model for UAV Power System Inspections

Consider the construction of a power system that uses drones for inspection, including multiple power towers, with any one of the power towers serving as a charging station point for the drone. From any charging station point, the UAV is required to inspect the transmission line between any two power towers in order to accomplish the inspection task.

The UAV has a range of L and a speed of v, and it takes a budget of S to send out a fully charged UAV, and it needs to eventually return to any one of the stations for recharging. There are n power towers with coordinates $D_j = \left(x_j, y_j\right)_{j=1}^{n}$, where l towers are used as charging stations. In the inspection planning of the power system, the path of the UAV should minimize the total traveling length to reduce the energy consumption and time cost. At the same time, for a single UAV, the time required to complete the inspection task should also be minimized to improve the inspection efficiency. In addition, considering the economic cost, deploying more drones will increase the budget expenditure, so the number of drones needs to be minimized as much as possible.

Multiple UAVs are considered to perform power system inspection tasks collaboratively, so, as mentioned above, for n power towers in an area, m inspection UAVs take off from their respective departure points, inspect all the towers under their responsibility n_i sequentially in a planned order, and eventually return to any one of the departure sites. Where, in order to simplify the problem, the influence of the current environmental wind field on the UAV range is not considered. The objective function is shown in Eq. (14) to Eq. (16):

$$MinF_1 = \max\left\{\sum_{i=1}^{n_i}\sum_{j=1}^{n_i} x_{i,j} C_{i,j} / v_{i,j}, k = 1, 2, \cdots, m\right\} \tag{14}$$

$$MinF_2 = \sum_{i=1}^{m} S \tag{15}$$

$$MinF_3 = \sum_{k=1}^{m}\sum_{i=1}^{n_i}\sum_{j=1}^{n_i} x_{i,j} C_{i,j} / v_{i,j} \tag{16}$$

where $x_{i,j}$ is an indicative state scalar. When $x_{i,j} = 1$, it represents that this UAV flies from power tower i to power tower j. Conversely, when it is 0, it represents that the UAV does not fly from power tower i to power tower j. $C_{i,j}$ is the distance between power tower i and power tower j. v is the speed after setting the speed of all drones constant.

II. C. 2) Deep Reinforcement Learning-based Optimization for UAV Inspection

Deep reinforcement learning is a method that combines deep learning and reinforcement learning, and its core idea is to learn through the interaction between the intelligent body and the environment so that the intelligent body can find the optimal behavioral strategy. In the multi-site UAV power system inspection planning model, the UAV can be regarded as the intelligent body and the power system environment as the environment, and the optimal inspection path planning strategy can be learned through the interaction between the intelligent body and the environment.

(1) State Representation

In UAV power system inspection planning, the selection of states is crucial for the effect of deep Q-learning. The state should accurately reflect the information of the environment where the UAV is located so that the intelligent body can make appropriate decisions. Considering that the UAV needs to start from any charging station point and pass through the transmission line between any two power towers, the state can be represented as a three-dimensional matrix.

The state s can be represented as a 3D matrix of $n \times n \times (l+1)$, where n is the number of power towers and l is the number of charging station points. The element s_{ijk} in the matrix is the case where the UAV flies from the power tower i to the power tower j while considering the charging station point k, $s_{ijk} = 1$ means that the UAV flies from the power tower i to the power tower j and passes through the charging station point k, and $s_{ijk} = 0$ means the negative case.

(2) Action space

The action space defines the actions that the UAV can take in the current state. In this model, the action space can be represented as all possible actions to fly from the current power tower to other power towers.

The action space A is a binary group (p, D), where p is the number of the power tower where the current UAV is located. D is a set containing the numbers of the target power towers that the UAV can fly to. With such an action space definition, it is possible to clearly understand what are the feasible flight targets for the UAV under the current position, providing a clear range of choices for the intelligent body's decision-making.

(3) Reward function design

The design of the reward function is a crucial part of the deep Q learning model (DQN), which directly affects the behavior and learning effect of the intelligences. In the UAV power system inspection planning model, the objectives are to minimize the total travel length, minimize the time required for the UAV to complete the inspection task, and minimize the total budget consumption. Therefore, the following reward function can be designed.

Traveling distance reward. A reward function can be designed to penalize the UAV for choosing a long distance path. Specifically, the reward function R_d is as follows:

$$R_d(i, j) = -C_{ij} \times x_{ij} \tag{17}$$

where C_{ij} is the distance between power tower i and power tower j. x_{ij} is an indicative state scalar indicating whether the UAV chooses to fly from power tower i to power tower j for inspection or not, $x_{ij} = 1$ means that the path is chosen and $x_{ij} = 0$ means that the path is not chosen.

Range Penalty. In order to ensure that the inspection path of the UAV does not exceed its range distance L, it is necessary to penalize the path selection that exceeds the range distance. Specifically, a penalty function P_l can be designed to penalize the path selection that exceeds the range distance. The penalty function is as follows:

$$P_i(i, j) = \begin{cases} 0 & \text{if} \sum_{i=1}^{n} \sum_{j=1}^{n} x_{ij} C_{ij} \le L \\ -\infty & \text{otherwise} \end{cases} \tag{18}$$

The total reward function is designed as:

$$R(i, j) = R_d(i, j) + t + P_l(i, j) + S \tag{19}$$

where $R_d(i, j)$ is the distance traveled reward. t is the time-to-completion reward, $P_l(i, j)$ is the range penalty. S is the total budget consumption penalty.

III. Deep learning-based anomaly detection for UAV inspection images

III. A. Generating Adversarial Networks

Generative Adversarial Network (GAN) is a generative model composed of deep learning models.The basic structure of GAN consists of two neural networks: generator and discriminator. The role of the generator is to transform the input random noise into data similar to the real data, while the discriminator determines whether the input data is real or not based on the input data. The core idea is to make the generator capable of generating high-quality and realistic fake data while the discriminator can accurately distinguish between real and fake data through the adversarial training of the two modules.

During the training process, the generator and the discriminator are trained alternately, the generator tries to generate more realistic forged images, while the discriminator tries to distinguish the forged images from the real ones. The training process of the two networks can be given by Eq:

$$\min_G \max_D L(D, G) = E_{x \sim p_{dea}(x)}[\log D(x)] + E_{z \sim p_z(z)}[\log(1 - D(G(z)))] \tag{20}$$

Where D denotes the discriminator, G denotes the generator, x denotes the real image, z denotes the random noise vector, $p_{data}(x)$ denotes the real data distribution, and $p_z(z)$ denotes the noise distribution. The loss function consists of two parts, the first part represents the probability that a real image is recognized as real and the second part represents the probability that a fake image is recognized as real. The objective of the generator is to minimize the probability that the discriminator discriminates a forged image as a forgery, i.e., to maximize $\log(1-D(G(z)))$, whereas the objective of the discriminator is to maximize the probability that the discriminator discriminates a real image as a real image as well as the probability that a forged image is determined as a forgery, i.e.,:

$$\log D(x)+\log(1-D(G(z))) \tag{21}$$

III. B. AnoGAN anomaly detection algorithm

The AnoGAN anomaly detection algorithm is a DCGAN based approach for image anomaly detection. The model consists of two parts, a generator and a discriminator. The generator network employs a series of anti-convolutional layers to map a latent vector Z_γ to an image similar to the training data. A batch normalization layer and an activation function are added after each of the deconvolutional layers to speed up training and improve the stability of the model. The discriminator network employs a series of convolutional layers that can classify the input image to determine whether the image is real training data or fake data generated by the generator. After each convolutional layer, a batch normalization layer and an activation function are added to improve the stability and generalization of the model.

The basic idea of the algorithm is that during training, DCGAN is utilized to learn the streaming distribution of normal samples in the latent space by using only normal samples. During testing, the input vector Z_γ that is closest to representing the input image is trained by back-propagation iterations of the already defined loss function, and the anomalies can be localized by feeding the already trained ideal vector Z_γ into the generator in order to generate the recovered image, and comparing the generated image with the original image. The anomaly score is then defined by the following loss function, above a certain threshold the original image is considered to have anomalies:

$$L(Z_\gamma)=(1-\lambda)\cdot L_R(Z_\gamma)+\lambda\cdot L_D(Z_\gamma) \tag{22}$$

where λ is the weight parameter. L_R is defined as shown in equation (23) for the L1 loss of the original and generated images, where G is the generator. The definition of L_D is shown in equation (24), where f is the output of a feature layer of the discriminator and G is the generator. Namely:

$$L_R(Z_\gamma)=\sum\left|x-G(Z_\gamma)\right| \tag{23}$$

$$L_D(Z_\gamma)=\sum\left|f(x)-f(G(Z_\gamma))\right| \tag{24}$$

However, the algorithm also has some drawbacks. First, the training process of GAN is very unstable and problems such as gradient vanishing and pattern crashing may occur, which can lead to the need for time-consuming adjustments to the model architecture and hyperparameters. Second, since the images generated by GAN are generated from a random noise vector, it is difficult to infer this noise vector by inverse operations, which makes it difficult to control and interpret in certain tasks. In addition, the algorithm needs to find the mapping of the image to be tested x to the input vector Z_γ when performing the inference, which is a cumbersome process that requires additional computational resources and time. Finally, since the algorithm directly employs a difference-making approach to compare the original and generated images for anomaly detection, this is prone to false detection due to the poor quality of the generated images.

III. C. SE-f-AnoGAN based anomaly detection

For this reason, f-AnoGAN optimizes the model structure and training process on this basis, respectively. Different from AnoGAN, the model introduces an encoder before the generator, which enables the images to be quickly mapped to get the feature representations in the latent space, eliminating the process of constant iterative optimization in the inference stage to find the best mapping, which greatly improves the detection speed. In addition, since the KL scatter in GAN cannot measure the distance between two non-overlapping distributions, f-AnoGAN introduces a Wasserstein generative adversarial network with a gradient penalty. Namely:

$$L_{WGAN-GP}=E_{x\sim P_G}[D(x)]-E_{x\sim P_{dea}}[D(x)]+\alpha E_{x\sim P_{peaky}}(\left\|\nabla_x D(x)\right\|-1)^2 \tag{25}$$

where the combination of the first two terms is the WGAN loss and the latter term is the gradient penalty loss. $P_{penalty}$ is the penalized distribution that lies between the distribution of real images and the distribution of generated images. $\nabla_x D(x)$ is the gradient of the discriminator and α is the tradeoff parameter. The Wasserstein distance is used instead of the KL scatter and a gradient penalty is added to the loss function to smooth the objective function, which solves the problems of training instability, pattern runout, and vanishing gradient in GAN.

In f-AnoGAN, the encoder loss function L_E consists of two parts: image reconstruction loss and feature residual loss. Namely:

$$L_E = \frac{1}{n}\left\|x - G(\hat{z})\right\|^2 + \lambda \frac{1}{n_d}\left\|f(x) - f(G(\hat{z}))\right\|^2 \tag{26}$$

where the first term denotes the picture reconstruction loss and the second term denotes the residual loss. G is the generative network. x is the original image $\hat{z}$ is the feature vector after the encoding network. n_d is the length of the vector after passing through the intermediate layer of the discriminator, and f is a nonlinear function. n is the original image size and λ is the hyperparameter.

In this paper, the SE-f-AnoGAN model for anomaly detection of UAV power inspection images is proposed on the basis of f-AnoGAN. The SE-f-AnoGAN model consists of a channel attention-based encoder, generator, discriminator, and binary classifier. Drawing on the idea of attention mechanism, a channel attention-based compression activation network (SENet) is introduced into the encoder of f-AnoGAN to expand the sensory field by aggregating spatial information and capture the important information of each channel in the high-dimensional feature map from a global perspective.

The SENet network structure consists of three parts: compression, activation, and weight assignment.

Firstly, the compression operation is carried out as follows. Namely:

$$z_c = \frac{1}{HW}\sum_{i=1}^{H}\sum_{j=1}^{W} u_c(i,j) \quad z \in R^c \tag{27}$$

where $u_c(i,j)$ is the pixel value at position (i,j) on the feature map of the c th channel. H,W are the length and width of the feature map, respectively, and z_c is the global average pooling result for the c th channel. R^c denotes the set of all channel feature maps real numbers.

Then comes the activation operation, which introduces two fully connected layers to automatically learn the weights of different channels. Namely:

$$s = \sigma(W_2 \cdot Re\,\mathrm{LU}(W_1 z)) \tag{28}$$

where W_1 and W_2 are the weight parameters of the fully connected layer. z is the semantic information value of the feature map channel.

Since the number of abnormal power images in the dataset is small while the number of normal power images is large, if a dataset with a highly unbalanced number of categories is used to train the binary classifier directly. It usually leads to the normal class gradient generated by the model during backpropagation to overwhelm the abnormal class, which ultimately makes the model's prediction more skewed towards the normal class, while the discriminative ability of the abnormal class is weaker. Therefore, in this paper, we refer to the classifier training methods based on long-tailed distribution datasets, and conduct research in terms of resampling and reweighting respectively.

Resampling is one of the most effective ways to solve the problem of category imbalance in long-tailed datasets, which is a method to achieve relative balance by readjusting the number of samples of different categories in each round of training of the network. The generalized formula for resampling is as follows:

$$p_i = \frac{n_i^m}{\sum_{j=1}^{C} n_j^m} \tag{29}$$

where p_i is the sampling frequency of the i th category, n_i and n_j are the number of samples of the i th and j th categories, respectively, c is the total number of categories, and m takes the value range of [0,1].

The reweighting method is a method that adjusts the weights of the loss function according to the proportion of the number of samples in each category. Since this method needs to update the feature centers during each round of training, resulting in a larger computational effort during model training, it is more suitable for small-scale datasets. In addition, individual sample noise will shift the feature center, which ultimately has some impact on the detection results. The learnable parameter tuning method is similar to the classifier retraining method, where a normalization term with hyperparameters is introduced on the basis of the original parameters of the network, and the optimal hyperparameters are obtained through the network autonomous learning training. Namely:

$$\tilde{w}_i = f_i * w_i, \text{where } f_i = \frac{1}{\|w_i\|^{\tau}} \tag{30}$$

where w_i is the original parameters of the network, f_i is the normalization term, τ is the hyperparameter, and $t\tilde{w}_i$ is the optimized parameter.

IV. Power UAV inspection anomaly detection algorithm analysis

IV. A. Deep Reinforcement Learning Patrol Optimization

IV. A. 1) Model training

For the path planning of multi-inspection UAVs, Tensorflow2.0 is used to construct the DQN model, and the model training is carried out with NVIDIA RTX 3060 GPU and CUDA11.0 GPU acceleration library.

The dimensions of the input and output layers of the DQN model are set according to the dimensions of the actual state space and action space of the case, and the intermediate layer adopts a typical structure. The relevant training parameters are adjusted according to the computer memory and computing speed, and the parameter settings are shown in Table 1.

Table 1: Parameter setting

	Parameter	Value
Parameter setting	Discount factor γ	0.93
	Learning rate	0.005
Network structure	Input layer - the first dimension	80*265
	The first layer - the second dimension	265*265
	The second - third layer dimension	265*64
	Third layer - output layer dimension	64*9
Training Settings	The maximum number of steps for each training	1800
	The size of each batch of training data	20
	Experience pool size	300

The process of reward value change during training is shown in Fig. 3. When the model starts training, there are no records in the experience pool, so it keeps exploring in the beginning stage, and the reward value is low and in an oscillating state.After the 400th training, it starts to optimize the network parameters by randomly taking out the records in the experience pool, and the reward value rises slowly thereafter. At about the 500th to 700th iterations, the model once fell into a local optimum. When the number of iterations approaches 1800 the model converges and the reward value stabilizes. This indicates a gradual decrease in the number of useless actions required to complete the path planning.

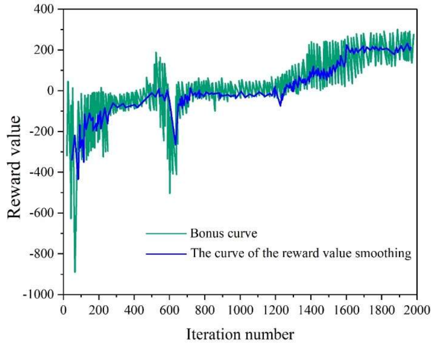

Figure 3: The process of winning the value of the training process

IV. A. 2) Simulation experiment construction

Adopting 20*20 square map, square size 40*40, using 35*35 red triangles instead of moving UAVs, setting blue dots on behalf of inspection start and end points, green squares on behalf of inspection detection points, and adding black obstacles between detection points, the experimental map is shown in Figure 4. In order to verify whether the design algorithm can obtain collision-free path planning, the experiment is stopped when the design inspection UAV collides with obstacles. Obstacles are set to block between inspection points to verify the ability of inspection UAV to bypass obstacles, and narrow channels are set between obstacles to verify whether the algorithm will fall into local optimization. Set the end of training after completing the inspection task 300 times.

When the number of memories stored in the memory pool is greater than 2000, the action-value network starts learning. The ε -greedy strategy is the initial ε probability is set to 1, and the greedy value is 0. Every time the action-value network learns, the ε probability decreases by g, and the greedy value increases by g. When the inspection UAV action decision is made, a 0~1 random number is generated, and when the random number is greater than the greedy value, the random action is selected. When the random number is smaller than greedy value, the maximum action value corresponding to the action is selected.

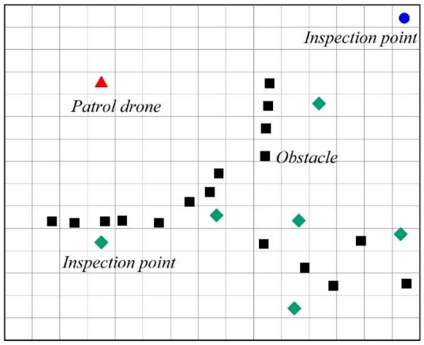

Figure 4: Laboratory chart

The inspection path length is shown in Fig. 5. At the beginning of the inspection task, the mobile UAV has a high number of actions, especially at 0~25 tasks, the mobile UAV can reach more than 16,000 actions to avoid obstacles.

Comprehensive inspection tasks and inspection path length can be concluded that the total length of the training path is 185631, the average path length is 637, and the training time is about 430 s. The above DQN algorithm design can enable the mobile UAV to complete the collision-free inspection path planning, and can continuously shorten the inspection path through training and learning. However, the learning efficiency is low, the training time is long, and the optimal path (the actual optimal path is 80 steps) is not learned by the time 300 inspection tasks are completed. The main reason is that in the late stage of inspection UAV learning, the path is gradually shortened, the number of learning times for completing an inspection task is gradually reduced, and the increase in greedy value becomes slower, and when 300 inspection tasks are completed, the greedy value is not increased to 1, and the inspection UAV does not choose the optimal action when making decisions about its actions. If the number of inspection tasks is increased, the inspection path can be further shortened, but the total path of the inspection UAV and the training time also increase. Therefore, further optimization of the above DQN algorithm design can be considered to reduce the number of training times, shorten the training time, improve the learning efficiency of the inspection UAV, so that the action value network is fully converged, and ultimately, the inspection UAV can learn to get the optimal path.

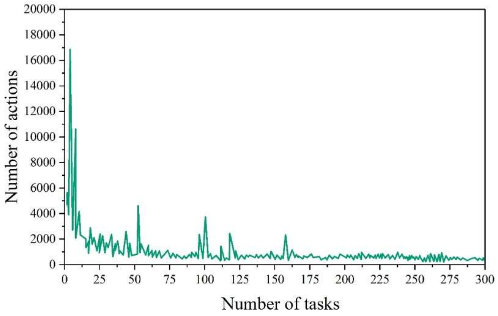

Figure 5: Length of inspection path

IV. B. Inspection image anomaly detection analysis

IV. B. 1) Experimental platforms

The experimental platform is based on Windows10 operating system, CPU model for the processor 12th Gen Intel(R) Core(TM) i7-12700H, GPU model for the NVIDIA GeForce RTX3060, the main frequency of 3.7Ghz, 12 cores and 24 threads, the operating memory is 32G. Data storage hardware for the 1T solid state drive.

Programming environment using Windows10 operating system, Python3.7 and Pytorch deep neural network development tools, this paper uses Pytorch deep learning framework has been widely used, with a quick start, the code is concise and flexible, debugging is convenient and so on.

IV. B. 2) Detection data sets

In this paper, MVTec AD dataset is used as the main detection dataset. This dataset covers high-definition images of different objects and texture types.The MVTec AD dataset contains a wide range of anomaly types, including 15 different categories. These categories include images with regular patterns, such as carpets and grids, and images with irregular patterns, such as leather, tiles and wood. In addition to this, the dataset also covers 10 different object categories, covering everything from rigid objects with well-defined shapes, such as zippers, transistors, and screws, to deformable objects, such as wires and hazelnuts.The small percentage of anomalous data in the MVTec AD dataset allows for a better validation of the capabilities of the SE-f-AnoGAN anomaly detection algorithm.

IV. B. 3) Abnormal detection results

The results of different types of anomaly detection in the dataset are shown in Fig. 6. The precision and accuracy rate of SE-f-AnoGAN model can also maintain high accuracy and precision rate in different data types. And there are different types of defects in each data type, which can also be detected accurately. Compared with the original DCGAN model, the recall and specificity also maintain high values, reducing the probability of missed and false detections. It proves a significant optimization of the DCGAN network model.

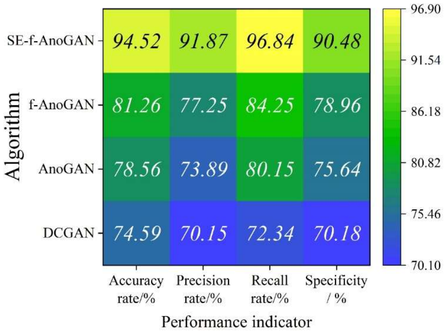

Figure 6: Data set different types of abnormal detection results

Continuing with the comparison experiments, the activation functions ReLU and GeLU of the DCGAN network and the middle generation model of the improved D-DCGAN network are compared in the data enhancement, keeping other parameters unchanged. The validity of the generated images is verified by FID and SSIM, while the results of evaluating different activation functions with different defects of the same type are chosen as shown in Fig. 7. The bars in the figure represent the FID values for different activation functions and the data in the table are the AVG-SSIM values for different activation functions.

It can be found that the FID values of ReLU are 50.26, 35.45, 103.29, and 42.38, while its Avg-SSIM values are 0.7263, 0.8124, 0.6425, and 0.7893, respectively.Whether it is from the SSIM values or from the FID, it can be clearly seen that the performance of the improved D-DCGAN is superior to that of the DCGAN, thus proving that the ReLU function improves the quality of the generated images.

The above discussion confirms the effectiveness of the optimization strategy, and a comprehensive evaluation of the detection performance of the SE-f-AnoGAN models is still needed. Therefore, several GAN models that perform well in image reconstruction and are able to achieve anomaly detection by training normal samples only are selected, including CycleGAN, EBGAN, f-AnoGAN, and MAD-GAN are taken as references for comparison experiments with the SE-f-AnoGAN model. The following experiments are conducted to compare their detection performance on the dataset of this paper in terms of each evaluation index. The experimental results are shown in Fig. 8.

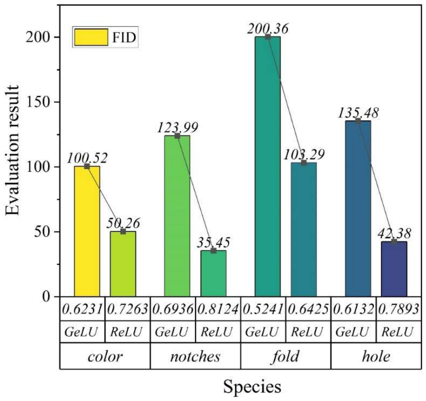

Figure 7: Evaluation of different activation functions

The results of the SE-f-AnoGAN model for each detection metric are 94.52%, 91.87%, 96.84%, 90.48%, and 95.54, respectively.Due to CycleGAN's advantage in generating high-quality images, it performs well in distinguishing between normal and abnormal samples (specificity), but its accuracy is slightly insufficient.EBGAN is trained adversarially to master real data features and combined with the reconstruction error of the self-encoder, it has a better ability to recognize abnormal samples, which makes it slightly better in specificity, but the overall detection effect still needs to be improved. The SE-f-AnoGAN model proposed in this paper, on the other hand, outperforms the other models in terms of precision, recall, and F1 score, showing a more balanced performance, thus proving the effectiveness of the improved D-DCGAN in improving the quality of the image generation, and thus enhancing the model's generalization ability and detection accuracy.

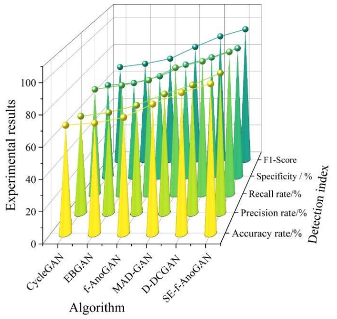

Figure 8: Experimental results

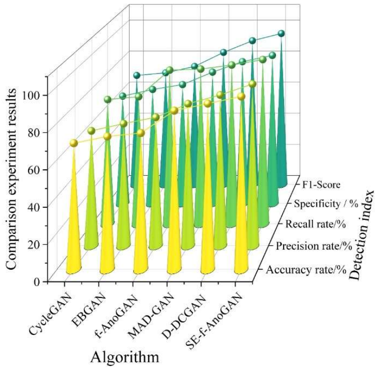

Figure 9: The comparison of the results after adding the attention mechanism

Add the attention mechanism to the networks CycleGAN, EBGAN, f-AnoGAN, and MAD-GAN, and continue to do comparison experiments on the data and on this paper. The results of the comparison experiments after adding the attention mechanism are shown in Fig. 9. The detection results of SE-f-AnoGAN model here are the same as the above figure. The accuracy, precision, and recall of the f-AnoGAN model of each comparison algorithm after adding the attention mechanism are 74.52%, 72.96%, and 90.63%, respectively, which are 1.27%, 2.33%, and

13.11% higher than those before adding. The dataset used in this paper has high quality images, which will directly improve the learning effect of the model, which in turn leads to higher quality images reconstructed by the generator. In this way, the defective features in the image will be more prominent, which will help to improve the performance metrics. So the four algorithms have a significant numerical increase in precision, accuracy, recall, and specificity with a large rise after the addition of the attention mechanism. It is proved that the addition of self-attention mechanism in the algorithms plays a certain effect of performance improvement.

V. Conclusion

In this paper, we design a UAV power inspection system, introduce a multi-sensor fusion algorithm to design an autonomous obstacle avoidance method for UAVs, and incorporate a deep reinforcement learning model to co-optimize the UAV inspection path. The inspection image of the joint UAV swarm proposes deep learning-based anomaly detection technology.

(1) The path planning of the inspection UAV swarm uses Tensorflow2.0 to construct the DQN model, and the DQN model converges when the number of iterations is close to 1800, and the reward value tends to stabilize, at which time the inspection path of the UAV swarm is optimal, and the useless actions are basically reduced. The duration of the UAV swarm completing the inspection task based on the DQN model is about 430s.

(2) Using the publicly available MVTec AD dataset as the main detection dataset, the deep learning-based UAV inspection image anomaly detection technique designed in this paper is able to maintain a high accuracy and precision rate even in different data types. That is, the results of the SE-f-AnoGAN model for each detection index are 94.52%, 91.87%, 96.84%, 90.48%, and 95.54, respectively.

Multi-sensor fusion oriented UAV swarm power inspection path planning and deep learning anomaly detection technology can not only achieve the best UAV power inspection path, but also optimize the anomaly detection accuracy and precision rate, which can provide some substantial help in power UAV inspection.

References

[1] Pan, K., Palensky, P., & Esfahani, P. M. (2019). From static to dynamic anomaly detection with application to power system cyber security. IEEE Transactions on Power Systems, 35(2), 1584-1596.

[2] Yang, L., Fan, J., Liu, Y., Li, E., Peng, J., & Liang, Z. (2020). A review on state-of-the-art power line inspection techniques. IEEE Transactions on Instrumentation and Measurement, 69(12), 9350-9365.

[3] Xu, C., Li, Q., Zhou, Q., Zhang, S., Yu, D., & Ma, Y. (2022). Power line-guided automatic electric transmission line inspection system. IEEE Transactions on instrumentation and measurement, 71, 1-18.

[4] Pagnano, A., Höpf, M., & Teti, R. (2013). A roadmap for automated power line inspection. Maintenance and repair. Procedia Cirp, 12, 234-239.

[5] Lin, T., & Pham, H. (2022). Modeling security surveillance systems with state dependent inspection-maintenance strategy. IEEE Transactions on Computational Social Systems, 10(5), 2467-2478.

[6] Huang, J., Xu, J., Meng, L., & Qin, Z. (2016, April). The comprehensive benefit evaluation model of manual inspection in transmission line. In 6th International Conference on Electronic, Mechanical, Information and Management Society (pp. 1541-1548). Atlantis Press.

[7] Luo, Y., Yu, X., Yang, D., & Zhou, B. (2023). A survey of intelligent transmission line inspection based on unmanned aerial vehicle. Artificial Intelligence Review, 56(1), 173-201.

[8] Lopez Lopez, R., Batista Sanchez, M. J., Perez Jimenez, M., Arrue, B. C., & Ollero, A. (2021). Autonomous uav system for cleaning insulators in power line inspection and maintenance. Sensors, 21(24), 8488.

[9] Kim, S., Kim, D., Jeong, S., Ham, J. W., Lee, J. K., & Oh, K. Y. (2020). Fault diagnosis of power transmission lines using a UAV-mounted smart inspection system. IEEE access, 8, 149999-150009.

[10] Li, Z., Zhang, Y., Wu, H., Suzuki, S., Namiki, A., & Wang, W. (2023). Design and application of a UAV autonomous inspection system for high-voltage power transmission lines. Remote Sensing, 15(3), 865.

[11] Boukabou, I., & Kaabouch, N. (2024). Electric and magnetic fields analysis of the safety distance for UAV inspection around extra-high voltage transmission lines. Drones, 8(2), 47.

[12] Foudeh, H. A., Luk, P. C. K., & Whidborne, J. F. (2021). An advanced unmanned aerial vehicle (UAV) approach via learning-based control for overhead power line monitoring: A comprehensive review. IEEE Access, 9, 130410-130433.

[13] Lekidis, A., Anastasiadis, A. G., & Vokas, G. A. (2022). Electricity infrastructure inspection using AI and edge platform-based UAVs. Energy Reports, 8, 1394-1411.

[14] He, Y., Liu, Z., Guo, Y., Zhu, Q., Fang, Y., Yin, Y., ... & Liu, Z. (2024). UAV based sensing and imaging technologies for power system detection, monitoring and inspection: a review. Nondestructive Testing and Evaluation, 1-68.

[15] Chen, D. Q., Guo, X. H., Huang, P., & Li, F. H. (2020). Safety distance analysis of 500kv transmission line tower uav patrol inspection. IEEE Letters on Electromagnetic Compatibility Practice and Applications, 2(4), 124-128.

[16] Cong Hu,Lingfeng Lv & Tian Zhou. (2025). UAV inspection insulator defect detection method based on dynamic adaptation improved YOLOv8. Journal of Real-Time Image Processing,22(2),74-74.

[17] Haruki Yotsui,Kodai Matsuoka & Kiyoyuki Kaito. (2025). Dual sampling method for evaluating uncertainty when updating a Bayesian estimation model of a high-speed railway bridge. Reliability Engineering and System Safety,259,110901-110901.

[18] Yufei Yang,Zhenghua Meng,Wei Guo & Jinbo Ma. (2025). Full coverage path planning for multiple farming machines based on an improved A* algorithm. Journal of Physics: Conference Series,2991(1),012001-012001.