UAV Aerial Image Generation of Crucial Components of High-Voltage Transmission Lines Based on Multi-Level Generative Adversarial Network

Jinyu Wang 1,2, Yingna Li 1,2,∗ and Wenxiang Chen 1,2

1 Faculty of Information Engineering and Automation, Kunming University of Science and Technology, Kunming 650500, China
2 Computer Technology Application Key Lab of the Yunnan Province, Kunming 650500, China
∗ Correspondence: liyingna@kust.edu.cn

Abstract: With the aim of improving the image quality of the crucial components of transmission lines taken by unmanned aerial vehicles (UAV), a priori work on the defective fault location of high-voltage transmission lines has attracted great attention from researchers in the UAV field. In recent years, generative adversarial nets (GAN) have achieved good results in image generation tasks. However, the generation of high-resolution images with rich semantic details from complex backgrounds is still challenging. Therefore, we propose a novel GANs-based image generation model to be used for the critical components of power lines. However, to solve the problems related to image backgrounds in public data sets, considering that the image background of the common data set CPLID (Chinese Power Line Insulator Dataset) is simple. However, it cannot fully reflect the complex environments of transmission line images; therefore, we established an image data set named “KCIGD” (The Key Component Image Generation Dataset), which can be used for model training. CFM-GAN (GAN networks based on coarse–fine-grained generators and multiscale discriminators) can generate the images of the critical components of transmission lines with rich semantic details and high resolutions. CFM-GAN can provide high-quality image inputs for transmission line fault detection and line inspection models to guarantee the safe operation of power systems. Additionally, we can use these high-quality images to expand the data set. In addition, CFM-GAN consists of two generators and multiple discriminators, which can be flexibly applied to image generation tasks in other scenarios. We introduce a penalty mechanism-related Monte Carlo search (MCS) approach in the CFM-GAN model to introduce more semantic details in the generated images. Moreover, we presented a multiscale discriminator structure according to the multitask learning mechanisms to effectively enhance the quality of the generated images. Eventually, the experiments using the CFM-GAN model on the KCIGD dataset and the publicly available CPLID indicated that the model used in this work outperformed existing mainstream models in improving image resolution and quality.

Keywords: unmanned aerial vehicle (UAV); high-voltage transmission line inspection; generative adversarial nets; image generation

1. Introduction

The quality of images—vital for the correct and comprehensive communication of their content to the outside world, especially in specialized fields such as aerospace, medical treatment, and electrical power—often plays an important role in their use [1]. High-voltage transmission lines cover a large area in complex and diverse geographical environments (mountains, basins, reservoirs, etc.). Affected by their structure, the natural environment, and other factors, high-voltage transmission lines are highly susceptible to defective faults such as insulator defects, anti-vibration hammer offsets, lighting rod breaks, space rod fractures, and tower rust. These failures can seriously endanger the safe operation of the power system [2].
With the continuous development of domestic UAV technology, UAVs have especially made breakthroughs in cost and size reduction, making UAVs widely used in the industry [3]. In terms of transmission line inspection, the use of drones is also of great value. Drones, for example, can acquire multiple more image data simultaneously as a workforce. However, the images taken by the UAV are affected by geographical factors such as high altitude and windy conditions, making it dither and challenging to focus the optical sensors carried by the UAV, resulting in blurred images. In addition, the shot image will also be affected by occlusion, backlight shooting, and shooting angle, resulting in reduced quality of the shot image. If the captured images are fed directly into the transmission line fault detection and inspection models, then the training effect of the model will be seriously affected, making the final recognition rate low, thus laying a significant hidden danger to the safe operation of transmission lines. Therefore, studying the generation and processing of high-quality images on transmission lines is of great practical significance.

In recent years, with the rapid development of artificial intelligence, various deep learning-based generative models have achieved good results both at the theoretical and application levels. Currently, common image generation techniques include the autoregressive model [4], variational auto-encoder model (VAE) [5], flow-based model [6–8], and generation adversarial network (GAN) [9]. However, in 2016, Oord introduced the autoregressive model in the image generation task. The autoregressive model uses pixel-by-pixel learning and prediction, which makes the computational parallelism of image generation low, and the sampling speed is slow. However, VAE has made exemplary achievements in some fields. However, VAE has made outstanding achievements in some areas. However, they are still limited by the high dimensional characteristics of the probability density distribution of sample data, thus it is very challenging to learn a model that can fit its data distribution. As one of the deep-generation models with excellent development potential, the image generation effect of the flow model is not only inferior to that of GAN models but also accompanied by complex computational problems while generating high-quality images. In a word, the existing issues of these models have affected their widespread application in the field of image generation to varying degrees.

GAN has a simple overall structure and can generate a large number of image data samples without the construction of complex objective functions. However, also, the GAN technology continuously improves its network performance through the connection between the generator and the discriminator, providing practical solutions for the sampling and training problems under a high-dimensional probability density distribution, which makes the GAN favored by the majority of researchers. However, some drawbacks have gradually emerged with the continuous extensions of GAN in practical engineering applications. Since the discriminator only judges whether its input is from the actual sample without considering the diverse characteristics of the input data. Such a behavior of the discriminator can seriously affect the model training of the generator, making the generator unable to truly understand the accurate distribution of the input data, which eventually leads to poor diversity of the images generated by the generator. Additionally, classic GANs often struggle to capture images’ structure, texture, detail, etc., meaning they cannot directly produce many high-quality (i.e., high-resolution) images of different categories.

In response to these problems, researchers have proposed a large body of work based on improvements to GAN. For instance, deep convolutional GAN (DCGAN) [10] introduced deep convolutional neural networks to GAN for the first time. Wasserstein GAN (WGAN) [11] introduced Lipschitz continuous row constraints for discriminative networks. Information-maximizing generative adversarial nets (InfoGAN) [12] improved the model from an information-theoretic perspective. Additionally, conditional generative adversarial networks (CGAN) [13] introduced auxiliary variables. Apart from that, there are also quite a few researchers who have now used GAN to achieve excellent effects in font restoration [14], image conversion [15], high-resolution image semantic segmentation [16], and other tasks.
In the image generation task, the Matching-aware discriminator and Learning with manifold interpolation (GAN-INT-CLS) [17] can generate an image resolution of $64 \times 64$. To further improve the resolution, the text-conditioned auxiliary classifier generative adversarial network (TAC-GAN) [18] uses an auxiliary classifier generative adversarial network (AC-GAN) [19] for the text-to-image (T2I) task. The TAC-GAN fed the category labels and text description vectors into the generator as conditional information. The discriminator distinguishes between real and synthetic images and assigns labels to them. However, the generated image resolution is only increased to $128 \times 128$. Self-attention generative adversarial networks (SA-CGAN) [20] improve the quality of CGAN-generated images by enhancing the relationships between image parts. Still, the semantic details of the pictures could not represent better when generating images with complex backgrounds.

Isola et al., based on the GAN idea, proposed the Pixel-to-pixel (Pix2pix) [15] network, which has good graph generation quality. However, the scarcity of paired data sets leads the Pix2pix model to poor generalization applications and few application scenarios as a supervised style migration network.

This work proposes a critical image generation network model for high-voltage transmission line components to solve the problem based on an improved generative adversarial network. The model in this paper can flexibly apply the image generation task of critical components of high-voltage transmission lines in complex backgrounds and other application scenarios. The CFM-GAN model consists of two generators and several discriminators. In Figure 1, the model in this paper produces images in two stages. Firstly, the global generator is responsible for extracting high-level abstract semantic features such as skeleton and texture to obtain low-resolution images (LR image). Secondly, the local generator is responsible for extracting the underlying basic features such as image resolution and degree of fineness to obtain high-resolution images (HR image).

![Figure 1](image.png)

**Figure 1.** Test results of the CFM-GAN model on KCIGD and CPLID, first with a global generator to acquire low-resolution images then with a local generator to obtain high-resolution images.

The main distinctions between CFM-GAN and conventional models include: firstly, CFM-GAN uses a multilevel generator, which we use to ensure that CFM-GAN produces high-resolution and realistic images on the one hand, and to provide its application to areas such as image reconstruction, image synthesis, and image translation on the other. Secondly, we use Monte Carlo search (MCS) [21] to sample the global generator-generated target image several times and compute the appropriate penalty values to obtain richer semantic information about the image. The penalty values could then guide the local generator to generate a semantically richer target image to prevent the pattern collapse problem [22].
effectively. Thirdly, after optimizing the generator structure, the discriminator needs to improve the adversarial capability. This paper modifies the traditional single-layer network model to a multi-scale discriminator network with three layers having the same design based on the parameter sharing [23] strategy and the feature pyramid network (FPN) [24]. The multi-scale discriminator network adopts three structurally identical, responsible for different levels of abstraction, semantics to commonly improve the discriminator’s discriminate ability. The above three modifications enable the network to pay attention to the high-level semantics of the generated image while also considering its texture details. Of course, these modifications make the model not need to perform complex loss function settings and subsequent processing operations while generating high-resolution semantic information. In addition, to solve the problem of a small public data set and the simple image backgrounds of the critical components of high-voltage transmission lines, we constructed the dataset KCIGD by using aerial images of crucial parts of high-voltage transmission lines taken by UAVs.

We compare and analyze CFM-GAN with mainstream image generation models and demonstrate that this model could produce high-resolution and high-fidelity images. The significant contributions to the work are listed below.

• In the current work, we propose an improved generation adversarial network model (CFM-GAN) consisting of two generators and several discriminators to generate images containing key components of high-voltage transmission lines. In addition, this paper adopts a multilevel generator combining a global generator and a local generator to ensure that CFM-GAN can produce high-resolution images with rich details and textures and increase the application scenarios of the model.

• To effectively search the intermediate generation state of the generated image, we introduce a penalty mechanism based on the Monte Carlo search in the generator to score the intermediate generation state of the generator. The introduction of the tool guides the generator to acquire richer semantic information. It enables the generated images to include more semantic details.

• Based on the mechanism of the argument shared, this paper proposes a multi-scale discriminant architecture. This structure can determine if the input picture is original or generated by using information from different levels of abstraction.

• This paper considers that there is currently no complete publicly available dataset of critical components of transmission lines, thus we establish a data set named KCIGD for generating high-quality images of essential components of high-voltage transmission lines. In addition, this paper conducted a comparative experiment between the CFM-GAN model and the current mainstream model on the KCIGD data set. The experiment proved the effectiveness and extensibility of the CFM-GAN model.

The rest of the arrangements for this article are below. The second part illustrates the area relating to image generation. Section 3 introduces the GAN. The CFM-GAN model is elaborated on in Section 4. Section 5 provides an experimental comparison to validate the model’s validity in this paper. The final section summarizes the paper’s content and our future research plans.

2. Related Work

Various deep learning-based generative models aim to produce image samples that the naked eye cannot distinguish between real and fake. Development trends in image generation models indicate that techniques such as autoregressive, VAE, flow-based, and GAN models are developing and growing.

The autoregressive model was introduced into the image generation task by Oord in 2016. On this basis, Oord proposes the pixel recurrent neural networks (Pixel RNN) [25] and pixel convolutional neural networks model (Pixel CNN) [26]. Among them, Pixel RNN obtains the corresponding pixel values by scanning each pixel in the image and then uses these values to predict the distribution of each pixel value in the image, thus achieving the purpose of image generation. The Pixel CNN captures the distribution of the
dependency between image pixels through network parameters and then generates pixels in the image in turn along two spatial dimensions, and thus uses the mask convolution layer to learn the distribution of all pixels in an image in parallel, and, finally, generates related images. Compared with Pixel RNN, Pixel CNN effectively reduces the number of network parameters and computational complexity. Still, because it limits the size of the receptive field during image generation, the image quality generated by Pixel CNN could not be satisfactory. Parmar et al. [27] significantly broadened the vision of image generation by introducing the self-attention mechanism into the Pixel CNN, thus considerably improving the quality of images. The sub-pixel network proposed by Menick [28] is to improve the resolution of the generated image by converting the basic unit of the generated image into image blocks. Chen et al. [29] solved the problem of short dependence duration in recurrent neural networks. They effectively improved the quality of the generated images by introducing causal convolutions into the self-attention mechanism.

VAE learns the probability distribution of data mainly by the maximum likelihood criterion. Gregor et al. [30] combined the spatial attention mechanism and sequential VAE to propose the deep recurrent attentive writer (DRAW) model to enhance the resulting image performance. Wu et al. [31] integrated the multiscale residual module into the adversarial VAE model, effectively improving image generation capability. Parmar et al. [32] proposed a generative automatic encoder model called dual contradistinctive generative autoencoder (DC-VAE). DC-VAE can synthesize images at different resolutions. Moreover, introducing the double contradiction loss function makes the images generated by DC-VAE significantly improved in quality and quantity. Hou et al. [33] proposed an improved variational autoencoder to force it to generate clear and realistic images by using the depth feature consistency principle and the generative adversarial training mechanism. The introspective variational autoencoder (IntroVAE) [34] uses adversarial training VAE to distinguish original samples from generated images. IntroVAE presents excellent image generation ability. Additionally, to ensure the stability of model training, it also adopts hinge-loss terms for generated samples. The $\beta$-VAE framework joint distribution of continuous and discrete latent variables (Joint-VAE), proposed in [35], can jointly learn the generative distributions of constant and discrete latent variables based on supervised categorical labeling and, finally, categorize the generated images effectively. Moreover, since the encoder can effectively predict the labels, the framework can still use tags for a conditional generation even without a classifier.

The main ideas of the generative flow-based model were originally from NICE [6], RealNVP [7], and GLOW [8]. RealNVP has made multiple improvements based on NICE, including replacing the additive coupling layer with the affine coupling layer to improve the fitting ability of the transformation. They introduced the checkerboard mask convolution technique to process image data. The multi-scale structure is adopted to learn the characteristics of different scales and randomly scrambles the channel behind the coupling layer to effectively fuse the information between different channels to improve the model’s performance. Based on RealNVP, the GLOW network further improves the method of random channel disruptions, replacing it with a $1 \times 1$ convolution layer to effectively integrate the information of different channels and significantly enhance the quality of image generation. Compared with GAN and VAE, the generative flow-based model can generate higher-resolution images and accurately infer hidden variables. In contrast to autoregression, the flow model can carry out a parallel computation and efficiently carry out data sampling operations. However, the flow model is also accompanied by complex computational problems while generating high-quality images.

GAN has attracted the attention of many researchers because of its vast application potential. The fusing of GAN and CGAN (Fused-GAN) is presented by Bodla et al. [36]. Fused-GAN generates image samples of diversity and high similarity by combining two generators. The first generator unconditionally generates feature maps; the second uses feature maps and conditional inputs to generate final image samples. The authors of [37] constructed a cascaded GAN structure by introducing a Laplace pyramid structure, progressively improving the resolution of these generated images by learning the residuals
between adjacent layers. However, it still requires a deeper network structure and more computing resources. KARRAS et al. [38], by adding latent codes and random noise to each network layer, a style-based generator architecture for the generative adversarial network (Style-GAN) can produce higher-quality images with more apparent details and style diversity than other GAN networks, but only for small image datasets. However, the resulting images may be artifacts or texture sticking for complex image data sets (such as ImageNet). Self-attention generative adversarial nets (SAGAN) [20] address the problem of poor generation when dealing with complex structure images by introducing a self-attention mechanism in the network for the feature extraction and learning of crucial image regions. A generative model from a single natural image (Sin-GAN) performs multi-granularity learning by down-sampling a single image at different sizes, enabling the model to generate various high-quality images at arbitrary sizes [39].

Although previous studies have achieved good experimental results in different image generation tasks, considering the unique geographical location of the transmission lines makes the image backgrounds obtained from UAV aerial photography extremely complex. The complex environment leads to poor compatibility between the above image generation model and the image with the transmission line background. After all, not all scenes have enough semantic information for model evaluation, and not all models can match UAVs and other equipment. This results in the above generation models still having difficulty balancing the quality of the generated images with the generation speed. The proposed CFM-GAN model in this paper is also inspired by [40], which progressively generates transmission line images based on GAN networks. Our model further improves the problem of low resolution and poor semantic detail representation of the generated images by introducing a penalty mechanism to do high-resolution constraints and a semantic information guide to the generator.

3. Basic Knowledge of GAN

The process of proposing a GAN is heavily influenced by the “two-person zero-sum game theory,” which consists of a generator and discriminator models. The generator generates as realistic as possible image data according to the noise vector. The discriminator distinguishes the generated data from the actual image data as accurately as possible. In the confrontation, the two fight each other and progress until they reach the optimal state of Nash equilibrium [41]. The objective loss function for generating adversarial network training is:

$$\min_G \max_D V_{GAN}(G,D) = \mathbb{E}_{x \sim p_r(x)} [\log D(x)] + \mathbb{E}_{z \sim p_g(z)} [\log (1 - D(G(z)))]$$  \hspace{1cm} (1)

where \(G\) denotes the generative network, \(D\) represents the discriminative network, \(x\) indicates the sample data, and \(z\) represents random noise. \(p_r\) and \(p_g\) mean the probability distribution of the sample data and the distribution of the generated data, respectively.

This paper, furthermore, considers that the GAN loss function encourages the generation of images with more color information and that traditional loss functions (such as the \(L_1\) or \(L_2\) separation from the original image and the resulting image) encourage the generation of an image with more detailed information, such as image edges. The CFM-GAN model combines the GAN loss function with the traditional loss function. However, it has been shown that the \(L_1\) distance can be more effective in reducing the blurring of the generated image compared to the \(L_2\) distance [15]. Hence, we incorporate the \(L_1\) distance into the CFM-GAN network model. The \(L_1\) distance equation is as follows.

$$\min_G V_{L1}(G) = \mathbb{E}_{x,x' \sim p_r(x,x')} [\| x - x' \|_1]$$  \hspace{1cm} (2)

The final loss function for CFM-GAN is expressed as:

$$G^* = \arg\min_G \max_D V_{GAN}(G,D) + \lambda V_{L1}(G)$$  \hspace{1cm} (3)
\( \lambda \) represents the numerical of the weight set.

4. CFM-GAN

The configuration of CFM-GAN is illustrated in Figure 2. This study aims to produce high-resolution images of critical components of high-voltage transmission lines with evident details and features. To this end, in the CFM-GAN model, the generator \((G)\) uses the sample image \((I_o)\) to generate images \((I_g)\) as similar as possible to the sample to deceive the discriminator. The discriminator \((D)\) distinguishes the sample image \((I_o)\) from the generated image \((I_g)\) as accurately as possible. The \((S)\) represents the search. The discriminator searches the intermediately generated state of the image based on the Monte Carlo search to obtain the final penalty value. The penalty value can guide the generator’s generation direction faster and better by perceiving the generator’s generation strategy in advance. Then, the generator also updates the generator’s parameters according to the penalty value \((V_D^G)\) until the generator and discriminator reach Nash equilibrium.

![Figure 2. The framework of CFM-GAN.](image)

4.1. Multi-Level Generator

In a conventional generative adversarial network, the generator usually consists of only one component. To effectively improve the quality of image generation, as shown in Figure 3, this chapter changes the structure of the generator network to consist of global generator \(G_1\) and local generator \(G_2\), among which the local generator’s purpose is to improve the resolution of the generated image. The image input resolution of the global and local generators is \(256 \times 256\) and \(512 \times 512\), respectively. The global generator network comprises a pre-convolution layer, an intermediate residual block, and a back-end convolution layer. Similarly, the local generator is also composed of these three parts. However, before the image is input into the local generator’s residual block, the global generator’s output and the result of the pre-convolution layer of the local generator is superimposed. This way, the whole generation network outputs images with a high-resolution and rich semantic details. This paper trained the local generator and the global generator jointly. The local generator needs the feature results of the global generator as a part input of the training of residual error and back-end convolution model; thus, the original image is first down-sampled by a factor of two and then as input to the global generator. After finishing the global generator’s training, the local generator is trained with the result of forward propagation.

By training a local generator, this paper can extract basic features, such as the resolution and semantic details of the image, to produce a high-resolution target image. We use the Monte Carlo search strategy to mine hidden spatial semantic information from low-resolution image samples to further improve image resolution and semantic information. Additionally, the punishment mechanism and multi-channel attention mechanism are then combined to effectively constrain and guide the generation of image semantic details to output high-resolution target images with rich semantic information.
down-sampled by a factor of two and then as input to the global generator. After finishing the global generator’s training, the local generator is trained with the result of forward propagation. By training a local generator, this paper can extract basic features, such as the resolution and semantic details of the image, to produce a high-resolution target image. We use the Monte Carlo search strategy to mine hidden spatial semantic information from low-resolution image samples to further improve image resolution and semantic information. Additionally, the punishment mechanism and multi-channel attention mechanism are then combined to effectively constrain and guide the generation of image semantic details to output high-resolution target images with rich semantic information.

Figure 3. The multi-level generator consists of MoCo feature extraction, a global generator, a Monte Carlo strategy, and a local generator. Firstly, MoCo extracts the image’s feature and fed into the global generator’s residual layer. Secondly, the global generator is employed to generate LR. Next, the Monte Carlo searching strategy is used to explore the hidden spatial contents in LR. Finally, the multi-channel attention mechanism feeds the mined semantic details to the local generator for generating HR.

4.1.1. Penalty Mechanism

The resolution of the generated image has improved through the secondary network structure combining the local and global generators. At present, many researchers have adopted similar methods to improve the quality of the generated image. However, to introduce more semantic information guidance in the image generation process and improve image resolution, we propose a penalty mechanism based on the Monte Carlo search (MCS) strategy, which can mine the hidden information in the image sample space to generate high-resolution images with rich semantic detail information.

To this end, we use the Monte Carlo search to perform an intermediate generated state search on the results of the global generator $G_1$ as follows:

$$ \left\{ I_{g}^{1}, \ldots, I_{g}^{N} \right\} = MC_{G_{\beta}}(I_{g}^{\prime}, N) $$  \hspace{1cm} (4)

$$ I_{g}^{i} = G_{\beta}(I_{g}^{\prime}, z) $$  \hspace{1cm} (5)

where $I_{g}^{i} (i = 1, \ldots, N)$ denotes the corresponding generated state obtained after a collection of the intermediate state, $MC_{G_{\beta}}$ indicates the hidden space state obtained by performing a Monte Carlo search of the sample space. Additionally, $G_{\beta}$ denotes the virtualization
generation module obtained by the Monte Carlo search technique and the sharing parameters with a global generator, from which comes the generation of N intermediate result images. In addition, this paper introduces the noise vector into the model training to ensure the diversity of low-resolution image sampling. Additionally, the vector of noise vectors \( z \sim N(0, 1) \) and the global generators’ output are fed together into the local generator. Introducing noise vectors allows the sampling network to pay different attention to different feature vectors and obtain richer semantic messages derived from the low-resolution generated images.

Next, the sampled N intermediate states and the images generated by the global generator are sent to the discriminator. Additionally, a penalty value is then calculated through the output of the discriminator. The calculation process is as follows:

\[
V^D_G = \frac{1}{N} \sum_{i=1}^{N} \left( (1 - D(i^{\text{g}}_i; \theta_d)) + \|I^{\text{g}}_i - I^{\text{t}}_i\|_1 \right)
\]  

(6)

where \( D(i^{\text{g}}_i; \theta_d) \) denotes the possibility that this output image \( i^{\text{g}}_i \) of the generator is the original image.

4.1.2. Attention Mechanism

First, to supply enough semantic information to guide the local generator to generate images, we sampled the low-resolution images \( N \) times by the Monte Carlo search for the intermediate result images. Second, previous studies generated images through only three RGB channels. Here, to construct a larger semantic generation space and extract richer semantic information, we randomly sampled the generated intermediate result images through the multi-channel attention mechanism to obtain N intermediate results. We used the final feature set and made a finer-grained extraction for information from different channels. Finally, the feature maps extracted by the multi-channel attention mechanism were fed into the local generator model to gain high-resolution images with detailed characteristics that were evident.

Next, to establish a more effective channel correlation and extract semantic information from the N intermediate result images, we performed a convolution operation on these medium result images, computed as follows.

\[
I^{\text{g}}_i = \text{Softmax}(W^i_A + b^i_A), \text{ for } i = 1, \ldots, N
\]  

(7)

where \( \{W^i_A\}_{i=1}^{N} \) denote the convolution kernel parameters, \( \{b^i_A\}_{i=1}^{N} \) indicates the bias, and the Softmax function \( \text{Softmax}(\cdot) \) obtains the attention weight matrix to give higher weight to important pixel points, smaller weights to irrelevant pixel points, and enhance useful features while suppressing irrelevant ones. We weighed \( \otimes \) and summed \( \oplus \) the attention weight matrix and the sampled N intermediate result images as the final output result \( I'_{\text{g}} \).

\[
I'_{\text{g}} = (I^1_A \otimes I^1_\text{g}) \oplus \cdots \oplus (I^N_A \otimes I^N_\text{g})
\]  

(8)

4.1.3. Objective Function

Continuous optimization of multigrain generators by the minimization of \( J_G(\theta_g) \).

\[
J_G(\theta_g) = \mathbb{E}_{I_\text{o} \sim P(I_\text{o}), I_\text{t} \sim P(I_\text{t})} [\log(1 - D(I_\text{o}, G(I_\text{o}, I_\text{t})))] + V^D_G
\]  

(9)

where \( D(I_\text{o}, G(I_\text{o}, I_\text{t})) \) equals to the fraction of these resulting images derived from the generator \( G \) with raw images and noise vectors, put another way, and it represents the possibility that this discriminator \( D \) will identify its produced images as the original image. \( \theta_g \) defines the parameters of the generator. \( V^D_G \) means the penalty value obtained from Formula’s (6) calculation of the generated image.
4.2. Multi-Level Discriminator

The architecture of the distinguisher network constructed in this paper is shown in Figure 4. The essential components of this discriminator network are deep convolutional networks with simple architectures. If we blindly deepen or expand the model to increase the discriminatory power of the discriminator, it will result in a dramatic growth of the model parameters. Additionally, it will then cause a data drag on the network training and testing images. Therefore, we present a multi-scale discriminator network with the same structure and different processing scales.

![Figure 4. The discriminator consists of a triple network with an identical configuration and different processing scales. The feature maps are first obtained by feature extraction of the original image using shared convolutional layers. Then, the feature maps are down-sampled by 2x and 4x and are sent into D1, D2, and D3, respectively, which can obtain the discriminator’s final score.](image)

### 4.2.1. Multitasking Mechanism

In this paper, we propose a multiscale discriminator network. The network mainly uses a multi-task acquisition policy to evaluate the discriminators based on the network parameter-sharing mechanisms. Taking the feature pyramid network as a thought, we modify the traditional single-layer network structure model into a three-layer structure, denoted by \( D_1 \), \( D_2 \), and \( D_3 \), respectively. Firstly, the three discriminators use the same shared-convolutional layer to extract the raw image’s features and the generated images’ features. Additionally, the discriminators then obtain their corresponding feature maps. Then, the obtained feature maps are down-sampled by 2x and 4x, and the obtained different scales of characteristic images are fed to these three discriminators. Finally, the three discriminators are used in this paper to process the characteristic images at various scales, respectively.

It is worth mentioning that the multiscale discriminators have the same structure. The three layers of discriminators are responsible for different levels of semantic abstraction. Feature information processed by the identifier at a reduced input scale enhances the abstract semantic details of the resulting images. The featured message processed by the discriminator could promote the abstract semantic details of the generated images. Additionally, the discriminator processed with a larger input ratio could enhance the detailed texture details of the produced images. Therefore, the structure of the multiscale discriminator is more conducive to the training of CFM-GAN. When we need to generate high-resolution image samples, we incorporate discriminators into the CFM-GAN model rather than building the CFM-GAN model from scratch, which significantly improves the scalability of the CFM-GAN model.
4.2.2. Objective Function

The process of the multi-task learning mechanism can be expressed as:

\[ J_D(\theta_d) = - \sum_{k=1,2,3} \mathbb{E}_{I_o \sim P_r(I_o), I_g \sim P_g(I_g)} \log D_k((I_o, I_g); \theta_d) \] (10)

where \( I_o \) and \( I_g \) are equal to the original image and generated image, respectively, and \( P_r \) and \( P_g \) represent the probability distributions of the raw picture and its resulting image, respectively. \( \theta_d \) indicates the experimental parameters of the discriminator. \( D_k \) denotes one of the three discriminators of \( D_1 \), \( D_2 \), and \( D_3 \).

At this point, the CFM-GAN model performs adversarial learning between multi-granularity generators and multiscale discriminators until a Nash equilibrium is reached. The training algorithm of the model is presented by Algorithm 1.

**Algorithm 1.** The training procedure of CFM-GAN

Input: Original image \( I_o = \{I_{o1}, \cdots, I_{oN}\} \) of crucial components of high-voltage transmission lines; Generator \( G \); Multiscale discriminator \( \{D_i\}_{i=1}^{k} \); g-steps, the number of the generator’s training steps; d-steps, the number of discriminators’ training steps.

Output: The trained generator \( G \).

1: Initialize coarse-grained generator \( G \) and multiscale discriminator \( \{D_i\}_{i=1}^{k} \) with random weights;
2: repeat
3: for g-steps implement
4: The generator \( G \) produce fake images;
5: The penalty value \( V_G \) of fake image was calculated by Formula (6);
6: Optimize and update the generator \( G \) parameters by minimizing Formula (9);
7: end for
8: for d-steps implement
9: The generator \( G \) is used to produce fake images \( I_g = \{I_{g1}, \cdots, I_{gN}\} \);
10: Based on the original images \( I_o = \{I_{o1}, \cdots, I_{oN}\} \) and the generated images \( I_g = \{I_{g1}, \cdots, I_{gN}\} \), the parameters of the multiscale discriminator are updated via minimizing Formula (10);
11: end for
12: until CFM-GAN completes convergence
13: return

4.3. Network Structure

Generating images for critical components of high-voltage transmission lines may share a great deal of potential feature information between the original image and the generated image. To make the underlying information quickly and stably transmitted in the network, our generator and discriminator use U-net [42] as their basic architecture. It is worth mentioning that U-net uses its encoder to encode and extract the input image features to obtain the abstract features of the image, and U-Net then uses its decoder to restore the extracted feature information to the original image size.

To enhance the stability of the model expression and generate higher-quality images in this paper, we used Momentum contrast (MoCo) [43] to extract the features of the images before sending the original images into the generator. MoCo randomly clips any two images (48 × 48) in the same image (512 × 512), and the image (key) remains unchanged while the other image (query) is randomly sampled. MoCo will generate different vectors for these two images, then compress these vectors’ information using the attention mechanism, and, finally, transform the image information into a two-dimensional vector. After that, the vector goes through the linear layer and dimension extension. Finally, the feature information is fed into the model as the generator’s residual layer’s convolution parameter.

\[ \theta_k \leftarrow m\theta_k + (1 - m)\theta_q \] (11)
where \( m \) belongs to \((0,1)\) as a momentum coefficient, and only the parameter \( \theta_k \) of the image (query) will be updated through backpropagation. In contrast, the current image (key) parameter is based on the previous \( \theta_k \) and the current \( \theta_q \) to achieve an indirect momentum update.

The CFM-GAN proposed in this paper consists of two generators and multiple discriminators. As indicated by Tables 1 and 2, the generator includes a down-sampling layer, a residual block layer, and an up-sampling layer. The down-sampling layer extracts features from the image by increasing the size of the perceptual field to obtain the feature map. The residual block layer enhances the feature map’s semantic information by retaining the feature map’s essential information. The up-sampling layer generates the target image with the extracted abstract semantic information.

Before feeding into the discriminator, the image of the crucial components of the high-voltage transmission line used in this paper and the primary features of the original image are, first, extracted by a convolution kernel with a \( 3 \times 3 \) size, step size 1, and edge padding 1. After obtaining feature maps of the identical magnitude (512 \( \times \) 512) as the raw selection, the feature maps are down-sampled by \( 2x \) (256 \( \times \) 256) and \( 4x \) (128 \( \times \) 128), respectively, and the feature images of various dimensions are fed to the discriminators of the corresponding dimensions. As shown in Table 3, the first layer of the discriminator uses the Leaky Relu [44] activation function without normalization. The final layer of the discriminator also uses a wholly attached version of the filter to produce a one-dimensional output. It is worth mentioning that the identifier’s network architecture is the same even if our input is of a different resolution.

### Table 1. The system architecture of the local generator.

<table>
<thead>
<tr>
<th>Network</th>
<th>Layer Information</th>
<th>Input</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pre-Convolutional</td>
<td>CONV-(N32, K7x7, S1, P3), Leaky ReLU</td>
<td>(512, 512, 3)</td>
<td>(256, 256, 32)</td>
</tr>
<tr>
<td></td>
<td>CONV-(N64, K3x3, S2, P1), IN, ReLU</td>
<td>(256, 256, 32)</td>
<td>(128, 128, 64)</td>
</tr>
<tr>
<td></td>
<td>CONV-(N64, K3x3, S1, P1), IN, ReLU</td>
<td>(128, 128, 64)</td>
<td>(128, 128, 64)</td>
</tr>
<tr>
<td>Residual Block</td>
<td>CONV-(N64, K3x3, S1, P1), IN, ReLU</td>
<td>(128, 128, 64)</td>
<td>(128, 128, 64)</td>
</tr>
<tr>
<td></td>
<td>CONV-(N64, K3x3, S1, P1), IN, ReLU</td>
<td>(128, 128, 64)</td>
<td>(128, 128, 64)</td>
</tr>
<tr>
<td></td>
<td>CONV-(N512, K3x3, S0.5, P1), IN, ReLU</td>
<td>(256, 256, 64)</td>
<td>(256, 256, 512)</td>
</tr>
<tr>
<td>Up-Sample</td>
<td>CONV-(N3, K7x7, S1, P3), IN, ReLU</td>
<td>(256, 256, 512)</td>
<td>(512, 512, 3)</td>
</tr>
</tbody>
</table>

### Table 2. The system architecture of the global generator.

<table>
<thead>
<tr>
<th>Network</th>
<th>Layer Information</th>
<th>Input</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pre-Convolutional</td>
<td>CONV-(N64, K7x7, S1, P3), IN, ReLU</td>
<td>(256, 256, 3)</td>
<td>(256, 256, 64)</td>
</tr>
<tr>
<td></td>
<td>CONV-(N128, K3x3, S2, P1), IN, ReLU</td>
<td>(256, 256, 64)</td>
<td>(128, 128, 128)</td>
</tr>
<tr>
<td></td>
<td>CONV-(N256, K3x3, S2, P1), IN, ReLU</td>
<td>(128, 128, 128)</td>
<td>(64, 64, 256)</td>
</tr>
<tr>
<td></td>
<td>CONV-(N512, K3x3, S2, P1), IN, ReLU</td>
<td>(64, 64, 256)</td>
<td>(32, 32, 512)</td>
</tr>
<tr>
<td></td>
<td>CONV-(N1024, K3x3, S2, P1), IN, ReLU</td>
<td>(32, 32, 512)</td>
<td>(16, 16, 1024)</td>
</tr>
<tr>
<td>Residual Block</td>
<td>CONV-(N1024, K3x3, S1, P1), IN, ReLU</td>
<td>(16, 16, 1024)</td>
<td>(16, 16, 1024)</td>
</tr>
<tr>
<td></td>
<td>CONV-(N1024, K3x3, S1, P1), IN, ReLU</td>
<td>(16, 16, 1024)</td>
<td>(16, 16, 1024)</td>
</tr>
<tr>
<td></td>
<td>CONV-(N1024, K3x3, S1, P1), IN, ReLU</td>
<td>(16, 16, 1024)</td>
<td>(16, 16, 1024)</td>
</tr>
<tr>
<td></td>
<td>CONV-(N1024, K3x3, S1, P1), IN, ReLU</td>
<td>(16, 16, 1024)</td>
<td>(16, 16, 1024)</td>
</tr>
<tr>
<td></td>
<td>CONV-(N1024, K3x3, S1, P1), IN, ReLU</td>
<td>(16, 16, 1024)</td>
<td>(16, 16, 1024)</td>
</tr>
<tr>
<td></td>
<td>CONV-(N1024, K3x3, S1, P1), IN, ReLU</td>
<td>(16, 16, 1024)</td>
<td>(16, 16, 1024)</td>
</tr>
<tr>
<td></td>
<td>CONV-(N1024, K3x3, S1, P1), IN, ReLU</td>
<td>(16, 16, 1024)</td>
<td>(16, 16, 1024)</td>
</tr>
<tr>
<td></td>
<td>CONV-(N1024, K3x3, S1, P1), IN, ReLU</td>
<td>(16, 16, 1024)</td>
<td>(16, 16, 1024)</td>
</tr>
<tr>
<td></td>
<td>CONV-(N1024, K3x3, S1, P1), IN, ReLU</td>
<td>(16, 16, 1024)</td>
<td>(16, 16, 1024)</td>
</tr>
<tr>
<td></td>
<td>CONV-(N1024, K3x3, S1, P1), IN, ReLU</td>
<td>(16, 16, 1024)</td>
<td>(16, 16, 1024)</td>
</tr>
<tr>
<td></td>
<td>CONV-(N1024, K3x3, S1, P1), IN, ReLU</td>
<td>(16, 16, 1024)</td>
<td>(16, 16, 1024)</td>
</tr>
<tr>
<td></td>
<td>CONV-(N1024, K3x3, S1, P1), IN, ReLU</td>
<td>(16, 16, 1024)</td>
<td>(16, 16, 1024)</td>
</tr>
<tr>
<td></td>
<td>CONV-(N512, K3x3, S0.5, P1), IN, ReLU</td>
<td>(64, 64, 256)</td>
<td>(32, 32, 512)</td>
</tr>
<tr>
<td></td>
<td>CONV-(N256, K3x3, S0.5, P1), IN, ReLU</td>
<td>(32, 32, 512)</td>
<td>(16, 16, 1024)</td>
</tr>
<tr>
<td></td>
<td>CONV-(N128, K3x3, S0.5, P1), IN, ReLU</td>
<td>(16, 16, 1024)</td>
<td>(16, 16, 1024)</td>
</tr>
<tr>
<td></td>
<td>CONV-(N64, K3x3, S0.5, P1), IN, ReLU</td>
<td>(16, 16, 1024)</td>
<td>(16, 16, 1024)</td>
</tr>
<tr>
<td></td>
<td>CONV-(N3, K7x7, S1, P3), IN, ReLU</td>
<td>(256, 256, 64)</td>
<td>(256, 256, 3)</td>
</tr>
</tbody>
</table>
Table 3. The system architecture of the discriminators.

<table>
<thead>
<tr>
<th>Network</th>
<th>Layer Information</th>
<th>Input</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input Layer</td>
<td>CONV-(N64, K4x4, S2, P2), Leaky ReLU (512, 512, 3)</td>
<td>(512, 512, 3)</td>
<td>(256, 256, 64)</td>
</tr>
<tr>
<td></td>
<td>CONV-(N128, K4x4, S2, P2), IN, ReLU (256, 256, 64)</td>
<td>(256, 256, 64)</td>
<td>(128, 128, 128)</td>
</tr>
<tr>
<td></td>
<td>CONV-(N256, K4x4, S2, P2), IN, ReLU (128, 128, 128)</td>
<td>(128, 128, 128)</td>
<td>(64, 64, 256)</td>
</tr>
<tr>
<td></td>
<td>CONV-(N512, K4x4, S2, P2), IN, ReLU (64, 64, 256)</td>
<td>(64, 64, 256)</td>
<td>(32, 32, 512)</td>
</tr>
</tbody>
</table>

5. Experiments and Analysis

Firstly, we need an image database of critical components of high-voltage transmission lines to help us analyze the CFM-GAN module’s performance. The only public dataset for high-voltage transmission line critical components is CPLID [45]. It is worth mentioning that the images in CPLID are obtained mainly by cropping, flipping, and stitching with the image’s background; thus, the image background is simple and does not fully satisfy the application of the image generation model in real scenarios. Therefore, this paper uses an aerial video of a 500 kV high-voltage transmission line from a UAV in China as a data source and generates a dataset for generating key components images of transmission lines. Transmission lines are generally located at high altitudes and widely distributed in mountains, forests, rivers, lakes, fields, hills, and other areas. Therefore, the final image obtained has a complex background and contains more semantic detail to evaluate the model’s performance more effectively. Both the homemade dataset and the public dataset CPLID contain vital components of transmission lines, such as insulators, anti-vibration hammers, spacer bars, lightning rods, and towers. In addition, we named the original image dataset KCIGD, and the training set and test set contain 4200 and 700 original images, respectively.

The training in this study mainly consists of two stages: freezing and thawing. Because the image-generation model is pre-trained, its parameters also have a certain priority. In model training, to prevent the generator network parameters from running aimlessly, we freeze and thaw them. The forward propagation calculation will be carried out quickly in the freezing process using these pre-trained parameters. After the generator model is roughly trained, we will perform the thawing operation on the generator model parameters. The network model will guide the generator’s generation direction during the unfreezing process according to the discriminator’s scoring output. The generator will adjust the parameters’ optimization direction according to the discriminator’s scoring result. The network is slowly trained to generate images with as much semantic detail and as high of a resolution as possible. All parameters of the generated adversarial network used in the study were performed with the Adam optimizer. We performed 200 rounds of training, where the learning rate were kept constant at 0.0002 for the first 100 times and were reduced progressively to zero for the following 100 times. The initial weight coefficients were Gaussian distributed, and the mean was 0 with a standard deviation of 0.02. A NVIDIA RTX 2080 GPU with 8 GB of memory was employed to achieve the training and testing. The experimental operating system was Ubuntu 18.04 with 32 GB of memory. Additionally, all algorithms were built based on Pytorch 1.4. The amount of Monte Carlo searches applied in this study was 5, and the specific weight of the loss regarding the control discriminator and diagram extractive match was 10.

5.1. The Baselines

In this paper, to facilitate testing the performance of the CFM-GAN model, we compare and analyze CFM-GAN with the current mainstream image generation models.

VAE [46]: A popular generation model consists of a differentiable encoder and a decoder. VAE trains two networks by optimizing the variational boundary of the logarithmic likelihood of data. As a result of the addition of noise and the use of inappropriate element distance measures (such as a squared error), there are fuzzy problems in the samples generated by VAE.
Cascaded refinement network (CRN) [47]: Unlike the GAN method in image generation, the CRN does not rely on adversarial training. The CRN model uses an end-to-end convolutional network to generate corresponding images according to the input pixel semantic layout images. CRN creatively uses techniques of calculating matching losses between images, which compute different losses in the generated and semantically segmented images.

The combination of variational auto-encoders and generative adversarial nets (VAE-GAN) [48]: In the VAE model, the reduction factor for network optimization is the Euclidean difference from the initial image after decoding by the decoder. However, the loss value is not precisely inversely proportional to the image quality, thus the decoded image is then delivered to the discriminator to identify its generation effect, thus achieving the effect of using GAN to enhance the VAE image generation effect.

Pix2pix [15]: The Pix2pix model is based on an adversarial loss implementation. In short, the network learning mapping from Pixel x to Pixel y has achieved good results in tasks such as image translation and pixel transfer.

InsulatorGAN [49]: In this model, the insulator label box is restricted in the image generation process by combining the coarse–fine granularity model, which can detect insulator segments as accurately as possible.

5.2. Quantitative Evaluation

In this chapter, the resulting image qualities are first evaluated using the inception score (IS) and Fréchet’s inception distance (FID), which are specific evaluation metrics based on generative adversarial networks. Then, the pixel-level evaluation metrics [50] in structural similarity (SSIM), peak signal-to-noise ratio (PSNR), and sharpness difference (SD) are referred to, to determine the degree of resemblance of the original and generated images.

5.2.1. Inception Score (IS) and Fréchet’s Inception Distance (FID)

The IS index calculates the distance between the two probability distributions by KL divergence, which reflects the fit of the generated image’s probability distribution related to the actual image probability distribution to a certain extent. The greater the IS score, the greater the clarity of the generated images and the better the overall diversity. The IS is calculated as below.

$$\text{IS}(G) = \exp \left( \mathbb{E}_{x \sim p_g} D_{KL}(p(y|x) \parallel p(y)) \right)$$  \hspace{1cm} (12)

where image $x$ is sampled from the distribution $p_g$ of generated data, and $D_{KL}$ represents the distance in the middle of the generated image distribution and the original image distribution, i.e., the relative entropy.

Since the IS score uses the InceptionV3 network trained under the ImageNet dataset, the transmission line critical components included in the public dataset CPLID and the dataset KCIGD in this paper are not in the ImageNet dataset [51]. Therefore, we scored the CFM-GAN model based on the AlexNet network.

Fréchet’s inception distance (FID) can be used to calculate the Fréchet distance between the actual and generated samples in a Gaussian feature space distribution. A minor FID score indicates that the resulting images are closer to the original images and the resulting images are more realistic. The FID is calculated as follows.

$$\text{FID}(I_t, I_g) = \left\| \mu_t - \mu_g \right\|^2 + T_r \left( \sum_{l_t} \sum_{l_g} - 2 \left( \sum_{l_t} \sum_{l_g} \right)^{\frac{1}{2}} \right)$$  \hspace{1cm} (13)

where $\mu_t$ and $\mu_g$ represent the feature mean value of the original image and the generated image, $\sum_{l_t}$ and $\sum_{l_g}$ are the covariance matrix of the original image feature vector and the generated image feature vector. $T_r$ is the “trace” operation in linear algebra. This indicator also with the help of the InceptionV3 network, but the difference is that FID only uses the Inception network as a feature extractor. We also use AlexNet to extract its features, then
map the feature map to a $1 \times 1 \times 4096$ vector through the fully connected layer, and, finally, receive a $4096 \times 4096$ covariance matrix.

Table 4 lists the experimental results of the comparative analysis of CFM-GAN and current mainstream models using IS and FID metrics. The experiments indicate a higher score for the CFM-GAN model than for the other structures, i.e., the CFM-GAN model slightly outperforms the alternatives concerning the resulting image qualities and types.

<table>
<thead>
<tr>
<th>Model</th>
<th>KCIGD IS</th>
<th>CPLID IS</th>
<th>KCIGD FID</th>
<th>CPLID FID</th>
</tr>
</thead>
<tbody>
<tr>
<td>GAN</td>
<td>3.50</td>
<td>59.08</td>
<td>3.32</td>
<td>60.01</td>
</tr>
<tr>
<td>VAE</td>
<td>3.08</td>
<td>59.54</td>
<td>2.96</td>
<td>60.43</td>
</tr>
<tr>
<td>CRN</td>
<td>3.25</td>
<td>59.27</td>
<td>3.14</td>
<td>60.14</td>
</tr>
<tr>
<td>Pix2pix</td>
<td>3.33</td>
<td>58.64</td>
<td>3.42</td>
<td>59.58</td>
</tr>
<tr>
<td>VAEGAN</td>
<td>3.95</td>
<td>57.84</td>
<td>3.83</td>
<td>58.62</td>
</tr>
<tr>
<td>InsulatorGAN</td>
<td>4.05</td>
<td>55.23</td>
<td>3.98</td>
<td>56.83</td>
</tr>
<tr>
<td>CFM-GAN</td>
<td>4.23</td>
<td>54.98</td>
<td>4.15</td>
<td>56.69</td>
</tr>
</tbody>
</table>

5.2.2. Peak Signal-to-Noise Ratio (PSNR), Structural Similarity (SSIM), and Sharpness Difference (SD)

Peak Signal Noise Ratio (PSNR) evaluates an image by comparing the error between the original image and the corresponding pixels of the generated image. The unit is dB. The greater the PSNR values, the less distorted the generated images are and the more influential the results.

$$PSNR(I_g, I_t) = 10 \log_{10} \left( \frac{\max_{g} I_g}{\text{mse}(I_g, I_t)} \right)$$ (14)

$$\text{mse}(I_g, I_t) = \frac{1}{n} \sum_{i=1}^{n} (I_t[i] - I_g[i])^2, \max_{g} I_g = 255.$$ (15)

SSIM measures the similarity between images from three aspects: brightness, contrast, and structure. The closer the value of SSIM is to 1, the more similar the processed image structure is to the original image, i.e., the better the resulting image is.

$$SSIM(I_g, I_t) = \frac{(2\mu_I + c_1)(2\sigma_{IgIt} + c_2)}{(\mu_I^2 + \mu_I^2 + c_1)(\sigma_{Ig}^2 + \sigma_{It}^2 + c_2)}.$$ (16)

where $\mu_I$ and $\mu_I$ represent the mean of the images $I_g$ and $I_t$, respectively. $\sigma_{Ig}$ and $\sigma_{It}$ represent the standard deviation of images $I_g$ and $I_t$. $\sigma_{IgIt}$ denotes the covariance of the image $I_g$ and $I_t$. $c_1$ and $c_2$ are constants.

This study calculated the loss of sharpness among the generated and original images, drawing on the concept of gradient difference to measure the sharpness of the generated image.

$$\text{SharpDiff.}(I_g, I_t) = 10 \log_{10} \left( \frac{\max_{g} I_g}{\text{grads}} \right)$$ (17)

$$\text{grads} = \frac{1}{N} \sum_{i} \sum_{j} \left( ||\nabla I_g + \nabla I_t|| - (\nabla I_g + \nabla I_t) \right)$$ (18)

where $\nabla I = |I_{ij} - I_{i,j-1}|, \nabla I = |I_{ij} - I_{i,j-1}|$.

As seen in Table 4, the CFM-GAN score is higher than other mainstream models. This result indicates that the CFM-GAN model can generate high-resolution images of crucial components of high-voltage transmission lines and can also be well applied to scenes with complex image backgrounds such as mountains, lakes, and forests.
As seen in Table 4, the CFM-GAN score is higher than other mainstream models. This result indicates that the CFM-GAN model can generate high-resolution images of crucial components of high-voltage transmission lines and apply them to scenes with complex image backgrounds, such as mountains, lakes, and forests. The image quality generated by the model in this paper is better than that of InsulatorGAN. Analyzing the reason, MoCo plays a positive role in image feature extraction on the one hand. On the other hand, InsulatorGAN uses the punishment mechanism to constrain the generation of an insulator box. In this study, the proposed penalty mechanism can mine the hidden spatial semantic data inside the images, guide the local generator to produce a more realistic image, and make the final generated image more similar to the original image. In addition, as the similarity in value was high between the generated images and the authentic images, the images generated in this paper can also expand the images of critical components of high-voltage transmission lines.

To fully evaluate the behavior of the models, we also performed speed tests on different models; we also performed speed tests on other models. The tested speed of the CFM-GAN models in our paper is less than that of the current mainstream models, as shown in Table 5. This is because the CFM-GAN model combines global and local generators. The Monte Carlo search strategy sampling process also takes significant time. However, the number of 62 FPS is sufficient for real-world applications.

Table 5. SSIM, PSNR, and SD of various models. FPS is the number of images that are processed per second during the test.

<table>
<thead>
<tr>
<th>Model</th>
<th>KCIGD</th>
<th>CPLID</th>
<th>FPS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SSIM</td>
<td>PSNR</td>
<td>SD</td>
</tr>
<tr>
<td>GAN</td>
<td>0.35</td>
<td>16.05</td>
<td>17.63</td>
</tr>
<tr>
<td>VAE</td>
<td>0.28</td>
<td>15.41</td>
<td>17.21</td>
</tr>
<tr>
<td>CRN</td>
<td>0.30</td>
<td>15.72</td>
<td>17.28</td>
</tr>
<tr>
<td>Pix2pix</td>
<td>0.33</td>
<td>16.34</td>
<td>18.02</td>
</tr>
<tr>
<td>VAEGAN</td>
<td>0.65</td>
<td>27.80</td>
<td>21.79</td>
</tr>
<tr>
<td>InsulatorGAN</td>
<td>0.71</td>
<td>28.25</td>
<td>22.15</td>
</tr>
<tr>
<td>CFM-GAN</td>
<td>0.76</td>
<td>28.63</td>
<td>23.07</td>
</tr>
</tbody>
</table>

5.3. Generated Image Visualization

Figure 5 shows the experimental results. The first line and the second line represent the original image and mark the part of the image, respectively. The third line is the image generated according to the tagged image’s labeled area. According to the experimental results, we can see that the image generated by the model in this paper can reconstruct the image of the critical components of the transmission line naturally, and the generated image contains the features of the known image, which has a good generation effect.

The analysis of the result of the qualitative experiment is represented in Figures 6 and 7. The resolution of the test image is 512 × 512. The image generated by model CFM-GAN in the work is more evident than the original image, and the information about the key components and background of the high-voltage transmission line in the generated image is also richer. When the mainstream model generates images, it is prone to problems such as blurred, distorted, and distorted images. On the other hand, the CFM-GAN model can generate high-resolution images with rich semantic detail even when generating images under complex image backgrounds. Additionally, the resulting image is highly similar to the original image.
Figure 5. The experimental results of generating images are based on the essential components’ marked images on KCIGD dataset.

Figure 6. Test instances of each model on KCIGD dataset.
5.4. Sensitivity Analysis

In the present chapter, to determine the effects of various compositions on the CFM-GAN models, we perform sensitivity analyses on multi-level generators, the number of Monte Carlo searches introduced, multi-scale discriminators, the number of iterations, and the minimum training data set.

5.4.1. Two-Stage Generation

To verify the effect of the two-stage generator on the CFM-GAN model, we performed an experimental analysis on the KCIGD dataset with different numbers of generators introduced. As shown in Table 6, the image generated by a three-stage generator could be richer in semantic detail and higher in resolution than a single-stage generator. In addition, the effect was improved because the multiple structures worked independently. We fused the image details that the local generator was good at with the high-level abstract semantic information extracted by the global generator, which made multiple generators retain the semantic and texture information of the final generated image. Therefore, the multiple structures effect was better than the standard single-layer global generator. In other words, the global and local generator distinctions were not considered when using several-stage generators. It is just that the image details of model mining would be more prosperous with each additional generator. However, considering the speed of image generation, the effect of the two-stage model was the best.

Table 6. A comparison of the validity of generator networks.

<table>
<thead>
<tr>
<th></th>
<th>IS</th>
<th>FID</th>
<th>FPS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single generator</td>
<td>3.76</td>
<td>56.69</td>
<td>81</td>
</tr>
<tr>
<td>Two-stage generator</td>
<td>4.23</td>
<td>54.98</td>
<td>62</td>
</tr>
<tr>
<td>Three-stage generator</td>
<td>4.42</td>
<td>54.74</td>
<td>43</td>
</tr>
</tbody>
</table>
5.4.2. Monte Carlo Search

This paper confirms the impact of Monte Carlo search times on the model CFM-GAN performance. It compares the effects of various Monte Carlo search times on image generation results of crucial components of power lines. As shown in Table 7, when introducing a small number of Monte Carlo searches at the beginning, the enhancement of the image generation effect is the most evident. As the number of introductions increases, the result slowly increases, and the time spent increases rapidly. Therefore, when analyzing the number of introductions, achieving the best balance is achieved by combining the image generation and time spent when N = 5, i.e., the model is best balanced when the number of Monte Carlo searches is 5.

Table 7. Introducing a comparison of Monte Carlo search times.

<table>
<thead>
<tr>
<th>MCS</th>
<th>SSIM</th>
<th>PSNR</th>
<th>SD</th>
<th>FPS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Not introduced</td>
<td>0.60</td>
<td>27.52</td>
<td>20.46</td>
<td>74</td>
</tr>
<tr>
<td>N = 1</td>
<td>0.67</td>
<td>27.67</td>
<td>21.51</td>
<td>68</td>
</tr>
<tr>
<td>N = 3</td>
<td>0.73</td>
<td>27.86</td>
<td>22.42</td>
<td>66</td>
</tr>
<tr>
<td>N = 5</td>
<td>0.76</td>
<td>28.63</td>
<td>23.07</td>
<td>62</td>
</tr>
<tr>
<td>N = 7</td>
<td>0.78</td>
<td>28.71</td>
<td>23.64</td>
<td>55</td>
</tr>
<tr>
<td>N = 9</td>
<td>0.79</td>
<td>28.95</td>
<td>24.18</td>
<td>49</td>
</tr>
</tbody>
</table>

5.4.3. Multi-Level Discriminator

To validate the effect of a discriminator with multiple input scales on model performance, we compare the experimental results of introducing different numbers of discriminators on the KCIGD dataset. After obtaining a feature map of the same size as the original image (512 × 512), the feature maps are down-sampled by 2x (256 × 256), 4x (128 × 128), and 8x (64 × 64), and the feature maps of different scales fed into the discriminators of the corresponding sizes, respectively. Among them, the single-stage discriminator inputs the original image. Additionally, the original image and the image after 2x down-sampling are input to the two-stage discriminator. The three-stage discriminator model’s input is an original image, a 2x down-sampled image, and a 4x down-sampled image. On top of the three-stage model input, we add the result of 8x down-sampling as the input to the four-stage model. Table 8 shows that four discriminators slightly improve the pixel accuracy metric over using three. However, it decreases the speed by 5 FPS, thus it is not worth it. Therefore, the model achieves an optimal balance when using three discriminators. Compared with the original discriminator method, the reason for the effective improvement is that the three-layer structure uses the same convolution layer for feature extraction, and they, respectively, work on the input feature maps of different scales simultaneously. The discriminator with a high input scale focuses on the details and the texture information of the image. In contrast, the discriminator with a low input scale focuses on the high-level abstract semantic information of the image. The three work together, judging the input image’s authenticity, which finally urges the generator to generate a more realistic critical components image with more evident semantics.

Table 8. A comparison of the validity of discriminatory networks.

<table>
<thead>
<tr>
<th>Discriminator</th>
<th>SSIM</th>
<th>PSNR</th>
<th>SD</th>
<th>FPS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single discriminator</td>
<td>0.63</td>
<td>27.38</td>
<td>20.91</td>
<td>71</td>
</tr>
<tr>
<td>Two-level discriminator</td>
<td>0.69</td>
<td>27.67</td>
<td>21.89</td>
<td>67</td>
</tr>
<tr>
<td>Three-level discriminator</td>
<td>0.76</td>
<td>28.63</td>
<td>23.07</td>
<td>62</td>
</tr>
<tr>
<td>Four-level discriminator</td>
<td>0.78</td>
<td>28.92</td>
<td>23.74</td>
<td>57</td>
</tr>
</tbody>
</table>

5.4.4. Number of Epochs

In the image generation model, selecting iteration parameters is very crucial. Once the number of iterations is too low, it will not be able to reflect the actual distribution of image sample space accurately. On the contrary, if the number of iterations is too high,
overfitting behavior will occur in the model, thus affecting the generalization function of the model. Therefore, we compared the number of iterations for the CFM-GAN model with the KCIGD dataset to obtain better model performance. As shown in Table 9, as the number of iterations increases, the image generated by the model becomes better and better. However, when the number of iterations is greater than 200, CFM-GAN slightly reduces the efficiency of image generation due to the overfitting problem of the image generation model. In conclusion, when the number of iterations is 200, the model’s performance reaches the best balance.

Table 9. Impact of varying epoch numbers on experimental results.

<table>
<thead>
<tr>
<th>Number of Epochs</th>
<th>SSIM</th>
<th>PSNR</th>
<th>SD</th>
<th>FPS</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>0.38</td>
<td>17.69</td>
<td>19.03</td>
<td>65</td>
</tr>
<tr>
<td>100</td>
<td>0.61</td>
<td>19.57</td>
<td>20.17</td>
<td>60</td>
</tr>
<tr>
<td>150</td>
<td>0.69</td>
<td>25.62</td>
<td>22.48</td>
<td>61</td>
</tr>
<tr>
<td>200</td>
<td>0.76</td>
<td>28.63</td>
<td>23.07</td>
<td>62</td>
</tr>
<tr>
<td>250</td>
<td>0.73</td>
<td>28.03</td>
<td>22.85</td>
<td>58</td>
</tr>
</tbody>
</table>

5.4.5. Minimum Training Data Experiment

For verifying the impact of KCIGD dataset size on the model’s generalization performance, the CFM-GAN model was applied to training sample sets of different sizes for comparative analysis. As shown in Table 10, the performance of the CFM-GAN model does not decrease significantly with the reduction in training sets. Additionally, this is enough to indicate that the robustness of the CFM-GAN model is strong and can still extract critical information from images, even on small datasets. It overcomes the problem of the weak generalization ability of previous research models to a certain extent.

Table 10. Experimental results for minimum training data.

<table>
<thead>
<tr>
<th>The Amount of Training Set</th>
<th>SSIM</th>
<th>PSNR</th>
<th>SD</th>
</tr>
</thead>
<tbody>
<tr>
<td>4200 (100%)</td>
<td>0.76</td>
<td>28.63</td>
<td>23.07</td>
</tr>
<tr>
<td>3780 (90%)</td>
<td>0.72</td>
<td>27.96</td>
<td>22.92</td>
</tr>
<tr>
<td>3360 (80%)</td>
<td>0.69</td>
<td>27.01</td>
<td>21.47</td>
</tr>
<tr>
<td>2940 (70%)</td>
<td>0.66</td>
<td>26.12</td>
<td>20.89</td>
</tr>
<tr>
<td>2520 (60%)</td>
<td>0.60</td>
<td>24.98</td>
<td>19.62</td>
</tr>
</tbody>
</table>

5.5. Ablation Analysis

We performed ablation analyses for the CFM-GAN models to verify the effect of various ingredients in the CFM-GAN models. As indicated by Table 11, due to the MoCo model playing an active role in the feature extraction of the global generator, the index of model B is better than that of A. Model C has significantly superior metrics than B, demonstrating that the two-stage generation model combining global and local generators can enhance the sharpness of the generated images. Model D introduces a multiscale discriminator. It drives the generator to generate more realistic transmission line element images and enhances the model’s stability. By observing the scores of model E, we find that the penalization mechanisms significantly improve the model CFM-GAN’s properties, mainly because it imposes sufficient semantic information constraints on the intermediate states of the generator, which makes the resulting images.

Table 11. Results of ablation analysis.

<table>
<thead>
<tr>
<th>Model</th>
<th>Architecture</th>
<th>SSIM</th>
<th>PSNR</th>
<th>SD</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>GAN</td>
<td>0.35</td>
<td>16.05</td>
<td>17.62</td>
</tr>
<tr>
<td>B</td>
<td>A + MoCo</td>
<td>0.43</td>
<td>20.82</td>
<td>18.55</td>
</tr>
<tr>
<td>C</td>
<td>B + two-stage generation</td>
<td>0.58</td>
<td>22.58</td>
<td>20.36</td>
</tr>
<tr>
<td>D</td>
<td>C + multi-level discriminator</td>
<td>0.65</td>
<td>25.91</td>
<td>21.83</td>
</tr>
<tr>
<td>E</td>
<td>D + Monte Carlo search</td>
<td>0.76</td>
<td>28.63</td>
<td>23.07</td>
</tr>
</tbody>
</table>
5.6. Computational Complexity

By calculating the network training parameters and training times of several mainstream models, we can evaluate the models’ spatial and time complexity comprehensively. It can be found from Table 12 that the training time, together with the network parameters of CFM-GAN, is slightly increased compared with some other mainstream generation models. Still, the performance improvement brought by CFM-GAN is worth it.

Table 12. Network parameters (Param.) and the training time of various models.

<table>
<thead>
<tr>
<th>Model</th>
<th>Param.</th>
<th>Training Time (h)</th>
</tr>
</thead>
<tbody>
<tr>
<td>GAN</td>
<td>42 M</td>
<td>13.84</td>
</tr>
<tr>
<td>VAE</td>
<td>33 M</td>
<td>10.88</td>
</tr>
<tr>
<td>CRN</td>
<td>37 M</td>
<td>11.30</td>
</tr>
<tr>
<td>Pix2pix</td>
<td>46 M</td>
<td>14.51</td>
</tr>
<tr>
<td>VAEGAN</td>
<td>75 M</td>
<td>16.92</td>
</tr>
<tr>
<td>InsulatorGAN</td>
<td>82 M</td>
<td>22.73</td>
</tr>
<tr>
<td>CFM-GAN</td>
<td>84 M</td>
<td>23.58</td>
</tr>
</tbody>
</table>

6. Conclusions

This work proposes an image generation mode based on UAV aerial photography for crucial components of high-voltage transmission lines. First, CFM-GAN employs a global generator to acquire a target image at low resolution and, subsequently, employs a Monte Carlo search strategy for mining the hidden spatial information of the intermediate-produced images. A local generator is applied to produce the high-resolution target image according to the sufficient semantic information mined. Qualitative and quantitative experiments show that this paper can generate high-definition and semantic information-rich images of crucial components of transmission lines compared with all baselines. The ablation experiments show that the penalization mechanisms presented in the current work based on Monte Carlo search policies improve image quality. In addition, we consider that the high-voltage transmission lines are located in a complex environment, making the captured images’ semantic details rich. Additionally, critical components obtained from aerial photography of drone transmission lines occupy a relatively small portion of the image. The model can still generate highly realistic images of crucial components of transmission lines. Therefore, this type of dataset is representative to a certain extent. This study can obtain a good image generation effect on this kind of data set, and it is believed that the result on other data sets will not disappoint people. However, in the image generation part of this study, only images with a resolution of around $500 \times 500$ can be directly input into the model to generate high-quality images. An image with a resolution of more than $500$ (such as $1000 \times 2000$) can only be segmented (divided into eight blocks) to be inputted into the model to generate the image, and then it is synthesized and spliced to receive the final generated high-resolution image. For the next step, on the one hand, we should investigate the model’s applicability in different fields, such as image epoch panning and style transformation. On the other hand, we will also consider combining a model specially designed for processing high-resolution images (such as $5000 \times 5000$) with the image generation model in this study to solve the problem of input size limitations.

Author Contributions: Conceptualization, J.W. and Y.L.; methodology, J.W., Y.L. and W.C.; validation, J.W.; formal analysis, J.W. and Y.L.; investigation, J.W., Y.L. and W.C.; resources, J.W., Y.L. and W.C.; writing—original draft preparation, J.W.; writing—review and editing, J.W., Y.L. and W.C.; visualization, J.W.; supervision, Y.L.; project administration, Y.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the National Natural Science Foundation of China, grant number 61962031; the National Natural Science Foundation of China, grant number 51667011; and the Applied Basic Research Project of Yunnan province, grant number 2018FB095.

Institutional Review Board Statement: Not applicable.
Informed Consent Statement: Not applicable.

Data Availability Statement: The data in this paper are undisclosed due to the confidentiality requirements of the data supplier.

Acknowledgments: We thank the Yunnan Electric Power Research Institute for collecting the transmission line UAV inspection data, which provided a solid foundation for the verification of the model proposed in this paper. At the same time, we thank the reviewers and editors for their constructive comments to improve the quality of this article.

Conflicts of Interest: The authors declare no conflict of interest.

Acronyms

- Unmanned aerial vehicles (UAV)
- Generative adversarial nets (GAN)
- Chinese Power Line Insulator Dataset (CPLID)
- The Key Component Image Generation Dataset (KCIDG)
- GAN networks based on coarse–fine-grained generators and multiscale discriminators (CFM-GAN)
- Variational auto-encoders (VAE)
- Deep convolutional generative adversarial networks (DCGAN)
- Wasserstein generative adversarial nets (WGAN)
- Information-maximizing generative adversarial nets (InfoGAN)
- Conditional generative adversarial networks (CGAN)
- Matching-aware discriminator and learning with manifold interpolation (GAN-INT-CLS)
- Text-conditioned auxiliary classifier generative adversarial network (TAC-GAN)
- Auxiliary classifier generative adversarial network (AC-GAN)
- Text-to-image (T2I)
- Self-attention generative adversarial networks (SA-CGAN)
- Pixel-yo-pixel (Pix2pix)
- Low-resolution images (LR image)
- High-resolution images (HR image)
- Monte Carlo search (MCS)
- Feature pyramid network (FPN)
- Pixel recurrent neural networks (Pixel RNN)
- Pixel convolutional neural networks model (Pixel CNN)
- Deep recurrent attentive writer (DRAW)
- Dual contradistinctive generative autoencoder (DC-VAE)
- Introspective variational autoencoder (IntroVAE)
- The β-VAE framework joint distribution of continuous and discrete latent variables (Joint-VAE)
- Fused generative adversarial nets and conditional generative adversarial nets (Fused-GAN)
- Style-based generator architecture for generative adversarial networks (Style-GAN)
- Self-attention generative adversarial nets (SAGAN)
- Generative model from a single natural image (Sin-GAN)
- Momentum contrast (MoCo)
- Cascaded refinement network (CRN)
- Combination of variational auto-encoders with generative adversarial nets (VAEGAN)
- Stacked generative adversarial networks (StackGAN)
- Inception score (IS)
- Fréchet’s inception distance (FID)
- Structural similarity (SSIM)
- Peak signal-to-noise ratio (PSNR)
- Sharpness difference (SD)

31. Wu, Y.; Xu, L.H. Image Generation of Tomato Leaf Disease Identification Based on Adversarial-VAE. *Agriculture* 2021, 11, 981. [CrossRef]


**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.