Hindawi
Journal of Sensors
Volume 2023, Article ID 1987988, 15 pages
https://doi.org/10.1155/2023/1987988

Research Article

Direction Consistency-Guided Lightweight Power Line Detection Network for Aerial Images

Guanying Zhang,[1] **Yunhao Shu**,[1] **Wenming Zhu**,[1] **Jianxun Ma**,[1] **Yun Liu**,[2] **and Chang Xu**[2]

[1]*State Grid Changzhou Power Supply Company, State Grid Jiangsu Electric Power Co. Ltd., Changzhou 213000, China*
[2]*College of Information Science and Engineering, Hohai University, Nanjing 213000, China*

Correspondence should be addressed to Chang Xu; xuchang@hhu.edu.cn

Received 6 July 2023; Revised 6 November 2023; Accepted 7 November 2023; Published 19 December 2023

Academic Editor: Yunchao Tang

Accurate detection of power lines in aerial images is of great significance in ensuring grid security. However, complex power line scenarios and the thin and light structure of power lines both make it difficult to detect power lines accurately. Most of the existing approaches use traditional deep learning methods, using networks with a large number of parameters, computation, and memory occupation, thus making them not lightweight enough to perform on mobile devices. Based on this, a lightweight power line detection network based on direction consistency and location attention is proposed. The network is designed with a coordinate-aware feature extraction layer, which performs feature extraction by four-layer stacking to achieve faster inference speed while ensuring the network has fewer parameters. This layer is also able to sense the coordinates of the center pixel of the convolution in the image during the convolution process, thus preserving the location information of the power lines. In order to enhance the power line representation, a two-stage context-guided module is later utilized to simultaneously learn local features, surrounding context, and global context. Then, the features are input into a Gaussian kernel estimation module and features are aggregated in the corresponding directions through Gaussian kernels of eight different directions. The main directions of the power lines in the image and the corresponding Gaussian convolution kernels are obtained by filtering the feature responses. In addition, a kernel-guided decoder module is proposed to take advantage of the estimated power line features in the main direction of Gaussian kernel aggregation. This module can effectively enhance the power line representation and maintain the continuity of power lines. Meanwhile, low-level features are introduced to recover the edge details to realize high performance in distinguishing dense power lines. Both ablation experiments and comparison experiments on the transmission towers and power lines aerial-image and Power Line Aerial Image Dataset show that the proposed power line detection network has a good segmentation performance in complex scenarios. The proposed method performs the best in the comparison experiments, improving over the suboptimal method by 3.51% on average for the max F-measure metric.

1. Introduction

Segmenting power lines from aerial imagery is a very challenging task [1]. Aerial images have different viewpoints, and when the camera on board, the unmanned aerial vehicle (UAV) captures the target in an elevated perspective, the background is mostly the sky. However, when the viewpoint is switched to an overhead perspective, the background can be woods, cities, mountains, countryside, etc., with remarkable diversity and complexity. When the color of the target and the background tend to be similar, it is difficult to accurately and completely segment the target from the image. Finally, power lines have very thin structural characteristics [2] and usually cover only a small part of the image, for example, only a few pixels wide in aerial images, so the segmentation of power lines is easily fragmented, resulting in the loss of the original continuity and poor segmentation performance [3, 4].

Most of the existing power line detection work uses manual feature extraction-based methods, which have several drawbacks. First, the early methods divide the power line detection work into three parts, namely, manual feature extraction, straight line detection, and power line selection. Yan et al. [5] presented an algorithm to automatically extract the power line from aerial images, first a radon transform is used to extract line segments of the power line, followed by the grouping method to link each segment, and finally the

Kalman filter technology is applied to extract entire power lines. Zhang et al. [6] proposed a method of power line detection and tracking, first the Hough transform is used to extract line segments of power lines, and then K-means is utilized in the Hough space to cluster and filter the straight lines according to the characteristics of the power lines, subsequently, a Kalman filter is used to track the power lines. However, the background of power lines is not always the sky, and when there is line interference in the background, it is difficult to eliminate the interference and obtain power lines accurately by the straight line detection method [7]. Second, with the edge detection method [8, 9], the power lines need to have a very obvious contrast with the surrounding background. Moreover, accurate segmentation can only be achieved in ideal situations, while in actual aerial images, the colors of the power lines and the background may be very similar. Recently, deep learning-based methods [10–12] have also been gradually applied to power line detection, Titov et al. [10] built a defect detection system in blocks, and yolov3 is used for detecting and classifying power line poles in images or videos. Pan et al. [11] raised a power line extraction network combined with the encoder–decoder framework to extract power lines automatically with an introduced self-attention block and an introduced multiscale feature enhancement block. Some methods transform power line detection into generic line segment detection [13, 14], but similar to the problem of traditional straight line detection, power lines are not always the only straight shapes in an image. Some methods [15, 16] also treat power line detection as a salience detection problem, which relies on the joint inference of line salience and continuity; this, however, does not apply to aerial images with complex backgrounds. Other methods [17–19] directly turn the problem into a pixel-by-pixel classification based on CNN networks during training, which counts too much on the aggregation of local information, without considering the power lines' structure and global consistency well. The existing models adopt a lot of stacked layers to improve the accuracy of power line detection, without taking into account the lightweight and real-time requirements when the power line detection algorithms are applied to UAV inspection.

To address the issues raised above, we propose a lightweight power line detection network based on direction consistency and location attention. First, a coordinate-aware feature extraction module (CAFEM) is designed to complete feature extraction by four-layer stacking, replacing the classical backbone networks such as VGG16 (visual geometry group network) [20] and ResNet [21]. In this way, the network has a smaller account of parameters and faster inference speed, and the center pixel of the convolution in the image can be sensed during the convolution coordinates, thus preserving the position information of the power lines. After that, the local features and the surrounding context are learned simultaneously by using a two-stage context-guided module, and the power line representation is enhanced by learning the global context. The features are subsequently fed into the proposed Gaussian kernel estimation module, and the features of the corresponding directions are aggregated by using Gaussian kernels of eight different directions. The main directions of the power lines in the images and the corresponding Gaussian convolution kernels are obtained by filtering the feature responses. Then, a Gaussian kernel-guided decoder module is proposed to effectively strengthen the power line representation and maintain the continuity of power lines. It takes advantage of the estimated Gaussian kernels to aggregate the power line features in the main direction, and the low-level features extracted by the coordinate-aware features are introduced to recover the edge details of power lines and effectively distinguish the dense power lines. After passing through two decoders, the power line detection results are finally obtained with a layer of convolutional layers and upsampling operations. The network has ultimately a lightweight structure with only three downsampling stages, and the feature map resolution is reduced to 1/8 at the lowest, which can retain more discriminative spatial information compared with the mainstream five downsampling stages and 1/32 resolution.

The main contributions of this paper can be summarized as follows:

(1) A feature extraction enhancement module based on direction consistency and location attention is proposed, including a CAFEM and a context-guided module. The CAFEM completes feature extraction by four-layer stacking, which ensures that the network has fewer parameters and faster inference. The coordinates of the center pixel of the convolution in the image can be sensed during the convolution process, and the position information of the power lines is preserved. Then, the context-guided module enhances the features by learning local features, surrounding context, and global context.

(2) A Gaussian kernel-guided decoder module based on direction consistency is proposed to effectively strengthen the power line representation and maintain the continuity of power lines. It uses the estimated Gaussian kernels to aggregate the power line features in the main direction. The low-level features extracted by the coordinate-aware features are introduced to recover the edge details of power lines and effectively distinguish the dense power lines.

(3) Moreover, the Power Line Aerial Image Dataset (PLAID) is constructed due to the limited available power line datasets for pixel-wise detection. The experimental results of our method show good performance on the self-build dataset.

The rest of this paper is organized as follows. Section 2 introduces the related research about power line detection and lightweight semantic segmentation. The proposed method and the self-build dataset are elaborated in Section 3. Section 4 gives experimental results and a detailed discussion of the proposed method. Finally, in Section 5, conclusions are made.

2. Related Work

2.1. Power Line Detection. Power line detection has been a topic of interest within the research community, with a variety of methods being proposed. In conclusion, these approaches

can be divided into two categories. One is based on manual feature extraction and another is based on deep learning algorithms.

Manual feature extraction-based methods mainly detect power lines based on edge detection or joint features. Edge detection-based methods were proposed earlier. Candamo et al. [22] argued to use Canny edge detector and morphological filtering followed by motion estimation to conduct edge detection. Golightly and Jones [23] viewed power lines as straight lines, which were detected by the Hough straight line detection algorithm. Later, some scholars proposed joint feature-based approaches to the segmentation of power lines. In Zhu et al. [24], a low–high pass block and an edge attention fusion module were proposed to extract spatial and semantic information, improving the power line detection result along the boundary. Wu et al. [25] combined Deeplab V3+ model and edge detection to finely segment bunches. Yang et al. [26] proposed an attention block and an attention fusion network to separately capture global contextual features and utilize local feature maps, thus acquiring more rich contextual information. Inspired by locusts' looming-sensitive neurons, namely the Lobula giant movement detector, Wu et al. [27] presented a neural computational model that fused a line-attention module with the Lobula giant movement detector model to solve the problem that previous models do not reflect small-size objects well.

With the development of deep learning, deep learning algorithm-based detection methods have shown promising detection capabilities for power line inspection. Based on the holistically-nested edge detection (HED) algorithm, Liang et al. [28] used the TensorFlow deep learning framework to build a HED network model to realize the pixel-wise segmentation. Hybrid models for power line detection have also been developed. Guo et al. [29] extracted the characteristics of transmission lines through a deep convolutional neural network, and the improved AlexNet model and SVM classification method were incorporated to realize the classification of various types of power equipment. For the accurate detection of power lines in aerial images, Xu et al. [30] proposed an end-to-end convolutional neural network. Multilevel, multilayered features of images were extracted by backbone networks, the perceptual field was increased to obtain features with more global contextual details using the joint attention module. Tian et al. [31] used CNNs to classify images of damaged power lines and SVMs to identify and calculate the severity of damaged power lines using statistical information. Vemula et al. [32] proposed a power line detection segmentation algorithm based on migration learning and improved Mask R-CNN, while the framerate is too slow for real-time performance. Nevertheless, it ignored the significant position information of the power line.

However, due to the limited hardware performance in real-world application scenarios, lightweight solutions are still needed to compress the models and reduce the number of parameters and computations.

2.2. Lightweight Semantic Segmentation Models. Lightweight semantic segmentation models require a good trade-off between accuracy and latency. Computational complexity is positively correlated with spatial resolution, and spatial resolution can be reduced by mediating spatial information loss.

Encoder–decoder architecture is an effective way to solve the problem, Enet [32] and SegNet [33] are typical examples. ICNet [34] used different calculations for inputs of different resolutions, specifically, PSPNet [35] was used for low-resolution images, the parameters were shared between branches for medium-resolution images, and light CNN was used for high-resolution images. In BiSeNetV2 [36, 37], a bilateral segmentation network was designed to extract detailed features and semantic features separately, and finally, they were fused to achieve a balance between accuracy and latency. However, this network had a poor performance for specific tasks, based on which STDCSeg [38] removed structural redundancy and reduced feature map dimensionality.

Some efficient blocks are applied to improve the efficiency of the network, Wu et al. [39] proposed a CG block that learned joint feature of both local feature and surrounding context, and enlightened by SENet [39], the global context was treated as a weighted vector to channel-wisely refine the joint feature. Also, depth-wise separable convolution is a key block to reduce the amount of calculation, which has been used in many network architectures, such as MobileNet [40], IGCV3 [41], and ShuffleNet [42, 43]. Shi et al. [44] replaced normal convolution with depth separable convolution, performing information exchange across channels to solve the problem of information blocking between channels. Yang et al. [45] proposed a lightweight semantic segmentation network based on U2Net, replacing the normal convolution in upsampling and downsampling in U2Net with deeply separable convolutions, effectively reducing the computational effort of the model. Wu et al. [46] introduced the InvolutionBottleneck module and modified the loss function to construct a lightweight YOLOv5-B, which is further used to detect and identify banana-bearing branches, rachides, and flower buds in orchards with complex background.

There are other approaches that can also lighten semantic segmentation. Ma et al. [47] combined the visual and linguistic encoders to jointly extract features. Then, a cross-modal fusion module was used to bridge the embedding space, and finally upsampled the features to the original resolution by a visual decoder, and the segmentation results could be derived by calculating the cosine similarity between visual and linguistic. Cheng et al. [48] presented a new lightweight segmentation network search method through local information exchange and global information fusion; specifically, a graph convolutional network bootstrap module was used to pass local information of neighboring units and a densely connected fusion unit to perform global information aggregation.

3. Materials and Methods

3.1. Network Framework. The architecture of the direction consistency-guided lightweight power line detection network is shown in Figure 1. First, in the feature extraction stage, the coordinate convolution layer [49] and the coordinate attention

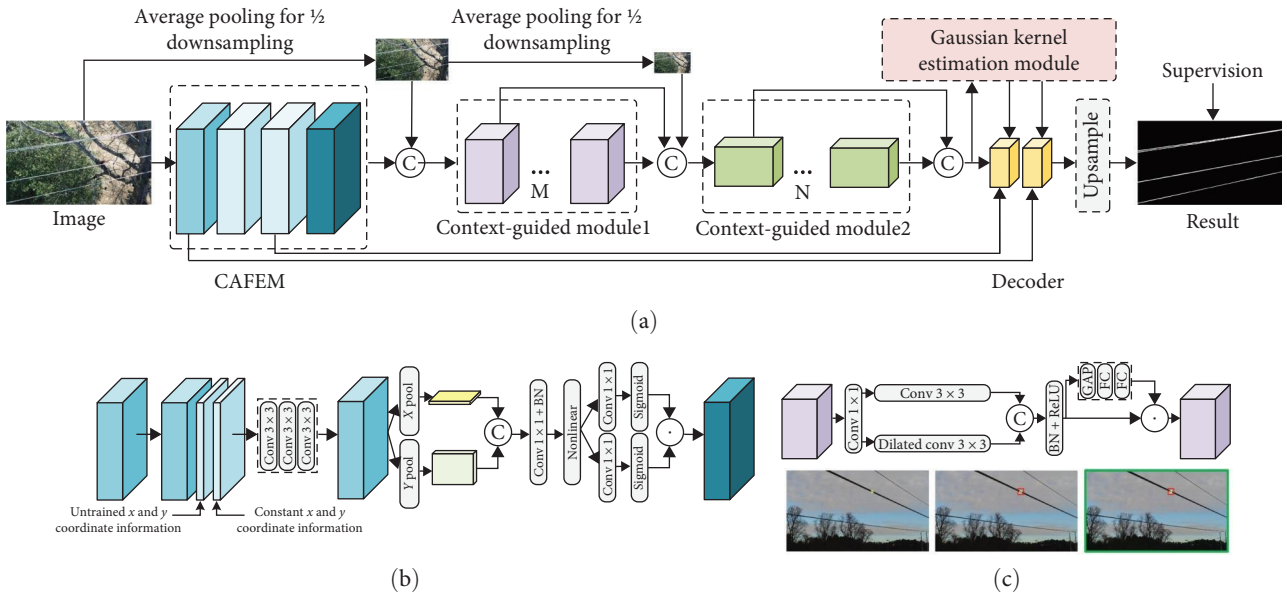

FIGURE 1: (a) Overall framework of the proposed network. (b) Coordinate-aware features extraction module. (c) Context-guided module.

layer [50] are combined to design a CAFEM. This module extracts power line features in a four-layer stack, which can effectively maintain and enhance the location information of power lines while lightweighting the network. Then, the feature representation of power lines is highlighted by capturing the local features, surrounding context, and global context of power lines through a context-guided module, which is divided into two stages and progressively processes features with different resolutions to form the initial attention information of power lines. To address the problems of coarse power line features, incomplete local connections, and still some background interference, a Gaussian kernel estimation module is proposed. Based on the characteristics of similar power line structures and the same angles in most scenes, the main direction of power lines can be searched, and Gaussian kernels with similar power line structures can be constructed. Based on this, a Gaussian kernel-guided decoder module is proposed, which not only uses Gaussian kernel to deconvolute and aggregate power line features to gradually recover continuity and strengthen power line representation but also introduces low-level features in coordinate-aware feature extraction for recovering power line edge detail representation and avoiding dense power lines from being segmented into the same target. After passing through two layers of decoders, the resolution is finally recovered by upsampling operation and the segmentation result is obtained by sigmoid operation.

3.2. Coordinate-Aware Feature Extraction Module (CAFEM). Because of the spatial and structural characteristics of power lines in images, not only the similarity between multiple power lines but also the continuity of single power lines exists, so it is very necessary to keep the spatial location information of power lines in the power line detection task. At the same time, in order to ensure that the network has a small number of parameters and low computational overhead, the selection of the backbone network is very important. The mainstream VGG, ResNet, and other series of networks have a large number of parameters, and the extracted features have the characteristics of substantial channels and low resolution, which are not suitable for lightweight networks.

Based on the above characteristics, the proposed method designs a lightweight CAFEM for early feature extraction to maintain and enhance the spatial location information of power lines. As shown in Figure 1(b), the module first extracts the power line features of the input image $I \in \mathbb{R}^{3\times H\times W}$ using a coordinate convolution layer with a step size of 2 and completes 1/2 downsampling. Then, two 3×3 standard convolutional layers are used to further extract power line features.

$$F_0 = f_{3\times 3}^{\text{coord}}(I; W_0), F_1 = f_{3\times 3}^{\text{coord}}(F_0; W_1), F_2 = f_{3\times 3}^{\text{coord}}(F_1; W_2), \tag{1}$$

where $f_{3\times 3}^{\text{coord}}(*)$ denotes a 3×3 coordinate convolution layer, F_0, F_1, and F_2 denote the three convolution results, respectively, and F_0, F_1, and $F_2 \in \mathbb{R}^{32\times \frac{H}{2}\times \frac{W}{2}}$. Finally, a coordinate attention layer $f_{CA}(\cdot)$ is used to enhance the power line feature representation and the corresponding location-aware information:

$$F = f_{CA}(F_2), \tag{2}$$

where F denotes the output feature encoding, $F \in \mathbb{R}^{32\times \frac{H}{2}\times \frac{W}{2}}$.

3.3. Context-Guided Module. In order to make full use of the inherent property of image segmentation, i.e., contextual information, a context-guided module is designed to fully capture multiple contexts to help pixel-level power line segmentation. As shown in Figure 1, when power line segmentation needs to be completed, if only a local region of the power line itself is focused, as shown in the yellow region in

Figure 1(c), this region is too small in area, which easily leads to insufficient information for pixel-level classification of the region. However, if the coverage area of the region is expanded, as shown in the red region in Figure 1(c), at this time, the region contains both the power line and the surrounding environment, it is easier to identify the power line as a significant target and thus assign more weights. Further, if the global context of the whole scene is captured, as shown in the green region in Figure 1(c), it not only provides a global representation of the scene but also aggregates power line representations with similar features and structures, which can effectively improve the confidence level of power line segmentation within the yellow region. Thus, both the surrounding context and the global context help to improve the power line segmentation accuracy.

Therefore, the context-guided module is proposed to fully utilize the local features, surrounding context, and global context. For the input features F, the number of channels is first changed using a layer of 1×1 convolution. Then, the local features F_{loc} and the corresponding surrounding contexts F_{sur} are learned using 3×3 standard convolution and dilated convolution, respectively:

$$F_{\text{loc}} = f^{3\times3}(f^{1\times1}(F; W_0); W_1), \tag{3}$$

$$F_{\text{sur}} = f_d^{3\times3}(f^{1\times1}(F; W_0); W_2), \tag{4}$$

where $f_d^{3\times3}(:; W_2)$ denotes the dilated convolution and d is the dilation rate, which can be adjusted according to the feature map size. Since the dilated convolution has a larger perceptual field compared to the standard convolutional layer, it can learn the surrounding environment effectively. After that, the two features are cascaded and batch normalization (BN) and rectified linear unit (ReLU) operations are performed:

$$F_{\text{joi}} = \text{ReLU}(\text{BN}([F_{\text{loc}}; F_{\text{sur}}])). \tag{5}$$

Then, the global average pooling layer and two fully connected layers are used to extract global context information, which is used as a weighted vector for improving the joint features F_{joi} after sigmoid operation σ to get the final feature output F_{out}:

$$F_{\text{out}} = F_{\text{joi}} \cdot \sigma\big(W_4 \cdot \big(W_3 \cdot f_{\text{avg}}(F_{\text{joi}})\big)\big). \tag{6}$$

Two feature processing stages, respectively, containing M and N Gaussian kernel-guided convolution modules are constructed with the context-guided module as the base component, and a step-2 convolution operation is performed on the input features in the first module of each stage to reduce the resolution. The input feature of the first stage is $F \in \mathbb{R}^{32\times\frac{H}{2}\times\frac{W}{2}}$, and the input feature of the second stage is cascaded from three parts: the output of the first and last modules of the first stage, and the input image after 1/2 downsampling, by which feature reuse is encouraged and feature propagation is enhanced. Similarly, the output feature F_2 of the second stage is the output of the first and last modules of the stage cascaded.

3.4. Gaussian Kernel Estimation Module. Since power line detection and segmentation is a pixel-level classification task and is not an instance-level classification task for the complete target, it is easy to break the local connection of power lines during the segmentation process and destroy the integrity of a power line. As power lines are erected between the poles with a certain regularity, most of the power lines in the same area present the same angle and similar appearance, and even though some of the power lines are in the shape of hanging chain lines, their overall still present a certain main direction. Based on the above characteristics, the proposed method proposes to use Gaussian kernels with a similar structure to power lines to convolve the power line features extracted from the network and aggregate the power line features in the main direction, so that the local breaks in the power lines can be reconnected to maintain continuity and strengthen the power line representation.

Therefore, it is first necessary to obtain Gaussian kernels with the same angle as the main direction of the power lines in the image. A Gaussian kernel estimation module is designed, as shown in Figure 2.

First, according to the structural characteristics of the power lines, the size of the Gaussian kernel is set to 13×13, the two eigenvalues λ_1 and λ_2 are set to 7 and 1, and the set of rotation angles with eight directional angles is constructed:

$$\Theta = \left\{0, \frac{\pi}{8}, \frac{\pi}{4}, \frac{3\pi}{8}, \frac{\pi}{2}, \frac{5\pi}{8}, \frac{3\pi}{4}, \frac{7\pi}{8}\right\}. \tag{7}$$

Since the power lines are symmetric, the rotation angle is set in the range $[0, \pi]$. Then, the covariance matrix Σ_i is constructed based on the rotation angle $\theta_i \in \Theta$ and the feature values:

$$\Sigma_i = \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i) \\ \sin(\theta_i) & \cos(\theta_i) \end{bmatrix} \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \begin{bmatrix} \cos(\theta_i) & \sin(\theta_i) \\ -\sin(\theta_i) & \cos(\theta_i) \end{bmatrix}. \tag{8}$$

A Gaussian kernel K_i with a kernel size of 13 based on the covariance matrix is constructed:

$$K_i(z) = \frac{1}{\sqrt{2\pi|\Sigma_i|}} \exp\left(-\frac{1}{2}(z-\mu)^T \Sigma_i^{-1}(z-\mu)\right), \tag{9}$$

where $z = [x, y]^T$, μ indicates the mean value, each Gaussian kernel is more sensitive to the power line features in the same direction as itself, and these Gaussian kernels are used to convolve with the input feature F_2, as shown in Figure 2:

$$G_i = f_{\text{gaus}}(F_2; K_i), \tag{10}$$

where $f_{\text{gaus}}(; K_i)$ denotes the convolution operation parameterized by a Gaussian kernel K_i. In this case, the convolved

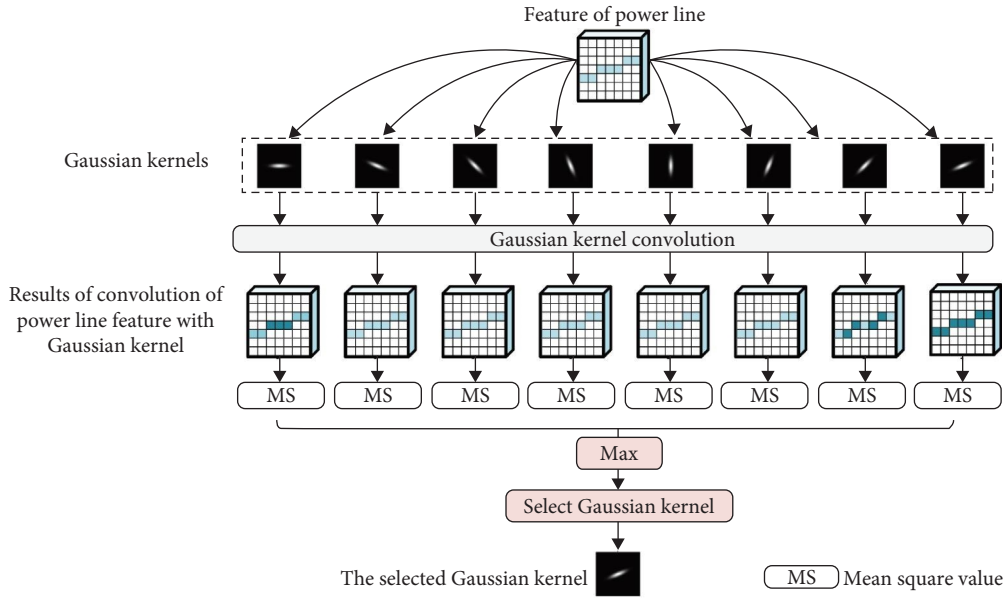

Figure 2: The flow of the Gaussian kernel estimation module.

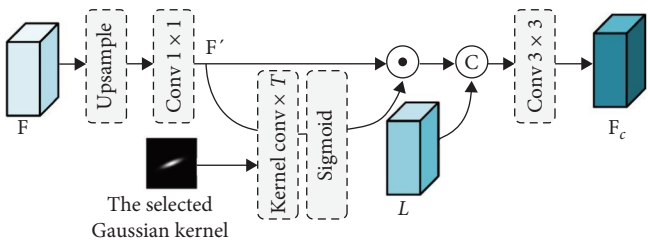

Figure 3: Architecture of proposed decoder.

Gaussian kernel with the same direction as the power line has a stronger degree of aggregation and a higher response value, as shown in Figure 2. Therefore, by calculating the mean square value of each result, the largest corresponding one can be selected, and the appropriate Gaussian kernel K can be estimated.

3.5. Kernel-Guided Decoder Module. In the final stage of the network, a two-layer decoder module is designed, which not only introduces the low-level features in the feature extraction module to gradually recover the feature resolution and power line details but also performs a convolution guided by a Gaussian kernel to aggregate the contextual information in the main direction of the power line to recover and maintain the continuity of the power line. The architecture of this decoder is shown in Figure 3. First, the input features F are upsampled to the same resolution as the introduced low-level features L, and a 1×1 convolution is performed:

$$\mathbf{F}' = f^{1\times1}(Up(\mathbf{F}); \mathbf{W}_0), \tag{11}$$

where $Up(\cdot)$ indicates upsampling operation. Then, the estimated Gaussian kernel K is used as a guide to complete T kernel convolution operations to reduce the number of channels to 1. The difference between the kernel convolution operation and the standard convolution is only that the unlearnable predefined Gaussian kernel is utilized, and the rest is the same. Finally, the power line attention weights are calculated by the sigmoid operation on the kernel convolution results and multiplied with $\mathbf{F}'$ to obtain the features of the enhanced power line representation:

$$\mathbf{F}_k = f_{\times5}(\mathbf{F}'; \mathbf{K}), \mathbf{F}'' = \mathbf{F}' \cdot \sigma(\mathbf{F}_k), \tag{12}$$

where $f_{\times5}(;\mathbf{K})$ denotes a series of five convolution operations with Gaussian kernel $\mathbf{K}$ as the parameter, $\sigma(\cdot)$ is a sigmoid activation function. Then, cascade with the low-level features and perform a 3×3 convolution to obtain the power line feature after recovering the details:

$$\mathbf{F}_c = f^{3\times3}([\mathbf{F}', \mathbf{L}]; \mathbf{W}_8). \tag{13}$$

After passing through the two-layer decoder, the features are restored to the original resolution of the input image using the upsampling operation. Then, convolution is performed to obtain the single-channel features, and the power line detection result is obtained after the sigmoid operation. So far, the network architecture proposed contains only 65 learnable convolutional layers with a small number of channels and 16 unlearnable convolutional layers based on Gaussian kernel with $M = 3, N = 15$, and two decoder layers with T of 5 and 3, respectively, which have less number of parameters and computational overhead compared with the deep convolutional network framework containing hundreds of layers and thousands of channels. Moreover, the network has only three downsampling stages and feature map resolution is down to a minimum of only 1/8, which can retain more discriminative spatial information compared to the

mainstream five downsampling stages and 1/32 feature map resolution.

3.6. Loss Function. To better supervise network learning and obtain higher quality power line segmentation results and clearer bounds, a hybrid loss consisting of a combination of BCE loss [51] and SSIM loss [52] is defined as follows:

$$\ell_{\text{total}} = \ell_{\text{BCE}} + \ell_{\text{SSIM}} + \ell_{\text{IoU}}, \tag{14}$$

where BCE loss is the most widely used loss in binary classification and segmentation and IoU is the Intersection over Union:

$$\ell_{\text{BCE}} = -\sum_{(i,j)} [G(i,j)\log P(i,j) + (1 - G(i,j))\log(1 - P(i,j))], \tag{15}$$

where G denotes the ground truth image and (i, j) denotes the pixel coordinates in the image. Originally designed for image quality assessment, SSIM captures structural information in the image and therefore integrates it into the training loss to learn the structural information of ground truth. Given $x = \{x_j, j \in [1, N^2]\}, \mathrm{y} = \{y_j, j \in [1, N^2]\}$, respectively, denote the pixel values of the two corresponding blocks cropped from the predicted result and the ground truth image. Then, the SSIM losses of x and y are defined as follows:

$$\ell_{\text{SSIM}} = 1 - \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}, \tag{16}$$

where μ_x, μ_y, σ_x, and σ_y represent the mean and standard deviation of x and y, respectively, σ_{xy} denotes covariance, and C_1 is set to 0.01^2 and C_2 is set to 0.03^2 to avoid denominator of 0. The BCE loss is computed at the pixel level, it does not consider the GT of other points around the pixel, and it weights both foreground and background pixels. This helps convergence for all pixels and guarantees a relatively good local optimum. SSIM loss is a metric for local areas that considers the local neighborhood of each pixel, and it assigns higher weights to pixels located in the border area between foreground and background, such as borders, and fine structures, so that the loss around the border is higher even if the predicted probability is the same for the border and the rest of the foreground.

4. Results and Discussion

4.1. Datasets and Implementation Details

4.1.1. Datasets. To meet the demand for UAV-based power line detection and verify the effectiveness of the proposed method, we construct the PLAID and use the publicly available transmission towers and power lines aerial-image (TTPLA) to provide a large number of diverse training images for training the optimal model.

The PLAID is constructed as follows. First, a DJI M300 RTK UAV equipped with an industrial camera is used to capture aerial images in different scenes of three 220 kV overhead high-voltage transmission lines with a total length of about 20 km. In order to enable the training model to cope with the scene changes (different lighting conditions, color distribution, complex backgrounds, etc.), avoid the problem of overfitting, and improve the generalization ability of the model, a total of 32 transmission line inspection videos are captured under various shooting angles in multiple lines, weather and lighting conditions, which are closer to the actual inspection situation. At the same time because the camera has a high shooting frame rate, it is necessary to prevent the problem of data redundancy by screening and eliminating images with high similarity, and only data enhancement operations such as normalization and random rotation are used. Finally, 2,000 aerial images with an image size of $2{,}448 \times 2{,}048$ are selected and further split into training and test sets with a ratio of 8 : 2. The image annotation tool labelme [53] is adopted for the pixel-level annotation of power lines and construct PLAID, part of which is shown in Figure 4.

TTPLA contains 1,242 power line images in urban and transmission line scenes with a resolution of $3{,}840 \times 2{,}160$. This dataset has many scenes, most of the images have complex backgrounds, and some of the images have elements that tend to interfere with power line detection such as lane lines. By training jointly with the PLAID, it can effectively complement each other and better train the network.

4.1.2. Implementation Details. The proposed method is implemented with the PyTorch framework and executed on a computer with an Intel Core i7, NVIDIA RTX-2080 with GPU memory of 12 GB. AdamW [54] optimizer and the step learning rate schedule are used during model training, and the total number of training epochs is set to 100. The initial learning rate and weight decay are set to 0.001 and 5e–4, and the learning rate decayed to 1/2 of the original value at 50 and 60 epochs, respectively. The feature values λ_1 and λ_2 are set to 7 and 3, and the number of Gaussian kernel-based guided convolution modules M and N in stages 1 and 2 are set to 3 and 9, respectively. The input images are uniformly scaled to 512×512.

4.2. Evaluation Metrics. F-measure, mean absolute error (MAE), and S-measure are used to evaluate our method and other comparison methods, all of which are widely used metrics in the field of image segmentation. Among them, F-measure is formulated as follows:

$$\mathrm{F}_\beta = \frac{(1 + \beta^2) \cdot \text{ precision} \cdot \text{recall}}{\beta^2 \cdot (\text{ precision} + \text{recall})}, \tag{17}$$

where β^2 is set to 0.3 to emphasize the importance of accuracy, and the maximum value of F-measure (Max F-measure, MaxF) is used to evaluate the performance of all methods.

The MAE evaluates the MAE between the power line detection results and the true value map, which is calculated as follows:

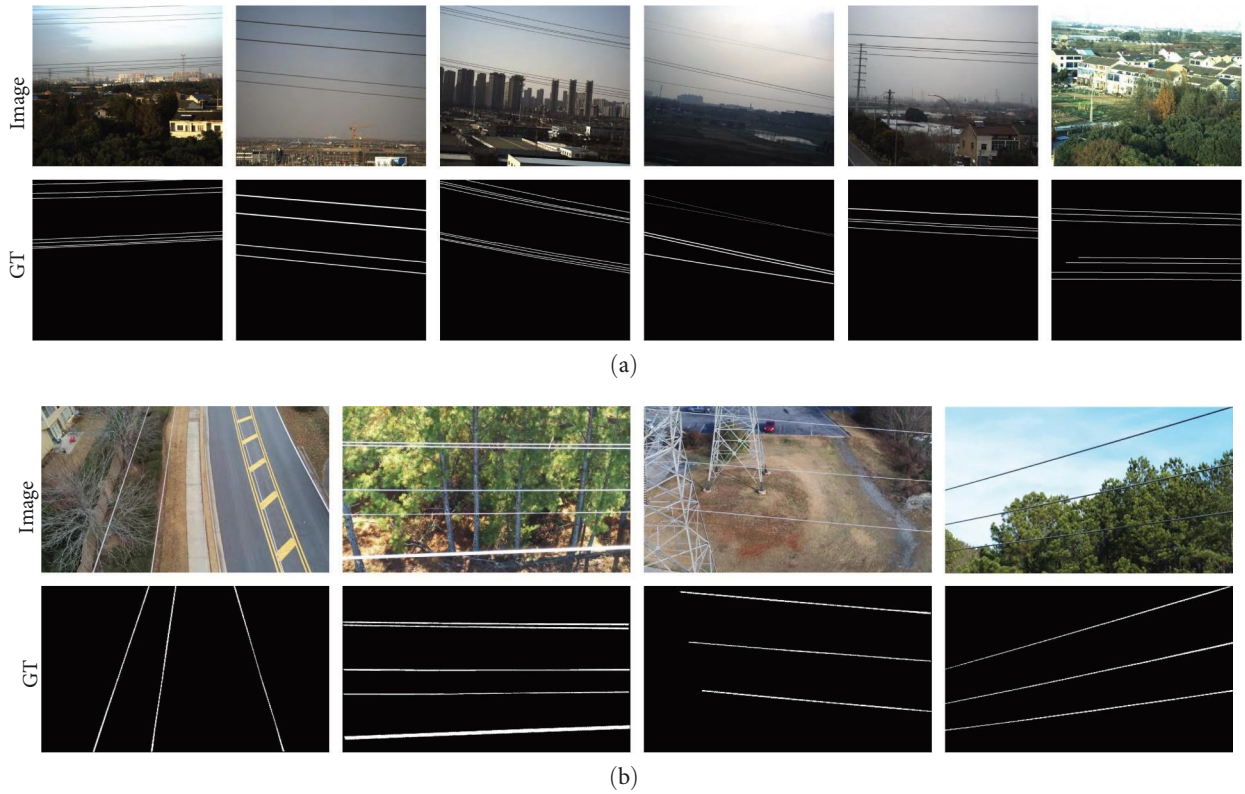

Figure 4: Examples of power line images: (a) PLAID and (b) TTPLA.

$$\mathrm{MAE} = \frac{1}{T}\sum_{t=1}^{T}|P_t - G_t|, \tag{18}$$

where P_t and Q_t refer to the detection result and the true value map normalized to the range of 0–1, respectively, and $T = H \times W$ denotes all pixel points in the image. Compared with F-measure, S-measure is closer to the human visual evaluation criteria for binary segmentation maps, and it focuses more on assessing the structural similarity of the detection results. Therefore, the S-measure metric is added to make a more comprehensive evaluation of the method, which is expressed as follows:

$$S = \gamma S_0 + (1-\gamma)S_r, \tag{19}$$

where S_0 and S_r denote the structural similarity of the region perception and object perception and the default setting of γ is 0.5.

4.3. Ablation Study. To verify the effectiveness of each module proposed in the method, the features of the Gaussian kernel estimation module as well as the decoder module are first visualized to understand more intuitively the feature variations in the modules and the effectiveness of the proposed method, and ablation experiments are conducted to compare each module to better analyze the results.

4.3.1. Feature Visualization. In order to better demonstrate the feature visualization results, five groups of images with complex backgrounds are selected, as shown in Figure 5, where Figure 5(a) has very thin power lines, resulting in a very low zone with the background. Figure 5(b) has not only the interference of lane lines but also the background switching of power lines, and the right fifth part of power lines is more similar to the background. Figure 5(c) has a similar situation, the left part of the power line is so similar to the background that the human eye can barely distinguish it. Figure 5(d) has interference from light changes, road cracks, and edges, and there are three groups of power lines with different detection difficulties. Figure 5(e) has a high degree of differentiation from the background, but there is interference from lines in the vertical direction and from the parts of the power line connection.

In this section, we visualize the features of these five groups of power line images with high difficulty. The visualization results in the Gaussian kernel estimation module for the input features and the convolutional features after eight different Gaussian kernels are shown in Figure 5. The first and second images of each group are the original images and the input features, followed by the eight convolutional features, and the corresponding value below for each feature is its mean square value.

From the five sets of results, it can be seen that when the Gaussian kernel direction is different from the power lines, the attention information in the convolution features for the power lines will be greatly scattered, resulting in a lower overall response of the image, and the attention information will only be enhanced when the main directions are the same

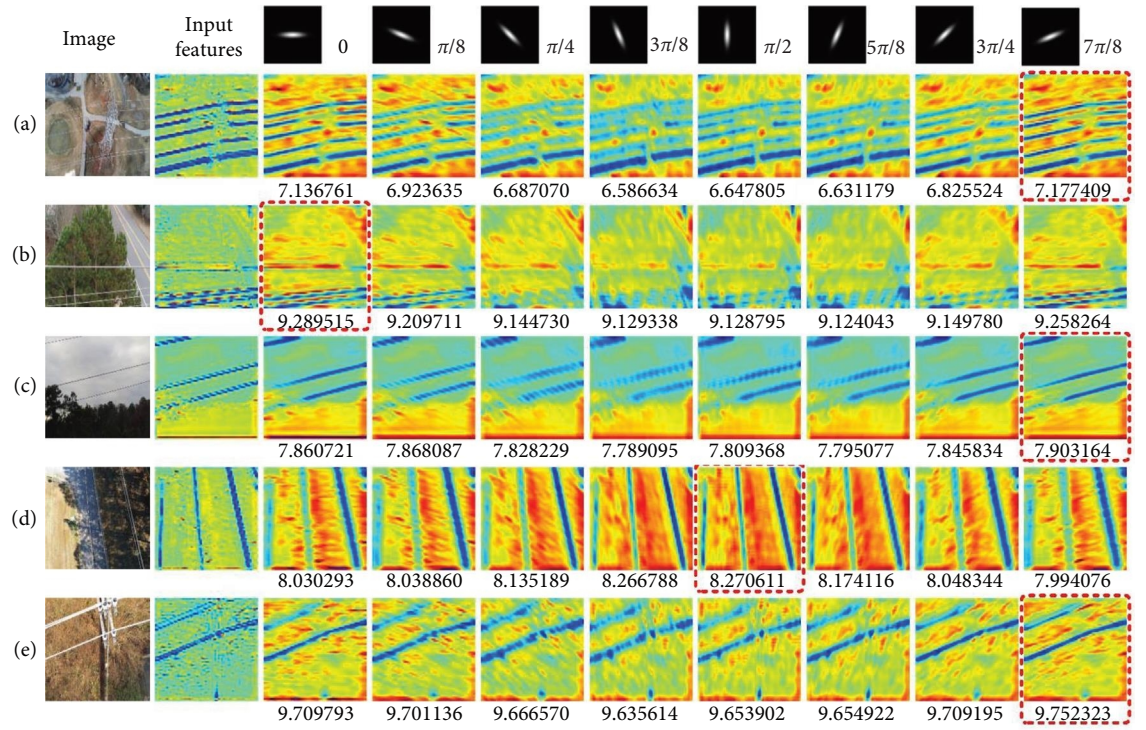

FIGURE 5: Feature visualization of Gaussian kernel estimation module. (a–e) Five samples randomly selected from the dataset.

or nearby, and more power line features will be aggregated to enhance the overall response of the image when the directions are the same, so the calculated mean square value is the largest, as in Figure 5, the results for each group of images marked with red boxes are shown.

After passing the Gaussian kernel estimation module, the two-layer decoder will perform convolution using the estimated Gaussian kernel to strengthen the power line representation of the input features. Figure 6 demonstrates the variation of the input features in the first layer decoder, where the first and second columns represent the original images and the input features of the decoder, the third column is the result of the 1×1 convolution of the input features, the fourth to ninth columns are the results of the convolution guided by the five-layer Gaussian kernel and the subsequent sigmoid operation, and the ninth column is multiplied with the third column to obtain the tenth column, the eleventh column is the result of cascading with the lower-level features and passing 3×3 convolution, and the last column is the final output of the network.

From the five sets of results in Figure 6, it can be seen that the input features of the decoder only form the attention information to the power lines, which is still relatively rough in the power line representation and edge details. Also, there exists the attention information to the background part, which easily interferes with the segmentation. After 1×1 convolution, most of the background interference can be removed, followed by five times kernel convolution in series. Meanwhile, it can be seen that the kernel convolution gradually strengthens the power line representation and restores the continuity of power lines in Figures 6(a) and (c), which effectively compensates for the power line information lost in the early feature extraction. After a sigmoid operation and multiplication with the original features, the power line representation is greatly improved and enhanced, and the distinction with background interferents is also improved. Finally, after cascading and convolving with the low-level features of the CAFEM, the power line details are effectively recovered, and it can be seen that the dense power lines in Figures 6(a) and 6(d) are successfully distinguished.

Figure 4 shows the final results of the proposed method. It can be seen that the detection results of Figure 6(b)–6(e) of images are very accurate. However, the characteristics of too thin power lines in Figure 6(a) greatly enhance the difficulty of extracting the power line features by the method, which makes the detection results not fine enough, and at the same time, the uppermost line in Figure 6(a) is completely fused with the background, which is difficult to observe, and the method only detects the more ambiguous part of the information. The power line background switching part in the upper left corner of Figure 6(d), due to the effect of lighting and more similar to the background, the detection result shows power line breakage here.

4.3.2. Ablation Study. In this section, we justify the effectiveness of each key component used in our model, and the ablation experiments are performed on both TTPLA and PLAID in Table 1. We start with the baseline variant which simply uses a three-layer convolutional layer followed by a two-layer decoder, and the results are listed in row 1 of Table 1, in which the detection performance is poor. We can find that the CAFEM improves the MaxF by

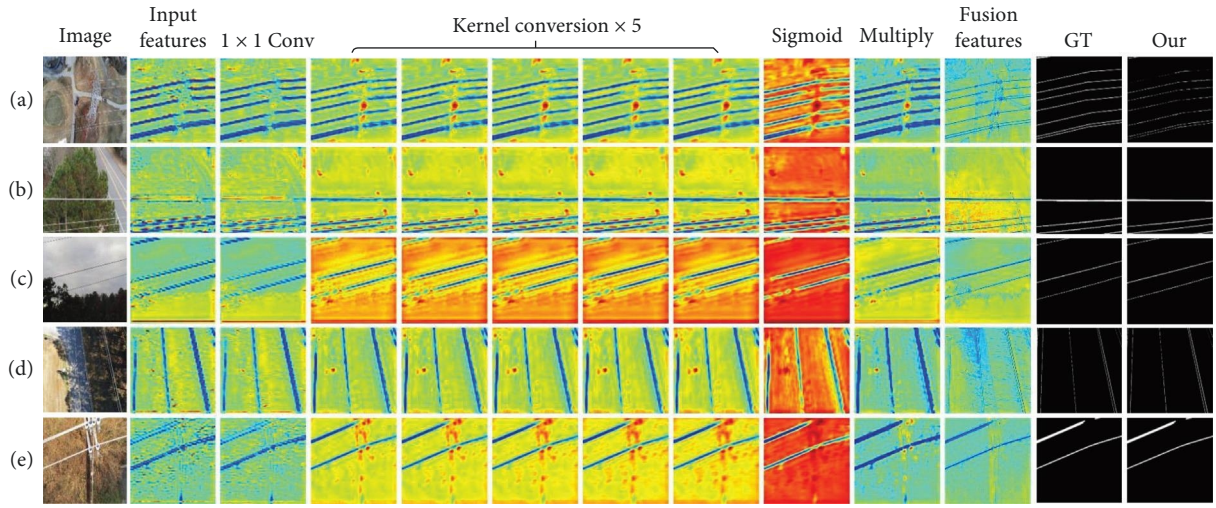

Figure 6: Feature visualization of the first decoder. (a–e) Five samples randomly selected from the dataset.

Table 1: Comparison of power line detection results with other methods on two datasets.

Variant no.	CAFEM	CGM	Decoder	PLAID			TTPLA		
				MaxF	MAE	S-measure	MaxF	MAE	S-measure
1				0.5793	0.0345	0.5275	0.6688	0.0430	0.6145
2	√			0.6251	0.0296	0.6434	0.7040	0.0361	0.6938
3		√		0.7191	0.0244	0.7921	0.8354	0.0254	0.7828
4			√	0.6079	0.0314	0.6193	0.6912	0.0401	0.6443
5	√	√	√	0.8071	0.0196	0.8549	0.9412	0.0217	0.8618

approximately 0.05 on both datasets, MAE by 0.05 and 0.07, and S-measure by 0.12 and 0.08, respectively. Also, the context-guided module enhances the MaxF by 0.14 and 0.17 on the two datasets, MAE by 0.01 and 0.17, and S-measure by 0.27 and 0.17, respectively. Since the Gaussian kernel guidance relies on the degree of attention of the input features to the power line representation, which would otherwise enhance the interference information and affect the detection results, the overall improvement is small when only the Gaussian kernel-guided module is added. However, the proposed method first extracts the power line representation using the CAFEM and the context-guided module to form the attention information for power lines. Then, the Gaussian kernel-guided decoder module is used to greatly enhance and compensate for the power line representation. Hence, the overall performance improvement is significant, with MaxF improving by 0.23 and 0.38, MAE decreasing by 0.0149 and 0.0213, and S-measure improving by 0.33 and 0.25 in both datasets compared to the basic network architecture. In short, our proposed modules are effective for power line detection.

The results of the visual comparison of the ablation experiment are shown in Figure 7. Since the introduction of decoder only in Table 1 is less effective, only the power line detection results for variant 2, variant 3, and variant 5 are shown in Figure 7. It can be seen, consistent with Table 1, that the introduction of the CAFEM retains extensive spatial location information for power lines, the introduction of the CGM makes for better continuity of power lines. However, it is the combination of the three modules that yields the final fine power line segmentation results.

4.3.3. Network Efficiency Comparison. First, in order to verify the lightweight and real-time of the proposed method, the number of network parameters, floating-point operations per second (FLOPs), memory consumption, and processing time per input image are compared between the proposed method and other deep learning-based comparison algorithms, and the results are shown in Table 2. As can be seen from the table, since the proposed method does not use the mainstream backbone network for feature extraction, and the number of convolutional layers in the network is small, the number of parameters in the network is small, only 0.46 M, which has a great advantage compared with other methods. Also, there are fewer operations in the network, which has lower computational complexity and faster processing time. However, due to the introduction of Gaussian kernel estimation and guidance module in the network, it leads to an increase in memory occupation, so it only has an advantage over HED, PFANet, and EGNet, and is slightly inferior to RCFNet, while the input resolution of PiCANet is only 224×224, which is difficult to compare. In summary, the network architecture of the proposed method has a more lightweight design, which can meet the lightweight and real-time requirements for power line detection during UAV inspection.

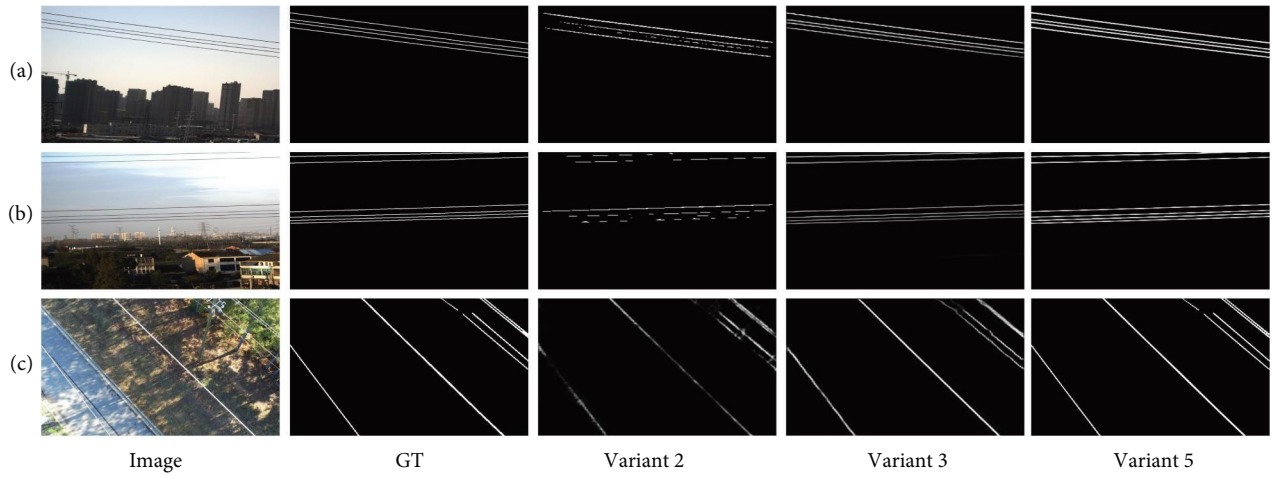

Figure 7: Visual comparison of the ablation experiments. (a–c) Three samples randomly selected from the dataset.

Table 2: Efficiency comparison results of different methods.

Method	Resolution	Param (*M*)	FLOPs (*G*)	Memory (*M*)	Processing time (s)
HED [55]	512 × 512	14.71	80.39	573.33	0.022
PFANet [56]	512 × 512	16.29	145.32	756.00	0.055
PiCANet [57]	224 × 224	47.21	54.04	208.96	0.084
RCFNet [58]	512 × 512	14.83	102.71	367.48	0.029
EGNet [32]	512 × 512	111.66	619.96	2041.28	0.177
Our	512 × 512	0.46	6.31	376.06	0.013

4.4. Comparison Experiments. To verify the effectiveness as well as the performance of the proposed method, the traditional edge detection method Canny [59], the straight line detection method LSD [60], the VGG16 network-based edge detection methods HED [55], RCFNet [58], the deep learning-based saliency detection methods PFANet [56], PiCANet [57], and EGNet [32] are selected as comparison methods. All deep learning-based networks are trained and tested on PLAID and TTPLA, in which PiCANet is more special and can only use 224 × 224 resolution input images.

4.4.1. Comparison Experiment on PLAID.

(1) Visual Comparison. To qualitatively validate the effectiveness of the proposed network, we visualize the results generated by our method and other methods on PLAID in Figure 8. As illustrated in Figure 8, introduced by the CAFEM and context-guided module, the proposed method could distinguish dense power lines and segment them with fine edges. Compared with other methods in Figure 8(a)–8(f), the segmentation results of our method can perform more complete and accurate segmentation of power lines instead of segmenting the background or getting rid of segmenting the power lines. Meanwhile, the Gaussian kernel-guided decoder improves the continuity of the segmentation results, which can be clearly seen in Figures 8(a) and 8(e) with PiCANet and EGNet.

(2) Quantitative Evaluation. The performance comparison experiments of different methods on the PLAID are shown in Table 3, where bolded are the optimal results and underlined data are the suboptimal results. From the table, it can be seen that the proposed method has the best performance because it performs directional learning for the structural features of power lines and improves 0.0019, 0.0062, and 0.0211 in three indexes in turn compared with the suboptimal method. The F-measure index suboptimal RCFNet has a large amount of background interference information, the MAE index suboptimal PFANet and the S-measure index suboptimal PiCANet have poor continuity of detected power lines and incomplete detection. Overall, our method outperforms other methods.

4.4.2. Comparison Experiment on TTPLA.

(1) Visual Comparison. To qualitatively validate the effectiveness of the proposed network, we visualize the results generated by our method and other methods on the PLAID in Figure 9. As illustrated in Figure 9, the scenes of TTPLA are more complex with many interfering factors compared to PLAID, while our method outperforms other methods in Figure 9(a)–9(d) with exactly accurate segmentation results. Specifically, the results of our method show better continuity and realize complete segmentation without background information interference. However, when the power lines are similar to the background, our method tends to miss part of the power line detection. Despite this situation, our method still exhibits optimal detection and segmentation performance.

(2) Quantitative Evaluation. The performance comparison experiments of different methods on TTPLA are shown

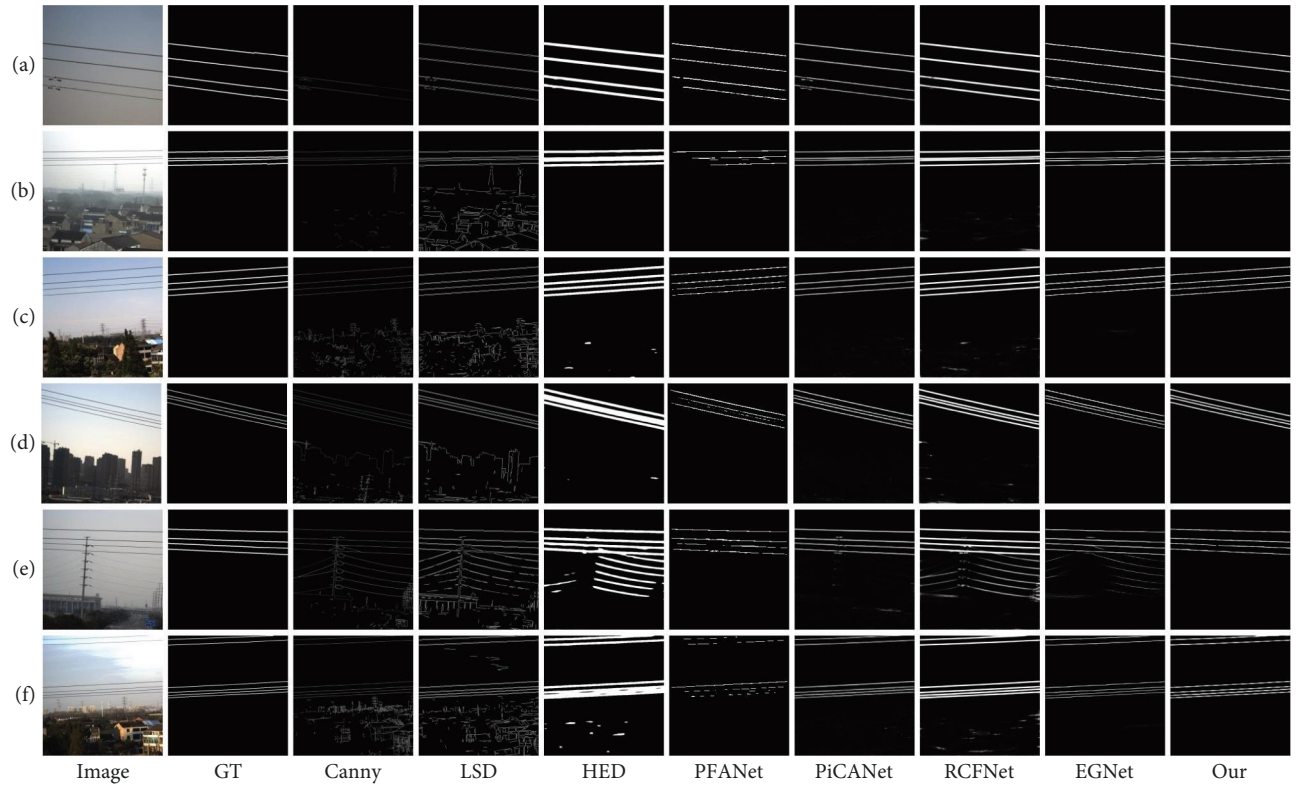

Figure 8: Visualization comparison results on PLAID. (a–f) Six samples randomly selected from the dataset.

Table 3: Performance comparison experiments on PLAID.

Methods	MaxF	MAE	S-measure
Canny [59]	0.5331	0.0415	0.5716
LSD [60]	0.4300	0.0422	0.6030
HED [55]	0.4819	0.0620	0.6236
PFANet [56]	0.6967	0.0266	0.6959
PiCANet [57]	0.7992	0.0271	0.8338
RCFNet [58]	0.8002	0.0377	0.7719
EGNet [32]	0.7320	0.0270	0.7738
Our	**0.8021**	**0.0204**	**0.8549**

The bold values are the optimal results and underlined values are the suboptimal results for each metric.

in Table 4, in which can be seen that compared with the suboptimal method RCFNet, our method has optimal performance with 0.07, 0.001, and 0.02 improvement in MaxF, MAE, and S-measure, respectively. Among the other methods, Canny and LSD are severely affected by the background interference in TTPLA and perform much worse than PLAID in terms of metrics. HED also has a large detection error and high MAE values, and the performance of PFANet, PiCANet, and EGNet is gradually improving but is still inferior to the suboptimal method RCFNet.

4.5. Discussion. A lightweight power line detection method is proposed, and the experimental results demonstrate that the proposed method can achieve good segmentation results while maintaining lightweight. We have applied the proposed network into the embedded development board NVIDIA Orin NX. Though the computation capability of embedded board is lower than desktops, the computation time of the proposed network is about 0.04 s on average, which is acceptable for real-world applications. Thus, in this way, the method can further cooperate with transmission line external obstacle detection, clearance distance measurement, and autonomous flight of UAV. However, due to the constrained modeling capability of the shallow network, when encountering multidirectional power lines, or at the intersection of power lines and transmission towers, the proposed method may be subject to misdetection owing to the presence of linear structural features similar to those of power lines in the transmission towers. Hereto, we will continue to investigate this later, possibly using the transformer module to extract

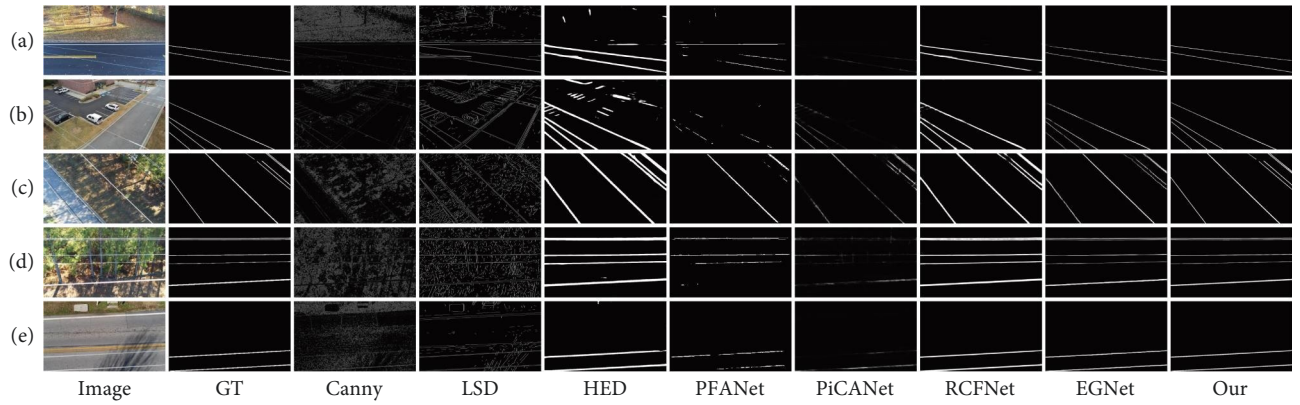

Figure 9: Visualization comparison results on TTPLA. (a–e) Five samples randomly selected from the dataset.

Table 4: Performance comparison experiments on TTPLA.

Methods	MaxF	MAE	S-measure
Canny [59]	0.1537	0.1944	0.4660
LSD [60]	0.2506	0.0715	0.5235
HED [55]	0.6034	0.0526	0.6997
PFANet [56]	0.5109	0.0390	0.5690
PiCANet [57]	0.7242	0.0405	0.6659
RCFNet [58]	0.8730	0.0227	0.8460
EGNet [32]	0.7972	0.0314	0.7322
Our	**0.9412**	**0.0217**	**0.8618**

The bold values are the optimal results for each metric.

global features and enhance the global modeling capability of the network.

5. Conclusions

In this paper, we proposed a direction consistency-guided lightweight power line detection network, which used a CAFEM to extract power line features early in the network while maintaining the location and structure information of power lines. A two-stage context-guided module processed input features with different resolutions to learn the local features, surrounding context, and global context of the power lines simultaneously, forming the initial attention information for the power lines. Then, the Gaussian kernel estimation module was used to search out the main direction of power lines in the features, and the Gaussian kernel with a similar structure to power lines was subsequently used in the decoder module to enhance the power line representation, to maintain the power line continuity. Meanwhile, low-level features were introduced to recover the power line details to ensure the accuracy of segmentation results. Moreover, PLAID was constructed and TTPLA was introduced for experiments. The ablation experiments on these two datasets verified the effectiveness of each module proposed, and the comparison experiments with other algorithms proved the superiority of our method, with more accurate power line segmentation, higher completeness, and lower mis-segmentation rate for background, which could eliminate interference.

Data Availability

The data are not publicly available due to the confidentiality of the research projects.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

This research was funded by the Incubation Project of State Grid Jiangsu Electric Power Co., Ltd., grant number JF2023012.

References

[1] Y. S. Dosso, E. Rizcallah, F. Kwamena, R. Goubran, and J. R. Green, "Deep Learning for Segmentation of Critical Electrical Infrastructure from Vehicle-Based Images," in *2022 IEEE Electrical Power and Energy Conference (EPEC)*, pp. 241–247, IEEE, Victoria, BC, Canada, 2022.

[2] M. R. M. Asyraf, M. R. Ishak, S. M. Sapuan et al., "Potential application of green composites for cross arm component in transmission tower: a brief review," *International Journal of Polymer Science*, vol. 2020, Article ID 8878300, 15 pages, 2020.

[3] L. Wang, Z. Chen, D. Hua, and Z. Zheng, "Semantic segmentation of transmission lines and their accessories based

on UAV-taken images," *IEEE Access*, vol. 7, pp. 80829–80839, 2019.

[4] H. Zhang, W. Yang, H. Yu, H. Zhang, and G.-S. Xia, "Detecting power lines in UAV images with convolutional features and structured constraints," *Remote Sensing*, vol. 11, no. 11, Article ID 1342, 2019.

[5] G. Yan, C. Li, G. Zhou, W. Zhang, and X. Li, "Automatic extraction of power lines from aerial images," *IEEE Geoscience and Remote Sensing Letters*, vol. 4, no. 3, pp. 387–391, 2007.

[6] J. Zhang, L. Liu, B. Wang, X. Chen, Q. Wang, and T. Zheng, "High speed automatic power line detection and tracking for a UAV-based inspection," in *International Conference on Industrial Control and Electronics Engineering*, pp. 266–269, IEEE, 2012.

[7] W. Zhao, Q. Dong, and Z. Zuo, "A method combining line detection and semantic segmentation for power line extraction from unmanned aerial vehicle images," *Remote Sensing*, vol. 14, no. 6, Article ID 1367, 2022.

[8] C. Zhu'an, Z. O. U. Zilong, X. U. Zhifang, P. Jiaqi, S. Chenjing, and H. Zhiqiang, "Automatic power line extraction algorithm for aerial image under complex background," *Bulletin of Surveying and Mapping*, no. 4, pp. 37–43, 2022.

[9] Z. Haocheng, L. E. I. Junfeng, W. Xianpei et al., "Power line identification algorithm for aerial image in complex background," *Bulletin of Surveying and Mapping*, no. 7, pp. 28–32, 2019.

[10] E. Titov, O. Limanovskaya, A. Lemekh, and D. Volkova, "The deep learning based power line defect detection system built on data collected by the cablewalker drone," in *2019 International Multi-Conference on Engineering, Computer and Information Sciences (SIBIRCON)*, pp. 0700–0704, IEEE, Novosibirsk, Russia, 2019.

[11] Y. Pan, F. Liu, J. Yang et al., "Broken power strand detection with aerial images: a machine learning based approach," in *2020 IEEE International Smart Cities Conference (ISC2)*, pp. 1–7, IEEE, Piscataway, NJ, USA, 2020.

[12] L. Yang, J. Fan, B. Huo, E. Li, and Y. Liu, "PLE-Net: automatic power line extraction method using deep learning from aerial images," *Expert Systems with Applications*, vol. 198, Article ID 116771, 2022.

[13] N. Xue, T. Wu, S. Bai et al., "Holistically-attracted wireframe parsing," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2788–2797, IEEE/CVF, 2020.

[14] Z. Zhang, Z. Li, N. Bi et al., "Ppgnet: learning point-pair graph for line segment detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7105–7114, IEEE/CVF, 2019.

[15] Y. Zhai, D. Wang, M. Zhang, J. Wang, and F. Guo, "Fault detection of insulator based on saliency and adaptive morphology," *Multimedia Tools and Applications*, vol. 76, no. 9, pp. 12051–12064, 2017.

[16] Ö. E. Yetgin and Ö. N. Gerek, "A comparison of corner and saliency detection methods for power line detection," in *2017 International Artificial Intelligence and Data Processing Symposium (IDAP)*, pp. 1–5, IEEE, 2017.

[17] C. Pan, X. Cao, and D. Wu, "Power line detection via background noise removal," in *2016 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pp. 871–875, IEEE, 2016.

[18] S. Zhao, Y. Wang, Z. Yang, and D. Cai, "Region mutual information loss for semantic segmentation," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[19] P. Cronin, X. Gao, C. Yang, and H. Wang, "Charger-surfing: exploiting a power line side-channel for smartphone information leakage," in *USENIX Security Symposium*, pp. 681–698, USENIX, 2021.

[20] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014.

[21] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, IEEE, 2016.

[22] J. Candamo, R. Kasturi, D. Goldgof, and S. Sarkar, "Detection of thin lines using low-quality video from low-altitude aircraft in urban settings," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 45, no. 3, pp. 937–949, 2009.

[23] I. Golightly and D. Jones, "Visual control of an unmanned aerial vehicle for power line inspection," in *12th International Conference on Advanced Robotics*, pp. 288–295, IEEE, 2005.

[24] K. Zhu, C. Xu, Y. Wei, and G. Cai, "Fast-PLDN: fast power line detection network," *Journal of Real-Time Image Processing*, vol. 19, pp. 3–13, 2022.

[25] F. Wu, Z. Yang, X. Mo et al., "Detection and counting of banana bunches by integrating deep learning and classic image-processing algorithms," *Computers and Electronics in Agriculture*, vol. 209, Article ID 107827, 2023.

[26] L. Yang, J. Fan, S. Xu, E. Li, and Y. Liu, "Vision-based power line segmentation with an attention fusion network," *IEEE Sensors Journal*, vol. 22, no. 8, pp. 8196–8205, 2022.

[27] C. Wu, F. Shuang, H. Wang, J. Zhao, and S. Yue, "Dynamic powerlines detection for UAVs by attention fused looming detector," in *2022 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–8, IEEE, 2022.

[28] H. Liang, Y. Yin, X. Wang, S. Li, H. Liang, and S. Li, "A new detection method of overhead power line based on HED algorithm," in *International Conference on Cognitive based Information Processing and Applications (CIPA 2021)*, J. Jansen, B. Liang, and H. Ye, Eds., vol. 85 of *Lecture Notes on Data Engineering and Communications Technologies*, pp. 491–499, Springer, Singapore, 2022.

[29] Y. Guo, Z. Pang, J. Du, F. Jiang, and Q. Hu, "An improved AlexNet for power edge transmission line anomaly detection," *IEEE Access*, vol. 8, pp. 97830–97838, 2020.

[30] C. Xu, Q. Li, Q. Zhou, S. Zhang, D. Yu, and Y. Ma, "Power line-guided automatic electric transmission line inspection system," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1–18, 2022.

[31] Y. Tian, Q. Wang, Z. Guo et al., "A hybrid deep learning and ensemble learning mechanism for damaged power line detection in smart grids," *Soft Computing*, vol. 26, pp. 10553–10561, 2021.

[32] S. Vemula and M. Frye, "Mask R-CNN powerline detector: a deep learning approach with applications to a UAV," in *2020 AIAA/IEEE 39th Digital Avionics Systems Conference (DASC)*, pp. 1–6, IEEE, San Antonio, TX, USA, 2020.

[33] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: a deep convolutional encoder-decoder architecture for image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, pp. 2481–2495, 2017.

[34] H. Zhao, X. Qi, X. Shen, J. Shi, and J. Jia, "ICNet for real-time semantic segmentation on high-resolution images," in *Computer Vision – ECCV 2018*, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds., vol. 11207 of *Lecture Notes in Computer Science*, pp. 418–434, Springer, Cham, 2018.

[35] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *2017 IEEE Conference on Computer*

Vision and Pattern Recognition (CVPR), pp. 6230–6239, IEEE, 2017.

[36] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang, "Bisenet: bilateral segmentation network for real-time semantic segmentation," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 325–341, ECCV, 2018.

[37] C. Yu, C. Gao, J. Wang, G. Yu, C. Shen, and N. Sang, "BiSeNet V2: bilateral network with guided aggregation for real-time semantic segmentation," *International Journal of Computer Vision*, vol. 129, no. 11, pp. 3051–3068, 2021.

[38] M. Fan, S. Lai, J. Huang et al., "Rethinking bisenet for real-time semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9716–9725, IEEE/CVF, 2021.

[39] T. Wu, S. Tang, R. Zhang, J. Cao, and Y. Zhang, "CGNet: a light-weight context guided network for semantic segmentation," *IEEE Transactions on Image Processing*, vol. 30, pp. 1169–1179, 2020.

[40] A. G. Howard, M. Zhu, B. Chen et al., "Mobilenets: efficient convolutional neural networks for mobile vision applications," 2017.

[41] K. Sun, M. Li, D. Liu, and J. Wang, "Igcv3: interleaved low-rank group convolutions for efficient deep neural networks," 2018.

[42] X. Zhang, X. Zhou, M. Lin, and J. Sun, "Shufflenet: an extremely efficient convolutional neural network for mobile devices," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6848–6856, IEEE, 2018.

[43] N. Ma, X. Zhang, H. T. Zheng, and J. Sun, "Shufflenet v2: practical guidelines for efficient cnn architecture design," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 116–131, ECCV, 2018.

[44] C. Shi, L. Lin, J. Sun, W. Su, H. Yang, and Y. Wang, "A lightweight YOLOv5 transmission line defect detection method based on coordinate attention," in *2022 IEEE 6th Information Technology and Mechatronics Engineering Conference (ITOEC)*, pp. 1779–1785, IEEE, 2022.

[45] W. Yang, W. Luo, J. Mao, Y. Fang, and J. Bei, "Substation meter detection and recognition method based on lightweight deep learning model," in *Proceedings of the SPIE*, vol. 12508, pp. 199–207, SPIE, 2022.

[46] F. Wu, J. Duan, P. Ai, Z. Chen, Z. Yang, and X. Zou, "Rachis detection and three-dimensional localization of cut off point for vision-based banana robot," *Computers and Electronics in Agriculture*, vol. 198, Article ID 107079, 2022.

[47] C. Ma, Y. Yang, Y. Wang, Y. Zhang, and W. Xie, "Open-vocabulary semantic segmentation with frozen vision-language models," 2022.

[48] G. Cheng, P. Sun, T.-B. Xu, S. Lyu, and P. Lin, "Local-to-global information communication for real-time semantic segmentation network search," 2023.

[49] R. Liu, J. Lehman, P. Molino et al., "An intriguing failing of convolutional neural networks and the coordconv solution," *Advances in Neural Information Processing Systems*, vol. 31, 2018.

[50] Q. Hou, D. Zhou, and J. Feng, "Coordinate attention for efficient mobile network design," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13713–13722, IEEE/CVF, 2021.

[51] P.-T. de Boer, D. P. Kroese, S. Mannor, and R. Y. Rubinstein, "A tutorial on the cross-entropy method," *Annals of Operations Research*, vol. 134, no. 1, pp. 19–67, 2005.

[52] J. Nilsson and T. Akenine-Möller, "Understanding," 2006.

[53] A. Torralba, B. C. Russell, and J. Yuen, "Labelme: online image annotation and applications," *Proceedings of the IEEE*, vol. 98, no. 8, pp. 1467–1484, 2010.

[54] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," 2017.

[55] S. Xie and Z. Tu, "Holistically-nested edge detection," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 1395–1403, IEEE, 2015.

[56] T. Zhao and X. Wu, "Pyramid feature attention network for saliency detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3085–3094, IEEE, 2019.

[57] N. Liu, J. Han, and M. H. Yang, "Learning pixel-wise contextual attention for saliency detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3089–3098, IEEE, 2018.

[58] Y. Liu, M. M. Cheng, X. Hu et al., "Richer convolutional features for edge detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3000–3009, IEEE, 2017.

[59] J. Canny, "A computational approach to edge detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, no. 6, pp. 679–698, 1986.

[60] R. G. Von Gioi, J. Jakubowicz, J.-M. Morel, and G. Randall, "LSD: a line segment detector," *Image Processing on Line*, vol. 2, pp. 35–55, 2012.