PLGAN: Generative Adversarial Networks for Power-Line Segmentation in Aerial Images

Rabab Abdelfattah, Xiaofeng Wang, *Member, IEEE*, and Song Wang, *Senior Member, IEEE*

***Abstract*— Accurate segmentation of power lines in various aerial images is very important for UAV flight safety. The complex background and very thin structures of power lines, however, make it an inherently difficult task in computer vision. This paper presents PLGAN, a simple yet effective method based on generative adversarial networks, to segment power lines from aerial images with different backgrounds. Instead of directly using the adversarial networks to generate the segmentation, we take their certain decoding features and embed them into another semantic segmentation network by considering more context, geometry, and appearance information of power lines. We further exploit the appropriate form of the generated images for high-quality feature embedding and define a new loss function in the Hough-transform parameter space to enhance the segmentation of very thin power lines. Extensive experiments and comprehensive analysis demonstrate that our proposed PLGAN outperforms the prior state-of-the-art methods for semantic segmentation and line detection.**

***Index Terms*— Power-line segmentation, generative adversarial networks, image segmentation, aerial images, line detection.**

I. Introduction

WHILE unmanned aerial vehicles (UAVs) have been used in many recreational, photography, commercial, and military applications, their flight safety may be threatened by the widespread power lines (PLs) [1]. Hitting a PL may not only destroy the UAVs but also damage power grids and electrical properties as well. Given their very thin structures, however, PLs are prone to be missed by many detection sensors. To enable UAVs to detect and localize PLs during flight, this paper presents a new computer-vision approach aiming to accurately segment PLs from *aerial images* that are taken by the cameras mounted on UAVs.

PL segmentation from aerial images is very challenging. From a bird's-eye view, the background of aerial images can be any place, e.g., desert, lakes, mountains, and cities, which shows significant variety and complexity. Moreover, PLs and their surrounding background may share a very similar color in many cases and, therefore, are difficult to distinguish from local image information. Finally, PLs have very thin structures and only cover a very small portion of the image, e.g., one- or few-pixel wide in aerial images. As a result, the PL segmentation is vulnerable to being fragmented, leading to poor segmentation performance.

There have been many deep-learning based algorithms developed for achieving state-of-the-art performance on general-purpose line-segment detection [2], [3], [4], [5], most of which rely on the saliency of lines and joint inference of junctions. Both of these properties, however, do not hold for PLs in most aerial images. The recent AFM model [6] detects line segments by constructing an attraction field map instead of inferring junctions. Nevertheless, it cannot handle complex backgrounds in aerial images, as validated in our later experiments. PL segmentation can be treated as a kind of semantic image segmentation, for which many advanced deep neural networks, such as FCN [7] and DeepLab [8], [9], have been developed with state-of-the-art performance on public image dataset, such as Cityscape and PASCAL VOC. However, without considering the shape and inter-pixel relations, these semantic segmentation networks cannot accurately capture very thin PLs with a similar color to the surrounding background in aerial images.

To find the inter-pixel relations and enforce the global consistency between pixels, in this paper, we propose employing generative adversarial networks (GANs) as a backbone for PL segmentation. The main motivation is to leverage the min-max loss of GANs to help 1) generate a natural (real) image that accurately reflects the relationship between adjacent pixels, and 2) create a high-quality feature embedding for semantic image segmentation. Specifically, this paper presents a new PLGAN (PL Generative Adversarial Networks) to segment PLs from aerial images by employing adversarial learning. In the proposed PLGAN, we first include a multi-task encoder-decoder network to generate an image with highlighted PLs. Then, we extract the last feature representation (i.e., the one right before the output layer) of the decoder network and embed it in a semantic segmentation network to improve PL segmentation. We define comprehensive loss functions, including adversarial, geometry, and cross-entropy ones, for PLGAN training. Furthermore, we include a loss function in the Hough transform parameter space to highlight the long-thin nature of PLs. Extensive experiments, including ablation studies and comparison experiments with prior methods, on the public TTPLA dataset [10] and Massachusetts roads dataset for road

Manuscript received 3 January 2022; revised 15 April 2023 and 30 July 2023; accepted 17 September 2023. Date of publication 30 October 2023; date of current version 20 November 2023. This work was supported by the National Science Foundation under Grant ECCS1830512. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Xiaolin Wu. *(Corresponding author: Rabab Abdelfattah.)*

Rabab Abdelfattah is with the School of Computer Science and Computer Engineering, University of Southern Mississippi, Hattiesburg, MS 39406 USA (e-mail: rabab.abdelfattah@usm.edu).

Xiaofeng Wang is with the Department of Electrical Engineering and Computing, University of South Carolina, Columbia, SC 29208 USA (e-mail: wangxi@cec.sc.edu).

Song Wang is with the Department of Computer Science and Engineering, University of South Carolina, Columbia, SC 29208 USA (e-mail: songwang@cec.sc.edu).

Digital Object Identifier 10.1109/TIP.2023.3321465

segmentation [11], validate the effectiveness of the proposed method.

Our main contributions are summarized as below.

- A novel PLGAN network has been developed to segment extremely thin lines, such as power lines (PLs) from aerial images, under complex backgrounds. To the best of our knowledge, this is the first generative adversarial network (GAN) developed for line structure segmentation. The novelty of this approach lies in utilizing PL-highlighted images for the discrimination process and incorporating a semantic decoder, which takes highly representative embedding vectors as inputs to generate semantic images. The purpose of employing adversarial training is to produce realistic PL-highlighted images that further differentiate PL pixels from complex backgrounds. Leveraging the advantages of GANs [12], an embedding vector can be produced through adversarial training to capture the features and structural information of the input image, followed by a semantic decoder to learn and perform semantic segmentation based on this embedding vector.
- A new loss function is introduced in the Hough transform parameter space. We use the Hough transform to map each pixel in the segmentation image to a sinusoidal curve in the parameter space. If a pixel on PLs is missing in the segmentation-image space, multiple points on the related sinusoidal curve will be missing in the parametric space. The Hough transform loss is defined to penalize those missing points in the parameter space, instead of one missing pixel in segmentation-image space. By doing so, the penalty for missing pixels in the segmentation-image space will be amplified, which forces the model to correct the flawed pixels. Meanwhile, the intersection of sinusoidal curves at the same points in the parameter space indicates that the associated pixels belong to the same PL. If any curves are missing in the parameter space, it will lead to a reduction in the intensity of the intersection points. Such a reduction implies that some pixels are missing in the segmentation-image space and the network must be penalized to learn how to identify and recover those missing curves. Thus, the proposed Hough transform loss can enhance global consistency for PLs in the segmentation-image space.
- Extensive experiments have been conducted to evaluate the performance of PLGAN on the TTPLA dataset and the Massachusetts Roads dataset. PLGAN outperforms the state-of-the-art semantic segmentation models under most evaluation metrics. In particular, compared with models with similar sizes, PLGAN achieves the best scores under all metrics.

For the remainder of the paper, Section II provides a brief overview of related work. Section III elaborates on describing the proposed PLGAN. Section IV reports the experimental results, followed by a brief conclusion in Section V.

II. Related Work

The related work is discussed in four parts: power lines (PLs), line segment detection, semantic segmentation, and GANs.

A. Power Lines

Most existing PL-related datasets were designed with specific properties to simplify PL detection, such as synthetic PLs [13], manually cropping aerial images to obtain sub-images focused on PLs [14], and capturing images from ground [15], to name a few. Compared with these datasets, the TTPLA dataset we use in this paper is more challenging and practical. It includes aerial images with very complex backgrounds and wide varieties in zoom levels, view angles, time during a day, as well as weather conditions [10].

Most existing work on PL detection adopts traditional computer vision methods [16], [17], [18], [19], [20], which have multiple drawbacks. First, it is often assumed that the PLs are parallel and straight so that context-assisted information can be used to extract PLs [17], [19], while this assumption may not hold in practice. Second, extracting edge maps with traditional approaches requires good contrast between the PLs and the surrounding background, which can only be achieved in ideal cases [21]. In practice, the color of the PLs and the background could be very similar in aerial images. Third, traditional methods usually rely on predefined hyper-parameters to generate meaningful results. However, defining these hyper-parameters is very challenging, especially for those datasets with images taken in a wide range of conditions (e.g., different zoom levels, points of view, background, light, and contrast).

Recently, deep-learning based methods were investigated [13], [14], [22], [23], [24], [25], [26] for PL detection. Yetgin et al. [22] proposed an end-to-end CNN architecture with a randomly initialized softmax layer for jointly fine-tuning the feature extraction and binary classification – PL and non-PL background are classified at the image level. Yetgin et al. further developed a feature classification method for PL segmentation, where features are extracted from the intermediate stages of the CNN. In [27], a CNN-based classifier is developed to identify the input-image patch that contains PL and then uses Hough transform as the post-processing to localize the PLs in each patch. In [28], a deep CNN architecture with fully connected layers is proposed for PL segmentation, where the CNN inputs are histogram-of-gradient features – a sliding window is moved over each patch to get a classification of PL or not. In [29], a UNET architecture is trained to segment PLs based on a generalized focal loss function that uses the Matthews correlation coefficient [30] to address the class imbalance problem. In [23], an attentional convolutional network is proposed for pixel-level PL detection, and it consists of an encoder-decoder information fusion module and an attention module, where the former fuses the semantic information and the location information while the latter focuses on PLs. In [24], dilated conventional networks with different architectures are tried to find the best architecture over a finite space of model parameters. Choi et al. [25] proposed a weakly supervised learning network for pixel-level PL detection using only image-level classification labels. However, besides the simplicity of the datasets as mentioned before, most of these CNN-based works formulate the problem as pixel-wise classification with convolutional neural networks (CNNs) and do not sufficiently

consider global consistency in detection, which is essential in detecting very thin structures [26].

B. Line Segment Detection

Significant progress has been achieved in line segment detection using deep neural networks in recent years. Most deep line detection approaches rely on junction information to locate valid line segments: some methods jointly detect the junctions and line segments [2], [3], while others detect only the junctions and then employ sampling methods to deduce the line segments [4], [5]. However, these methods are not applicable to our task since PLs in aerial images may not always be straight and often lack junctions.

C. Semantic Segmentation

Deep neural networks for semantic segmentation [31], [32] rely on pooling layers to reduce the spatial resolution in the deepest FCN layers. Consequently, predictions around segmentation boundaries often suffer from inadequate contextual information [26], [33], [34], [35]. Dilated convolutions have been introduced to capture larger contextual information [8], [31], [36], [37], [38], which, however, still cannot generate global context just from a few neighboring pixels [34]. Encoder-decoder structures have emerged to overcome the drawbacks of atrous convolutions [39], [40], [41]. However, the prediction accuracy is still limited when recovered from the fused features [42]. In addition, the softmax cross-entropy loss limits semantic segmentation performance [26], [33] by ignoring the correlation between pixels. Many of these limitations can be observed in segmenting very thin PLs, and we will include several of the above methods in our comparison experiments.

D. Semantic Segmentation Based GAN

Generative Adversarial Networks (GANs) [43] have been widely used in image translation [44], [45], super-resolution [46], inpainting [47], salient object detection [48], and image editing/manipulation [49]. There are also models that utilize GANs for creating semantic segmentation images. In the early research presented in [50], the authors introduced an approach that utilizes adversarial networks for performing semantic segmentation for the colored input image. In [51], the authors employ GANs and transfer learning for the segmentation of biomedical cell images. Hung in [52] proposed an adversarial learning scheme for semi-supervised semantic segmentation. It is worth mentioning, however, that directly applying GANs for segmentation may not be desirable for two reasons: (i) GANs usually employ softmax loss at the output layer, which prevents the networks from expressing uncertainties when generating semantic images [53]; (ii) The softmax probability vectors cannot produce exact zeros/ones, while the discriminator requires sharp zeros/ones. As a result, the discriminator may unnecessarily learn more complicated geometrical discrepancies by examining the small, but always existing, value gap between the distributions of the fake and real samples. In this paper, we embed GAN-extracted features for enhancing PL segmentation, instead of directly discriminating the semantic images.

III. PLGAN Approach

Notations: Let $I_r \in \mathbb{R}^{w\times h\times c}$ denote the input image, where $w \times h$ is the dimensions of the input image and c is the number of channels. Let $\hat{I}_s \in \mathbb{R}^{w\times h}$ be the semantic output of PLGAN as shown in Figure 1 and $\hat{I}_p \in \mathbb{R}^{w\times h\times c}$ be the PL-highlighted image (or "fake image") projected from the embedding vector $E_m(I_r)$. For the ground-truth of the PL-highlighted image, we simply set the intensity of PL pixels in an image to zero. I_s and I_p are the ground truth (GT) of $\hat{I}_s$ and $\hat{I}_p$, respectively. Let $\phi : \mathbb{R}^{w\times h\times c} \to \mathbb{R}^{h\times w\times c}$ denote the geometry transformation on an image. Given an image I, the transformed image is denoted as $I^t = \phi(I)$. To ensure that the transformed images have the appropriate dimensions as the inputs to PLGAN, we assume $w = h$. Given a matrix P, $[P]_{ij}$ denotes the entry at the ith row and the jth column of P. Accordingly, given an image I, $[I]_{ij}$ denotes the value at pixel (i, j) in the image. Given two cascaded functions or networks G and ϕ, $G \circ \phi(\cdot) = G(\phi(\cdot))$. $\|\cdot\|_1$ is the L_1 norm to calculate the absolute difference on each pixel.

A. PLGAN Structure (G_{PL})

Our objective is to develop a deep neural network that predicts the semantic image $\hat{I}_s$ based on the input image I_r. The proposed PLGAN structure consists of the PL-aware generator, two discriminators, and the semantic decoder. The discriminators are trained in an adversarial way against the PL-aware generator and the semantic decoder. As shown in Figure 1, the input image I_r is transformed into a latent space by the Embedder network E_m. The resulting embedding vector $E_m(I_r)$ contains context, appearance, and geometry information. This vector is mapped back to the image space through the output layer of the PL decoder and the semantic decoder for the PL-highlighted image $\hat{I}_p$ and the semantic image $\hat{I}_s$, respectively. During training, PLGAN will learn the features of the PL pixels and distinguish them from the background pixels based on adversarial loss functions. During testing, there are no additional overhead or post-processing steps; only E_m and S networks are used to generate semantic images.

The **PL-aware generator** (G) consists of the encoder and the PL decoder with the residual blocks [54] in the middle. The encoder and the PL decoder are composed of a sequence of convolution layers and transpose convolutional layers with a stride of 2, respectively, both followed by batch-normalization and ReLU activation, as shown in Figure 1.

The **semantic decoder** (S) outputs semantic images $\hat{I}_s$. At the same time, the discriminator focuses on the PL-highlighted images and their GT, which are color images. By doing so, the benefits of adversarial training can be fully explored, which, as discussed in Subsection II-D, cannot be achieved by directly applying GANs (PL-aware generator and discriminator only). The semantic decoder consists of a set of convolution, batch-normalization, leaky-ReLU layers, and

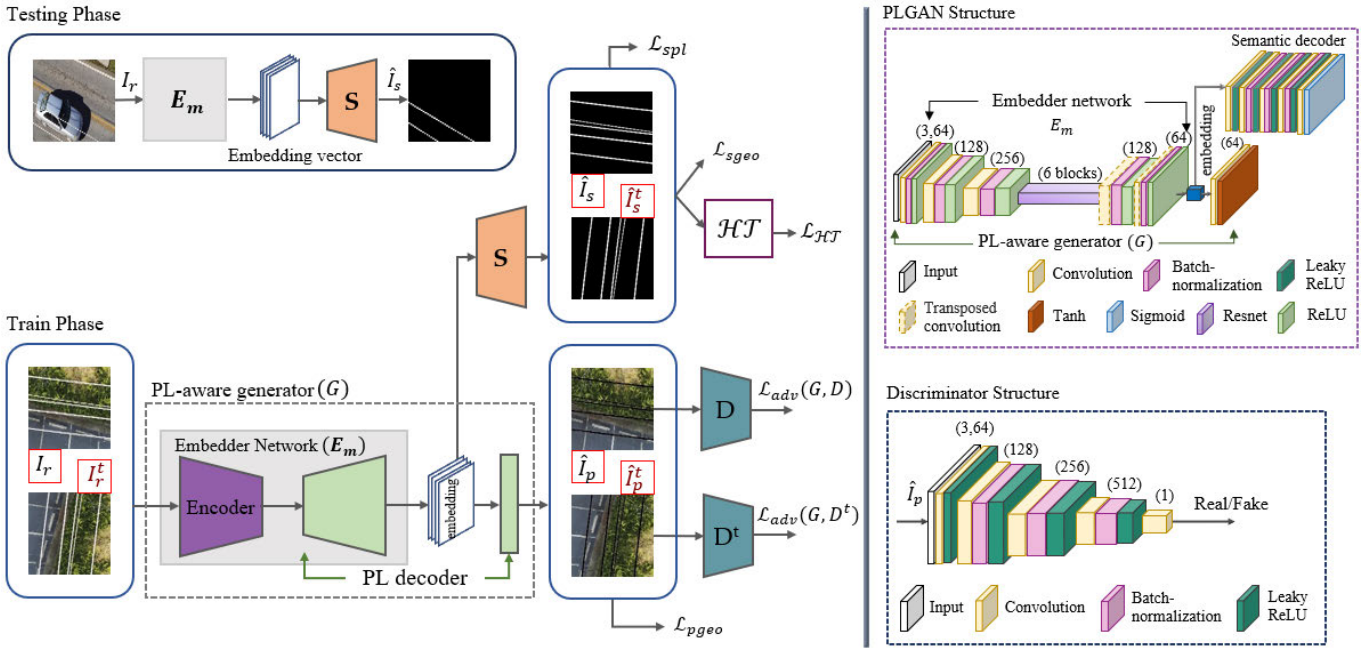

Fig. 1. An illustration of PLGAN framework. PLGAN consists of the PL-aware generator G, two discriminators D and D^t, and the semantic decoder S. The PL-aware generator contains the encoder and PL-decoder. The embedder network E_m, included in the generator, consists of the encoder and the PL-decoder except for the last output layer. The input to PLGAN is RGB image I_r and its transformed input image I_r^t (They are applied individually, not at the same time). The output of PLGAN is the semantic image for PLs $\hat{I}_s$. The G, S, D, and D^t forms adversarial training to generate PL-highlighted image $\hat{I}_p$ from G and the embedding vector from E_m. The embedding vector, which carries the context, appearance, and geometry information, is used as the input to S. The generated PL-highlighted images $\hat{I}_p$ and the transformed $\hat{I}_p^t$ are the inputs to D and D^t, respectively. The PL-aware generator and the semantic decoder are jointly trained by the combination of adversarial, semantic, geometry, and Hough transform loss functions. There is no overhead during testing, only E_m and S networks are used to generate semantic images.

a nonlinear sigmoid layer as the output layer, as shown in Figure 1.

The **adversarial discriminator** (D) is to distinguish the PL-highlighted image $\hat{I}_p$ from its GT I_p. Notice that $\hat{I}_p$ is very similar to the input image I_r, except that the PL pixels are highlighted. The PL area in $\hat{I}_p$ has a high-frequency structure because of sharp changes in intensity from the background pixels to the highlighted pixels. Given this high-frequency nature, the Markovian discriminator structure is used for its efficiency in tracking high-frequency structures [44]. The Markovian discriminator maps $\hat{I}_p$ at the patch level (i.e., patches are individually quantified to the fake or real value) and considers the structural loss, such as structural similarity, feature matching, and conditional random field, which will help compensate the loss of $\hat{I}_p$ at low frequencies. With these benefits, the discriminator is able to push the PL-aware generator to create more natural PL-highlighted images [55]. Besides D, an additional discriminator (D^t) is added to discriminate the transformed PL-highlighted image $\hat{I}_p^t$ and the GT I_p^t, which has a similar structure to D as shown in Figure 1. Particularly in cases with dark backgrounds, the Markovian discriminator may struggle to distinguish the high-frequency pixels of power lines from the background. In these cases, the semantic decoder will help detect those pixels by penalizing the prediction errors through semantic and Hough transform loss, which plays a crucial role in correcting the discriminator. Since the GAN (consisting of both the generator and discriminator) and the semantic decoder are trained together within an end-to-end framework, the overall performance relies on the combination of these two components, instead of GAN only.

B. Objective Formulation

The loss functions for different modules in PLGAN are defined as follows.

1) Adversarial Loss: The adversarial loss is applied to encourage G to fool the discriminator D by generating images that look similar to the real images. While, D is trained to distinguish between the real images (I_p) and fake images ($\hat{I}_p$). The least-square loss function is chosen for our training, instead of binary cross-entropy [56], for more stable training and convergence [57]. The adversarial loss is defined as:

$$\mathcal{L}_{adv}(G, D; I_r, I_p) = \frac{1}{2}\mathbb{E}_{I_p}\left[(D(I_p))^2\right] + \frac{1}{2}\mathbb{E}_{I_r}\left[(1 - D \circ G(I_r))^2\right] \quad (1)$$

where $\mathbb{E}_{I_p}$ and $\mathbb{E}_{I_r}$ are the empirical estimated expectations. The discriminator D is to maximize $\mathcal{L}_{adv}$ and G is to minimize this loss, which formulates adversarial training.

2) Semantic Loss: The cross entropy loss between I_s and $\hat{I}_s$ is defined as follows:

$$\mathcal{L}_{spl}(E_m, S; I_r, I_s) = \frac{\sum_{(i,j)\in\mathcal{N}} \left([I_s]_{ij} \log([\hat{I}_s]_{ij}) + (1 - [I_s]_{ij}) \log(1 - [\hat{I}_s]_{ij})\right)}{-|\mathcal{N}|} \quad (2)$$

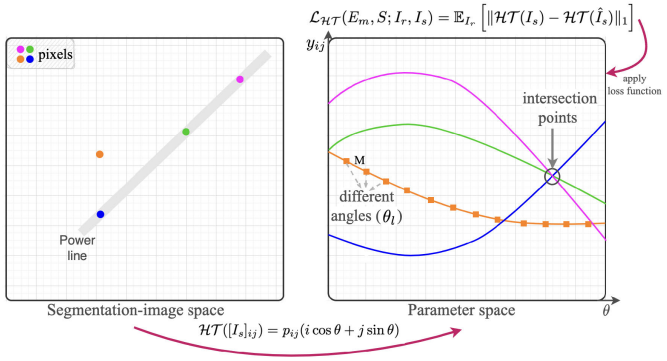

Fig. 2. Illustration for applying our proposed Hough Transform Loss on the parameter space. The pixels located on the same line (red, green, and blue) in the segmentation-image space should intersect at the same point in the parameter space. The set of angles (θ_l) in the parametric space is partitioned into M pieces, resulting in M outputs for each pixel in the segmentation-image space, i.e., the orange curve.

where $\mathcal{N}$ is the pixel set of interest (e.g., pixels belonging to PLs), $|\mathcal{N}|$ is the number of elements in $\mathcal{N}$, and $\hat{I}_s = S \circ E_m(I_r)$. It is worth mentioning that the semantic loss is determined pixel by pixel, which may not be able to capture the correlations between pixels. In this case, missing one PL pixel may lead to spatially-disjoint object segments, given that PLs are very thin in aerial images (e.g., 1 pixel width). To address this issue, we introduce the Hough transform loss function, which will be discussed next.

3) Hough Transform Loss: The motivation of using Hough transform loss is to force PLGAN to find and correct the flawed pixels along PLs so as to ensure global consistency for each PL. Each pixel in the semantic image is mapped to a sinusoidal curve in the parametric space by the modified Hough transform as shown in Figure 2

$$\mathcal{HT}([I_s]_{ij}) = p_{ij}(i\cos\theta + j\sin\theta) \tag{3}$$

where (i, j) is the pixel coordinate in the semantic image I_s, $p_{ij} \in [0, 1]$ is the confidence score at pixel (i, j) which indicates the likelihood of the pixel belonging to the line, $\theta \in [0, \theta_{\max})$ is the angle parameter, and $\theta_{\max}$ is the maximum value of θ (e.g., $\theta_{\max} = \pi$). During training, p_{ij} will eventually approach the neighborhood of 1 or 0, indicating that (i, j) belongs to the PLs or not, respectively. Otherwise, it will lead to a large loss in the parameter space and force PLGAN to refine its prediction. In practice, we partition the set $[0, \theta_{\max})$ into M pieces and therefore one pixel in segmentation-image space will result in M outputs, $y_{ij}(\theta_l)$, in parameter space, where $\theta_l = \frac{l\theta_{\max}}{M}$ for $l = 0, 1, \cdots, M-1$. The motivation for partitioning the Hough angle space into M segments is to emphasize the penalty in Hough transform parameter space when a pixel in power lines is missing in the segmentation-image space. In practice, one pixel in the segmentation-image space will result in M outputs in the parameter space, where each output corresponds to a segment of the angle space. By partitioning the angle space, missing one pixel on a power line in the segmentation-image space implies missing M points in the parameter space, which amplifies the penalty M times in the parameter space. This forces the PLGAN to correct the flawed pixels and recover the missing pixels in the segmentation-image space. The Hough transform Loss is defined as follows.

$$\mathcal{L}_{\mathcal{HT}}(E_m, S; I_r, I_s) = \mathbb{E}_{I_r}\left[\|\mathcal{HT}(I_s) - \mathcal{HT}(\hat{I}_s)\|_1\right] \tag{4}$$

with $\hat{I}_s = S \circ E_m(I_r)$, where $[\mathcal{HT}(I_s)]_{ij} = \mathcal{HT}([I_s]_{ij})$. From the other point of view, the pixels belonging to the same PL in segmentation-image space are intersected as sinusoidal curves into the parameter space and accumulated as a value into the same cell into the discrete parametric space. Therefore, the intersection points have strong intensities as a result of intersecting more than one curve into the same point in the parameter space. Each intersection point includes two specifics: –The intersection point represents multiple related pixels belonging to the same PL in the segmentation-image space. – If the intensity of the intersection point is reduced as a result of missing one or more curves in the parameter space, the network should be penalized to learn to find the missing curves. Hence, the missing pixels in the segmentation-image space are recovered. Consequently, our Hough transform loss function enhances global consistency for the power lines in the segmentation-image space.

It is worth noticing that our proposed HT loss function differs from the HT loss function proposed in [58], although both loss functions are applied on HT parameter space. The HT loss function presented in [58] is restricted with two assumptions, including a pre-defined number of lines in the scenes, and a single lane line is predicted in each output channel. The intersecting points are identified in the parameter space for each channel separately, and the loss function is calculated based on these intersection points. Additionally, the HT loss function optimizes only when the predicted probability of a segmented lane is larger than a specified threshold which means that it may not be applied to all pixels in the segmentation space. In contrast to our method, all segmentation PL lines are predicted in a single channel and are mapped into a single HT parameter space without constraints on the number of power lines per scene. Moreover, our proposed HT loss function is applied on the whole sinusoidal curves in the parameter space and optimizes regardless of the model's confidence level.

4) Geometry Loss: According to [10], the PLs, on average, take 1.68% of the total pixels in an aerial image. In addition, the color of PLs in aerial images may be close to the background. Both facts indicate that the visual evidence of PLs is very weak. There is a possibility of generating trivial $\hat{I}_p$ that is very similar to the background in colors and styles while removing or decreasing the foreground. The discriminator may not be able to correctly identify the flawed pixels in $\hat{I}_p$ from the GT I_p due to the high similarity between $\hat{I}_p$ and I_p at most pixels. To address this issue, we add penalties in geometry space that force the training to correct failures in the local regions of PLs after geometry transformation. Geometric transformations refer to operations such as flip and rotation, which do not change the image's semantic structure [56]. If a model is geometrically consistent, the output image generated by the model in response to an input image should preserve similar, if not exactly the same, features/structures to the

output image in response to the geometrically transformed for the input image. However, GANs alone usually do not have this property because they do not have any constraints to enforce geometric consistency. Therefore, inspired by the work in [56], we introduce the geometry loss function to penalize the difference between two output images (the PL-highlighted image $\hat{I}_p = G(I_r)$ output of the generator from the original input image, and the output from the inverse of its transformed image $\phi^{-1} \circ G \circ \phi(I_r)$), such that the model can be trained in a way towards enhanced geometric consistency in the GAN structure. In PLGAN, we also consider geometry consistency between the semantic image $\hat{I}_s = S \circ E_m(I_r)$ and the inverse of its transformed semantic image $\phi^{-1} \circ S \circ E_m \circ \phi(I_r)$. The geometry loss is defined as follows:

$$\begin{aligned}
&\mathcal{L}_{pgeo}(G; I_r) \\
&= \mathbb{E}_{I_r}\left[\|G(I_r) - \phi^{-1} \circ G \circ \phi(I_r)\|_1\right] \\
&\quad + \mathbb{E}_{I_r}\left[\|G \circ \phi(I_r) - \phi \circ G(I_r)\|_1\right] \\
&\mathcal{L}_{sgeo}(E_m, S; I_r) \\
&= \mathbb{E}_{I_r}\left[\|S \circ E_m(I_r) - \phi^{-1} \circ S \circ E_m \circ \phi(I_r)\|_1\right] \\
&\quad + \mathbb{E}_{I_r}\left[\|S \circ E_m \circ \phi(I_r) - \phi \circ S \circ E_m(I_r)\|_1\right].
\end{aligned}$$

With the penalty on the geometry loss, it is unlikely that G and $G \circ \phi$ both fail at the same location. Instead, they co-regularize each other to keep geometry-consistency [56]. So do $S \circ E_m$ and $S \circ E_m \circ \phi$. Similarly, we can define the adversarial loss, the semantic loss, and the Hough transform loss in the transformed domain as $\mathcal{L}_{adv}(G, D^t; I_r^t, I_p^t)$, $\mathcal{L}_{spl}(E_m, S; I_r^t, I_s^t)$, and $\mathcal{L}_{\mathcal{HT}}(E_m, S; I_r^t, I_s^t)$, respectively, with $I_r^t = \phi(I_r)$, $I_s^t = \phi(I_s)$, and $I_p^t = \phi(I_p)$. D^t is the discriminator for the transformed generated image I_p^t.

The **overall loss** function can be defined as follows:

$$\begin{aligned}
&\mathcal{L}_{G_{PL}}(G, D, D^t, S; I_r, I_s, I_p) \\
&= \mathcal{L}_{adv}(G, D; I_r, I_p) + \mathcal{L}_{adv}(G, D^t; I_r^t, I_p^t) \\
&\quad + \lambda_{spl}\left(\mathcal{L}_{spl}(E_m, S; I_r, I_s) + \mathcal{L}_{spl}(E_m, S; I_r^t, I_s^t)\right) \\
&\quad + \lambda_{\mathcal{HT}}\left(\mathcal{L}_{\mathcal{HT}}(E_m, S; I_r, I_s) + \mathcal{L}_{\mathcal{HT}}(E_m, S; I_r^t, I_s^t)\right) \\
&\quad + \lambda_{geo}\left(\mathcal{L}_{pgeo}(G; I_r) + \mathcal{L}_{sgeo}(E_m, S; I_r)\right) \qquad (5)
\end{aligned}$$

IV. Experiments

The experimental results are presented in this section, with comparisons to the state-of-the-art methods.

A. Datasets

TTPLA [10] is a public dataset that contains aerial images for PLs from different zoom levels and view angles, collected at different times and locations with different backgrounds. TTPLA dataset contains 8,083 instances of PLs, which take only 154M pixels, 1.68% of the total number of pixels in this dataset [10]. This dataset contains about 1,100 images. We used 905 training images, augmented by vertical/horizontal flipping, and 217 images for the test set. Each instance of PL is carefully annotated by a polygon using LabelME [63]. TTPLA also provides polygonal annotations of all the transmissions present in each image, and an instance of PL is usually considered to be ended when it enters the annotated polygon of the transmission tower, as shown in the second column of Figure 3. Since there are few public PL datasets available [10], [64], we also considered Massachusetts Roads dataset [11] instead to further evaluate the performance of PLGAN. Although Massachusetts Roads dataset has different context and features compared to TTPLA, it shares some similarities with PL datasets, such as: (i) It is still an aerial dataset; (ii) It contains roads that are thin and take only a small portion of the overall image (the pixels associated with the roads are a small percentage of the total number of pixels in the image). Thus, evaluating PLGAN on the Massachusetts Roads dataset can still provide valuable insights into the model's ability to segment thin structures in aerial images, even when the objects are not specifically PLs. This dataset is used for road segmentation, consisting of 1,108 training and 49 test images, including urban and rural neighborhoods with pixel-level annotations.

B. Implementation Details

The proposed PLGAN is implemented using PyTorch. The weights of all sub-nets are initialized based on normal distribution using the Xavier method with zero mean and gain 0.02. They are jointly optimized using Adam with the first and the second momentum setting to 0.5 and 0.999, respectively. The entire model is trained for 200 epochs with the image size of 512×512. The learning rate starts with 1×10^{-4} for the first 100 epochs and decays to zero during the second 100 epochs. All models are trained from scratch. The ground-truth of the PL-highlighted images is obtained by simply setting the intensity of the PL pixels in the images to zero. PLGAN uses ResNet as a backbone, following CyclicGAN, GcGAN, and Pix2Pix GAN, and the training starts with a Gaussian distribution (mean 0 and std 0.02). To ensure a fair comparison, each semantic segmentation model is executed on an Nvidia GeForce RTX 3090 GPU card.

C. Evaluation Metrics

We adopt a total of eight metrics to evaluate the detection performance of our model. Precision, recall, and intersection-over-union (IoU) are the widely used metrics in semantic segmentation [65]. Also, we consider F score as an evaluation metric which is the harmonic mean of average precision and average recall. It is defined as $F_\beta = \frac{(1+\beta^2)Precision \times Recall}{\beta^2 Precision + Recall}$, where we assign β with two values: $\beta = 1$ following [66] and $\beta = 0.3$ to emphasize more precision over recall which follows [67]. Furthermore, we investigate the completeness (comp.), correctness (corr.), and quality as the evaluation metrics, following the previous studies on thin-object detection [66], [68], [69], [70], [71]. Under these metrics, the definition of true positives can be extended to the case that allows the predicted pixel to shift a certain distance from its ground truth. Correctness and completeness represent the extended precision and recall, respectively, while quality $= \frac{comp. \times corr.}{comp. - comp. \times corr. + corr.}$. In our experiments, we allow the shift to be 2 pixels under these three evaluation metrics, following [66] and [71].

TABLE I

QUANTITATIVE PL SEGMENTATION PERFORMANCE OF THE PROPOSED PLGAN AND THE COMPARISON METHODS ON TTPLA DATASET [10]. BOLD REPRESENTS THE HIGHEST RESULTS AND UNDERLINE REPRESENTS THE SECOND-BEST

Models	Backbone	Precision	Recall	IoU	F_1	F_β	Corr	Comp	Quality	param (M) ↓
FPN [59]	Resnet−34	0.769	0.513	0.423	0.569	0.635	0.884	0.743	0.674	23.2
UNET	Resnet-34	0.846	0.583	0.515	0.662	0.735	0.904	0.823	0.754	24.4
LinkNet [60]	Resnet-34	0.836	0.569	0.496	0.645	0.719	0.903	0.809	0.741	21.8
UNET++ [61]	Resnet-34	0.843	**0.591**	0.522	0.668	0.739	0.896	0.833	0.760	26.1
MaNet [62]	Resnet-34	0.858	0.585	0.517	0.663	0.738	**0.923**	0.810	0.759	31.8
FPN [59]	Resnet-18	0.746	0.492	0.401	0.546	0.612	0.867	0.717	0.646	13.0
DeepLabv3+ [9]	Resnet-18	0.784	0.510	0.424	0.573	0.645	0.897	0.747	0.684	12.3
UNET	Resnet-18	0.827	0.560	0.492	0.641	0.715	0.879	0.805	0.725	14.3
UNET++ [61]	Resnet-18	0.836	0.571	0.506	0.653	0.727	0.886	0.811	0.734	15.9
LinkNet [60]	Resnet-18	0.794	0.569	0.484	0.635	0.698	0.865	0.804	0.711	11.7
MaNet [62]	Resnet-18	0.844	0.587	0.516	0.663	0.735	0.906	0.816	0.755	21.7
AIFN [23]	Resnet-18	0.799	0.541	0.486	0.645	0.719	0.845	0.783	0.685	18.3
Focal-UNET [29]	Resnet-18	0.784	0.577	0.504	0.662	0.724	0.836	0.811	0.700	18.4
Pix2pix [44]	Resnet-6	0.822	0.577	0.509	0.663	0.733	0.872	0.833	0.742	10.6
GcGAN [56]	Resnet-6	0.837	0.556	0.501	0.655	0.737	0.89	0.795	0.724	13.4
AFM [6]	UNET	0.495	0.432	0.307	0.457	0.498	0.721	0.684	0.579	44.0
LCNN [5]	Hourglass network	0.541	0.464	0.315	0.498	0.519	0.833	0.717	0.627	10.9
HAWP [3]	Hourglass network	0.581	0.421	0.315	0.485	0.532	0.862	0.704	0.633	11.6
PLGAN (ours)	Resnet-6	**0.863**	0.577	**0.533**	**0.687**	**0.769**	0.897	**0.849**	**0.787**	14.9

D. Comparison With Existing Methods on TTPLA Dataset

We compare the performance of PLGAN on TTPLA with a number of existing methods that can be grouped into three different categories. (i) Semantic image segmentation models: LinkNet [60], UNet++ [61], FPN [59], DeepLabv3+ [9], UNET [39], MaNet [62], AIFN [23] and Focal-UNet [29] as reported in Table I; (ii) GAN-based architectures: Pix2pix [44] and GcGAN [56] based on backbone 6 residual blocks (ResNet-6). GANs are evaluated based on the semantic images generated by assigning one to the pixels belonging to PLs and zero otherwise; (iii) Line segment detectors: AFM [6], LCNN [5], and HAWP [3]. AFM uses UNET as the backbone while the other two rely on stacked Hourglass network [72] as the backbone. Since the line segment detectors require a different type of annotation for their ground truth depending on the start and end points for each line, which is not compatible with our setting, we extend it to our problem to compare with them by preparing line segment annotation of PL on all the images in TTPLA, by following the general annotation pipeline in [6] on the original polygonal PL annotations.

Table I shows the quantitative results of the proposed PLGAN and all the above comparison methods on the test set of the TTPLA dataset. Figure 3 shows the segmentation results of sample images from both the proposed PLGAN and the comparison methods.

1) Comparison With Deep Semantic Segmentation Models: It is shown in Table I that PLGAN outperforms most of the baselines. Compared with PLGAN, we found that those baseline models produce more false positives in PL segmentation. For instance, UNET and FPN (columns 3 and 4 in Figure 3) misclassify many non-PL structures, such as sidewalks and lanes, as part of PLs. This observation can be interpreted from two aspects. First, most of these models are built upon the encoder-decoder structures, while the decoders fail to appropriately augment the complex background information when making pixel-wise predictions from the low-resolution feature maps generated by the encoder [42]. Second, the networks are trained based on the Softmax cross-entropy loss and ignore the interconnections between pixels as discussed in context [26], [33]. Therefore, it is hard to preserve global consistency [53]. Even though Focal-UNET [29] uses Focal loss function instead of BCE loss function for addressing the class imbalance in PL segmentation, it still suffers from the same limitation by not capturing the relation between pixels. We also notice that, although UNET++ and MaNet using ResNet-34 outperform PLGAN in recall and correctness, respectively, it is at the cost of many more parameters than PLGAN.

2) Comparison With GANs: As shown in column 5 of Figure 3, using pix2pix GAN to directly generate the semantic segmentation images reduces the performance by missing many PL pixels and generating false positives, resulting in many gaps along the segmented PLs. This is also reflected in the quantitative results shown in Table I. As discussed in the Related Work Section, this is the inherent limitation when generating/discriminating the semantic images directly: the discriminator pushes the generator to produce semantic images with sharp zeroes/ones and leaves a permanent possibility for the discriminator to examine the small, but always existing, value gap between the distributions of true labels and the predictions [53], which may hurt the performance of adversarial training. As shown in Table I, instead of directly using GAN to generate semantic images, the proposed PLGAN embeds features from GAN to a semantic segmentation network and can achieve much higher quality in PL segmentation.

3) Comparison With Line Segment Detector: As shown in Figure 3 (column 6), most of the line detectors can capture many PLs with very clean segmentation. This is totally reasonable since PLs are very-thin line structures, and line detectors fully take advantage of this geometry prior to ensuring the global consistency in PL segmentation. However, in using deep neural networks to boost the capability of line segment

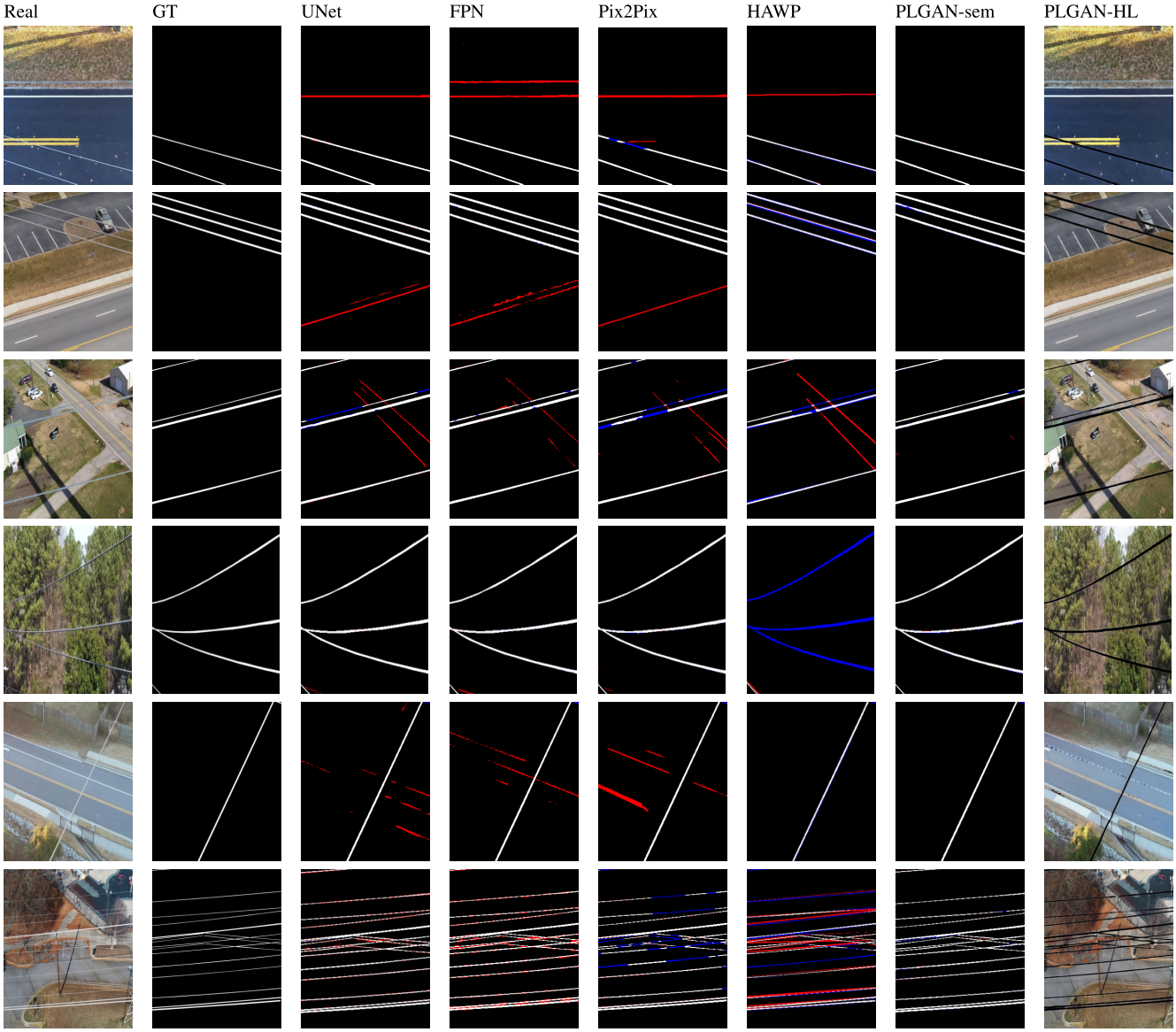

Fig. 3. Sample PL segmentation results produced by the proposed PLGAN and comparison methods on TTPLA dataset. The blue and red colors indicate the missing and false predictions, respectively. Appearing both colors for the same line means that this line has slight curvature, which can not detected correctly. Two pixels relaxation are used for all models to make the visualization more clear.

detection, most line detectors conduct spatial-region partitioning for network computation and feature representation. This inherently reduces the spatial resolution of features and may cause dislocation between the segmented PLs and their corresponding GTs. As a result, a group of lines can be missing in Figure 3. In addition, the line segment detectors cannot handle the curved power lines, as shown in the image in column 6 and row 4. Therefore, while most line segment detectors produce quite clean PL segmentation in some cases, its quality is still much lower than our PLGAN, as shown in Table I.

4) Comparison Considering Parameter Scale: It is important to highlight the observation that our model employs only half of the parameters used in the second-best models on the TTPLA dataset. In addition, when comparing our model (14.9M parameters) with models with similar scales (10.6M-18.5M parameters), our model outperforms all these models under every evaluation metric.

E. Comparison on Massachusetts Roads Dataset

Due to the lack of public PL datasets, we evaluate PLGAN on Massachusetts roads dataset for road extraction, which has the same nature as thin objects. We first follow the experiment setting in [65] and evaluate PLGAN using precision, recall, IoU, and F_1 score. We compare the performance of PLGAN with Rec-Middle [73], Rec-Last [74], ICNet [75], Rec-Simple [66], and DRU [65]. The results are reported in Table II. Then, we follow the experiment setup in [66] to evaluate the completeness, correctness, and quality of PLGAN. We compare our performance with Reg-AC [76], MNIH [11], and Rec-Simple [66]. The results are reported in Table III. In addition, we provide the results for Deeplab V3++, LinkNet, MaNet, and Unet++ using Resnet-34 as the backbone in both Tables II and III.

It can be found from both tables that PLGAN outperforms the state-of-the-art methods under most evaluation metrics. Our PLGAN achieves the highest precision, IoU, and F_1 as

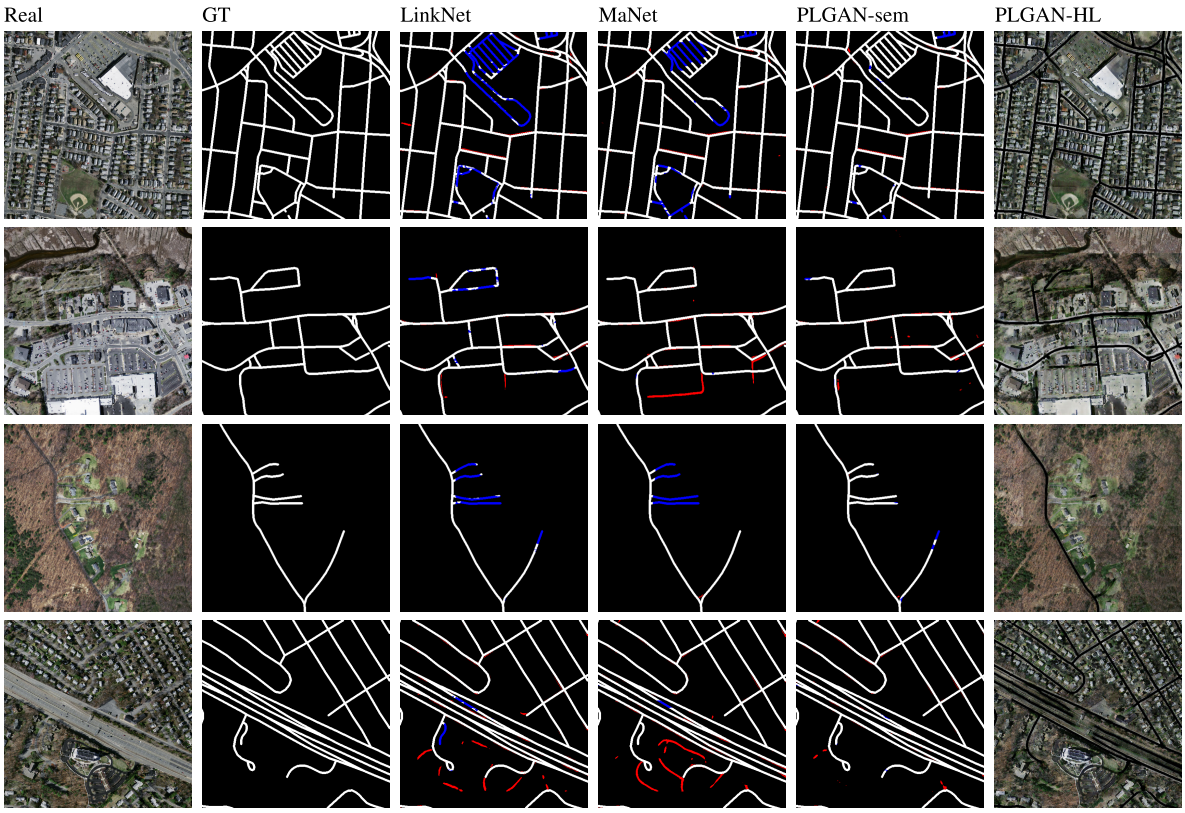

Fig. 4. Road extraction by our proposed PLGAN on Massachusetts roads dataset. The blue and red colors indicate the missing and false predictions, respectively. Two pixels relaxation are used for all models to make the visualization more clear.

TABLE II

COMPARISON ON MASSACHUSETTS ROADS DATASET BY PRECISION, RECALL, IOU, AND F_1 SCORE. BOLD REPRESENTS THE HIGHEST RESULTS, AND UNDERLINE REPRESENTS THE SECOND-BEST

Models	Precision	Recall	IoU	F_1
Rec-Middle [73]	0.518	0.767	0.494	0.574
Rec-Last [74]	0.551	0.786	0.526	0.648
ICNet [75]	0.500	0.626	0.476	0.656
Rec-Simple [66]	0.559	<u>0.802</u>	0.534	0.659
Linknet [60]	0.785	0.661	0.523	0.676
Deeplab V3+ [9]	0.773	0.667	0.525	0.678
MaNet [62]	0.789	0.681	0.539	0.689
DRU [65]	0.583	**0.865**	<u>0.560</u>	0.691
UNet++ [61]	<u>0.807</u>	0.655	0.540	<u>0.694</u>
PLGAN (Ours)	**0.813**	0.691	**0.571**	**0.721**

TABLE III

COMPARISON ON MASSACHUSETTS ROADS DATASET BY COMPLETENESS, CORRECTNESS, AND QUALITY. BOLD REPRESENTS THE HIGHEST RESULTS, AND UNDERLINE REPRESENTS THE SECOND-BEST

Models	Corr.	Comp.	Quality
Reg-AC [76]	0.254	0.348	0.172
MNIH [11]	0.531	0.752	0.452
Rec-Simple [66]	0.774	0.806	0.652
Linknet [60]	0.919	0.820	0.757
Deeplab V3+ [9]	0.914	0.822	0.756
MaNet [62]	0.922	<u>0.828</u>	<u>0.766</u>
UNet++ [61]	**0.943**	0.804	0.763
PLGAN (Ours)	<u>0.937</u>	**0.833**	**0.788**

reported in Table II, and the best completeness and quality in Table III. It is worth mentioning that UNet++ model in Table II and MaNet in Table III achieve the second-best F_1 and quality, respectively. This is mainly because UNet++ and MaNet use a significantly larger number of parameters (26.1M and 31.8M parameters, respectively) than PLGAN (14.9M parameters). Some segmentation testing samples are shown in Figure 4.

F. Ablation Study

We conducted an ablation study on the TTPLA dataset to evaluate the performance of different variants of our proposed PLGAN and to demonstrate the usefulness of its various components. For the first two variants, the PL-aware generator directly generated semantic segmentation images instead of PL-highlighted images since the semantic decoder was not included. The results of the first variant, including a PL-aware generator (G) with an adversarial loss function, are reported in the first row of Table IV. As a second variant, we applied the geometry loss function and reported the results in the second row of Table IV. In the third variant, we used the semantic decoder (S) to produce semantic images and used the PL-aware generator to generate PL-highlighted images (row 3 in Table IV). Next, we added the geometry loss function (geo) and the hough transform loss function ($\mathcal{HT}$) separately, and the results are presented in rows 4 and 5, respectively. Finally, we applied the geometry loss function on top of the previous variant in row 6 of Table IV. Through

TABLE IV
QUANTITATIVE ABLATION STUDY OF PLGAN VARIANTS

G	S	$\mathcal{HT}$	geo	Precision	Recall	IoU	F_1	F_β
✓				0.822	0.570	0.511	0.663	0.733
✓			✓	0.837	0.556	0.501	0.655	0.737
✓	✓			0.861	0.565	0.520	0.677	0.762
✓	✓		✓	0.857	0.569	0.523	0.679	0.762
✓	✓	✓		0.865	0.560	0.524	0.679	0.768
✓	✓	✓	✓	0.864	0.577	0.533	0.687	0.770

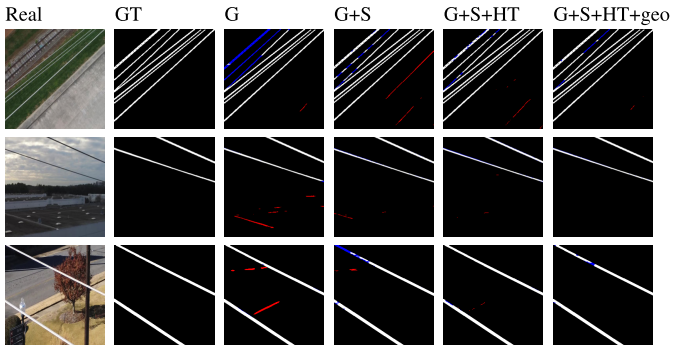

Fig. 5. Ablation study for different variant of PLGAN. The blue and red colors indicate the missing and false predication, respectively.

our ablation study, we aimed to highlight the effectiveness of each component in enhancing the performance of PLGAN. As shown in Table IV and Figure 5, we notice that applying the PL-highlighted images helps the generator to build the embedding vector as an input to the semantic decoder. Therefore, the performance across all metrics is improved in row 3. Additionally, we observe that $\mathcal{L}geo$ slightly enhances the recall (row 4) while $\mathcal{L}ht$ improves the precision (row 5). Finally, all the modules contribute to getting a higher F-score and IoU of PLGAN (row 6). Based on these observations, we conclude that our contributions are complementary, and the experimental results validate the importance of building end-to-end trainable models.

V. CONCLUSION

This paper introduces a novel GAN framework, PLGAN, specifically designed for power line segmentation in aerial images. PLGAN leverages adversarial training and effectively captures context, geometry, and appearance information for accurate prediction. In PLGAN, the generated PL-highlighted images are utilized by the discriminator, which compels PLGAN to emphasize power line regions within the images. By learning a joint representation in a shared latent space derived from the PL-highlighted image and the semantic image, PLGAN can generate more precise semantic images compared to state-of-the-art methods, as demonstrated through comprehensive experiments. As we aspire to improve our model in future work, it is worth mentioning that there are only a few small datasets on thin objects available in public, which may not be sufficient to fully train the PLGAN. To address this issue, weakly supervised learning, semi-supervised learning, or unsupervised learning techniques are promising. For instance, the work in [77] considers a region-to-region graph to capture spatial dependencies and local context. The method in [78] integrates recurrent layers to effectively capture temporal information and incorporates attention mechanisms that allow the model to focus on relevant regions. We will investigate these methods in our future work to enhance PLGAN's performance with limited data. Another direction for future research involves extending the applicability of PLGAN to diverse applications, such as video object detection [79], and salient object detection [80], while also exploring its potential to reduce the dependency on manually annotated pixel-level saliency masks through the use of limited pixel-level labeled data [48].

REFERENCES

[1] B. Bhanu, S. Das, B. Roberts, and D. Duncan, "A system for obstacle detection during rotorcraft low altitude flight," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 32, no. 3, pp. 875–897, Jul. 1996.
[2] K. Huang, Y. Wang, Z. Zhou, T. Ding, S. Gao, and Y. Ma, "Learning to parse wireframes in images of man-made environments," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 626–635.
[3] N. Xue et al., "Holistically-attracted wireframe parsing," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 2785–2794.
[4] Z. Zhang et al., "PPGNet: Learning point-pair graph for line segment detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 7098–7107.
[5] Y. Zhou, H. Qi, and Y. Ma, "End-to-end wireframe parsing," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 962–971.
[6] N. Xue, S. Bai, F. Wang, G.-S. Xia, T. Wu, and L. Zhang, "Learning attraction field representation for robust line segment detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1595–1603.
[7] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3431–3440.
[8] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking Atrous convolution for semantic image segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2017, pp. 1–6.
[9] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder–decoder with Atrous separable convolution for semantic image segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 801–818.
[10] R. Abdelfattah, X. Wang, and S. Wang, "TTPLA: An aerial-image dataset for detection and segmentation of transmission towers and power lines," in *Proc. Asian Conf. Comput. Vis. (ACCV)*, 2020, pp. 1–17.
[11] V. Mnih, *Machine Learning for Aerial Image Labeling*. Toronto, ON, Canada: Univ. Toronto, 2013.
[12] W. Xia, Y. Zhang, Y. Yang, J.-H. Xue, B. Zhou, and M.-H. Yang, "GAN inversion: A survey," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 3, pp. 3121–3138, Mar. 2023.
[13] V. N. Nguyen, R. Jenssen, and D. Roverso, "LS-Net: Fast single-shot line-segment detector," *Mach. Vis. Appl.*, vol. 32, no. 1, pp. 1–16, Jan. 2021.
[14] 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 Sens.*, vol. 11, no. 11, p. 1342, Jun. 2019.
[15] O. Russakovsky et al., "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.
[16] 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 Trans. Aerosp. Electron. Syst.*, vol. 45, no. 3, pp. 937–949, Jul. 2009.
[17] G. Yan, C. Li, G. Zhou, W. Zhang, and X. Li, "Automatic extraction of power lines from aerial images," *IEEE Geosci. Remote Sens. Lett.*, vol. 4, no. 3, pp. 387–391, Jul. 2007.
[18] I. Golightly and D. Jones, "Visual control of an unmanned aerial vehicle for power line inspection," in *Proc. 12th Int. Conf. Adv. Robot.*, 2005, pp. 288–295.
[19] Z. Li, Y. Liu, R. Hayward, J. Zhang, and J. Cai, "Knowledge-based power line detection for UAV surveillance and inspection systems," in *Proc. 23rd Int. Conf. Image Vis. Comput. New Zealand*, Nov. 2008, pp. 1–6.

[20] T. Santos et al., "PLineD: Vision-based power lines detection for unmanned aerial vehicles," in *Proc. IEEE Int. Conf. Auto. Robot Syst. Competition (ICARSC)*, Apr. 2017, pp. 253–259.

[21] Ö. E. Yetgin and Ö. N. Gerek, "A comparison of corner and saliency detection methods for power line detection," in *Proc. Int. Artif. Intell. Data Process. Symp. (IDAP)*, Sep. 2017, pp. 1–5.

[22] Ö. E. Yetgin, B. Benligiray, and Ö. N. Gerek, "Power line recognition from aerial images with deep learning," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 55, no. 5, pp. 2241–2252, Oct. 2019.

[23] Y. Li, Z. Xiao, X. Zhen, and X. Cao, "Attentional information fusion networks for cross-scene power line detection," *IEEE Geosci. Remote Sens. Lett.*, vol. 16, no. 10, pp. 1635–1639, Oct. 2019.

[24] R. Madaan, D. Maturana, and S. Scherer, "Wire detection using synthetic data and dilated convolutional networks for unmanned aerial vehicles," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2017, pp. 3487–3494.

[25] S. J. Lee, J. P. Yun, H. Choi, W. Kwon, G. Koo, and S. W. Kim, "Weakly supervised learning with convolutional neural networks for power line localization," in *Proc. IEEE Symp. Ser. Comput. Intell. (SSCI)*, Nov. 2017, pp. 1–8.

[26] S. Zhao, Y. Wang, Z. Yang, and D. Cai, "Region mutual information loss for semantic segmentation," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 11117–11127.

[27] C. Pan, X. Cao, and D. Wu, "Power line detection via background noise removal," in *Proc. IEEE Global Conf. Signal Inf. Process. (GlobalSIP)*, Dec. 2016, pp. 871–875.

[28] J. Gubbi, A. Varghese, and P. Balamuralidhar, "A new deep learning architecture for detection of long linear infrastructure," in *Proc. 15th IAPR Int. Conf. Mach. Vis. Appl. (MVA)*, May 2017, pp. 207–210.

[29] R. Jaffari, M. A. Hashmani, and C. C. Reyes-Aldasoro, "A novel focal phi loss for power line segmentation with auxiliary classifier U-Net," *Sensors*, vol. 21, no. 8, p. 2803, Apr. 2021.

[30] B. W. Matthews, "Comparison of the predicted and observed secondary structure of T4 phage lysozyme," *Biochimica Biophysica Acta (BBA)-Protein Struct.*, vol. 405, no. 2, pp. 442–451, Oct. 1975.

[31] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Semantic image segmentation with deep convolutional nets and fully connected CRFs," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2015, pp. 1–12.

[32] S. Zheng et al., "Conditional random fields as recurrent neural networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1529–1537.

[33] T.-W. Ke, J.-J. Hwang, Z. Liu, and S. X. Yu, "Adaptive affinity fields for semantic segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 587–602.

[34] G. Bertasius, L. Torresani, S. X. Yu, and J. Shi, "Convolutional random walk networks for semantic image segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6137–6145.

[35] Z. Huang, X. Wang, L. Huang, C. Huang, Y. Wei, and W. Liu, "CCNet: Criss-cross attention for semantic segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 603–612.

[36] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2016.

[37] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic image segmentation with deep convolutional nets, Atrous convolution, and fully connected CRFs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2018.

[38] H. Ding, X. Jiang, B. Shuai, A. Q. Liu, and G. Wang, "Context contrasted feature and gated multi-scale aggregation for scene segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2393–2402.

[39] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, 2015, pp. 234–241.

[40] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A deep convolutional encoder–decoder architecture for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017.

[41] G. Lin, A. Milan, C. Shen, and I. Reid, "RefineNet: Multi-path refinement networks for high-resolution semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5168–5177.

[42] Z. Tian, T. He, C. Shen, and Y. Yan, "Decoders matter for semantic segmentation: Data-dependent decoding enables flexible feature aggregation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3121–3130.

[43] I. J. Goodfellow et al., "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 27, 2014, pp. 2672–2680.

[44] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5967–5976.

[45] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2242–2251.

[46] W.-S. Lai, J.-B. Huang, N. Ahuja, and M.-H. Yang, "Deep Laplacian pyramid networks for fast and accurate super-resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5835–5843.

[47] D. Pathak, P. Krähenbühl, J. Donahue, T. Darrell, and A. A. Efros, "Context encoders: Feature learning by inpainting," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2536–2544.

[48] D. Zhang, H. Tian, and J. Han, "Few-cost salient object detection with adversarial-paced learning," in *Proc. NIPS*, vol. 33, 2020, pp. 12236–12247.

[49] J.-Y. Zhu, P. Krähenbühl, E. Shechtman, and A. A. Efros, "Generative visual manipulation on the natural image manifold," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2016, pp. 597–613.

[50] P. Luc, C. Couprie, S. Chintala, and J. Verbeek, "Semantic segmentation using adversarial networks," in *Proc. NIPS Workshop Adversarial Training*, 2016.

[51] M. Majurski et al., "Cell image segmentation using generative adversarial networks, transfer learning, and augmentations," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2019, pp. 1114–1122.

[52] W.-C. Hung, Y.-H. Tsai, Y.-T. Liou, Y.-Y. Lin, and M.-H. Yang, "Adversarial learning for semi-supervised semantic segmentation," in *Proc. Brit. Mach. Vis. Conf.*, 2018.

[53] L. Samson, N. van Noord, O. Booij, M. Hofmann, E. Gavves, and M. Ghafoorian, "I bet you are wrong: Gambling adversarial networks for structured semantic segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshop (ICCVW)*, Oct. 2019, pp. 951–960.

[54] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[55] S. Lee, G. Hwan An, and S.-J. Kang, "Deep recursive HDRI: Inverse tone mapping using generative adversarial networks," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 596–611.

[56] H. Fu, M. Gong, C. Wang, K. Batmanghelich, K. Zhang, and D. Tao, "Geometry-consistent generative adversarial networks for one-sided unsupervised domain mapping," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 2422–2431.

[57] X. Mao, Q. Li, H. Xie, R. Y. K. Lau, Z. Wang, and S. P. Smolley, "Least squares generative adversarial networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2813–2821.

[58] Y. Lin, S.-L. Pintea, and J. van Gemert, "Semi-supervised lane detection with deep Hough transform," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2021, pp. 1514–1518.

[59] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 936–944.

[60] A. Chaurasia and E. Culurciello, "LinkNet: Exploiting encoder representations for efficient semantic segmentation," in *Proc. IEEE Vis. Commun. Image Process. (VCIP)*, Dec. 2017, pp. 1–4.

[61] Z. Zhou, M. M. R. Siddiquee, N. Tajbakhsh, and J. Liang, "UNet++: A nested U-Net architecture for medical image segmentation," in *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*. Cham, Switzerland: Springer, 2018, pp. 3–11.

[62] T. Fan, G. Wang, Y. Li, and H. Wang, "MA-Net: A multi-scale attention network for liver and tumor segmentation," *IEEE Access*, vol. 8, pp. 179656–179665, 2020.

[63] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman, "LabelMe: A database and web-based tool for image annotation," *Int. J. Comput. Vis.*, vol. 77, nos. 1–3, pp. 157–173, May 2008.

[64] V. N. Nguyen, R. Jenssen, and D. Roverso, "Intelligent monitoring and inspection of power line components powered by UAVs and deep learning," *IEEE Power Energy Technol. Syst. J.*, vol. 6, no. 1, pp. 11–21, Mar. 2019.

[65] W. Wang, K. Yu, J. Hugonot, P. Fua, and M. Salzmann, "Recurrent U-Net for resource-constrained segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 2142–2151.

[66] A. Mosinska, P. Marquez-Neila, M. Kozinski, and P. Fua, "Beyond the pixel-wise loss for topology-aware delineation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3136–3145.

[67] H. Mei et al., "Don't hit me! Glass detection in real-world scenes," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 3684–3693.

[68] X. Qin, Z. Zhang, C. Huang, C. Gao, M. Dehghan, and M. Jagersand, "BASNet: Boundary-aware salient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 7471–7481.

[69] C. Wiedemann, C. Heipke, H. Mayer, and O. Jamet, "Empirical evaluation of automatically extracted road axes," in *Proc. Empirical Eval. Techn. Comput. Vis.*, vol. 12, Jun. 1998, pp. 172–187.

[70] Q. Zou, Z. Zhang, Q. Li, X. Qi, Q. Wang, and S. Wang, "DeepCrack: Learning hierarchical convolutional features for crack detection," *IEEE Trans. Image Process.*, vol. 28, no. 3, pp. 1498–1512, Mar. 2019.

[71] A. Sironi, V. Lepetit, and P. Fua, "Multiscale centerline detection by learning a scale-space distance transform," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 2697–2704.

[72] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2016, pp. 483–499.

[73] R. P. Poudel, P. Lamata, and G. Montana, "Recurrent fully convolutional neural networks for multi-slice MRI cardiac segmentation," in *Reconstruction, Segmentation, and Analysis of Medical Images*. Springer, 2016, pp. 83–94.

[74] S. Valipour, M. Siam, M. Jagersand, and N. Ray, "Recurrent fully convolutional networks for video segmentation," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2017, pp. 29–36.

[75] H. Zhao, X. Qi, X. Shen, J. Shi, and J. Jia, "ICNet for real-time semantic segmentation on high-resolution images," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 405–420.

[76] A. Sironi, E. Türetken, V. Lepetit, and P. Fua, "Multiscale centerline detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 7, pp. 1327–1341, Jul. 2016.

[77] C. Yan, Q. Zheng, X. Chang, M. Luo, C.-H. Yeh, and A. G. Hauptman, "Semantics-preserving graph propagation for zero-shot object detection," *IEEE Trans. Image Process.*, vol. 29, pp. 8163–8176, 2020.

[78] K. Chen, L. Yao, D. Zhang, X. Wang, X. Chang, and F. Nie, "A semisupervised recurrent convolutional attention model for human activity recognition," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 5, pp. 1747–1756, May 2020.

[79] W. Zhao, J. Zhang, L. Li, N. Barnes, N. Liu, and J. Han, "Weakly supervised video salient object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 16821–16830.

[80] N. Liu and J. Han, "DHSNet: Deep hierarchical saliency network for salient object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 678–686.