IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received September 16, 2018, accepted October 4, 2018, date of publication October 8, 2018, date of current version October 31, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2874544

Scribble-Supervised Segmentation of Aerial Building Footprints Using Adversarial Learning

WEIMIN WU[1], (Senior Member, IEEE), HUAN QI[2], ZHENRUI RONG[1], LIANG LIU[3], AND HONGYE SU[1], (Senior Member, IEEE)

[1]State Key Laboratory of Industrial Control Technology, Institute of Cyber-Systems and Control, Zhejiang University, Hangzhou 310027, China
[2]Department of Engineering Science, University of Oxford, Oxford OX3 7DQ, U.K.
[3]Daniel J. Epstein Department of Industrial and Systems Engineering, University of Southern California, Los Angeles, CA 90089, USA

Corresponding author: Weimin Wu (wmwu@iipc.zju.edu.cn)

This work was supported by the National Natural Science Foundation of China under Grants 61773343 and 61621002.

ABSTRACT Aerial image segmentation usually requires a large amount of pixel-level masks in order to achieve quality performance. Obtaining these annotations can be both costly and time-consuming, limiting the amount of data available for training. In this paper, we present an approach for learning to segment aerial building footprints in the absence of fully annotated label masks. Instead, we exploit cheap and efficient scribble annotations to supervise deep convolutional neural networks for segmentation. Our proposed model is based on an adversarial architecture that jointly trains two networks to produce building footprint segmentations that resemble synthetic label masks. We present competitive segmentation results on the Massachusetts Buildings data set by using only scribble supervision signals. Further experiments show that our method effectively alleviates building instance separation issue and displays strong robustness towards different scribble instance levels. We believe our cost-effective approach has the potential to be adapted for other aerial image interpretation tasks.

INDEX TERMS Aerial image, generative adversarial network, image segmentation, weak supervision.

I. INTRODUCTION

The process of examining aerial imagery with the purpose of recognizing high-level semantics such as objects and scenes is referred to as aerial image interpretation [1]. In recent years, technological advances have motivated the development of geographic analysis. Numerous aerial images are produced, collected and stored everyday, bringing new challenges for the development of techniques to meet the requirements of geographical knowledge understanding and discovery. To this end, exploiting big data analysis techniques to automate aerial image interpretation has become an increasingly promising approach, as evidenced in [2]–[4].

One of the most fundamental aerial image interpretation tasks is to assign a semantic label (e.g. building, tree, car) to each pixel of an aerial image, i.e. converting the raw input into a semantically meaningful raster map before further processing such as vectorization [5]. This corresponds to *semantic image segmentation* in computer vision, which can be formulated as a classification problem by mapping the pixel set into a pre-defined label set. So far, the most effective technique for semantic segmentation is supervised machine learning, which usually requires a large amount of fully annotated training samples. Within the training set, each sample image needs to be paired with a fully annotated 'ground-truth' label mask of the same size.

Supervised image segmentation with ground-truth masks (*strong* supervision) have achieved great success over the past few years. Methods such as deep learning generally requires a large training set with fully annotated label masks. Although powerful annotation tools have been developed over the years to speed up the process [6], [7], it may still takes minutes for an experienced annotator to label one image [8]. Take aerial building footprint segmentation as an example. It is not uncommon for a 500×500 px image to contain more than 100 building instances, making the mask annotation process even more demanding. Therefore, it can be quite costly and time-consuming to obtain fully annotated label masks for aerial images, limiting the amount of data available for training. It is desirable for a learning-based model to work with *weak* supervision of certain form, which is expected to be much cheaper to attain than its strong counterpart.

Driven by the motivation of balancing annotation efficiency and model performance for automated aerial image interpretation, we propose to perform building footprint

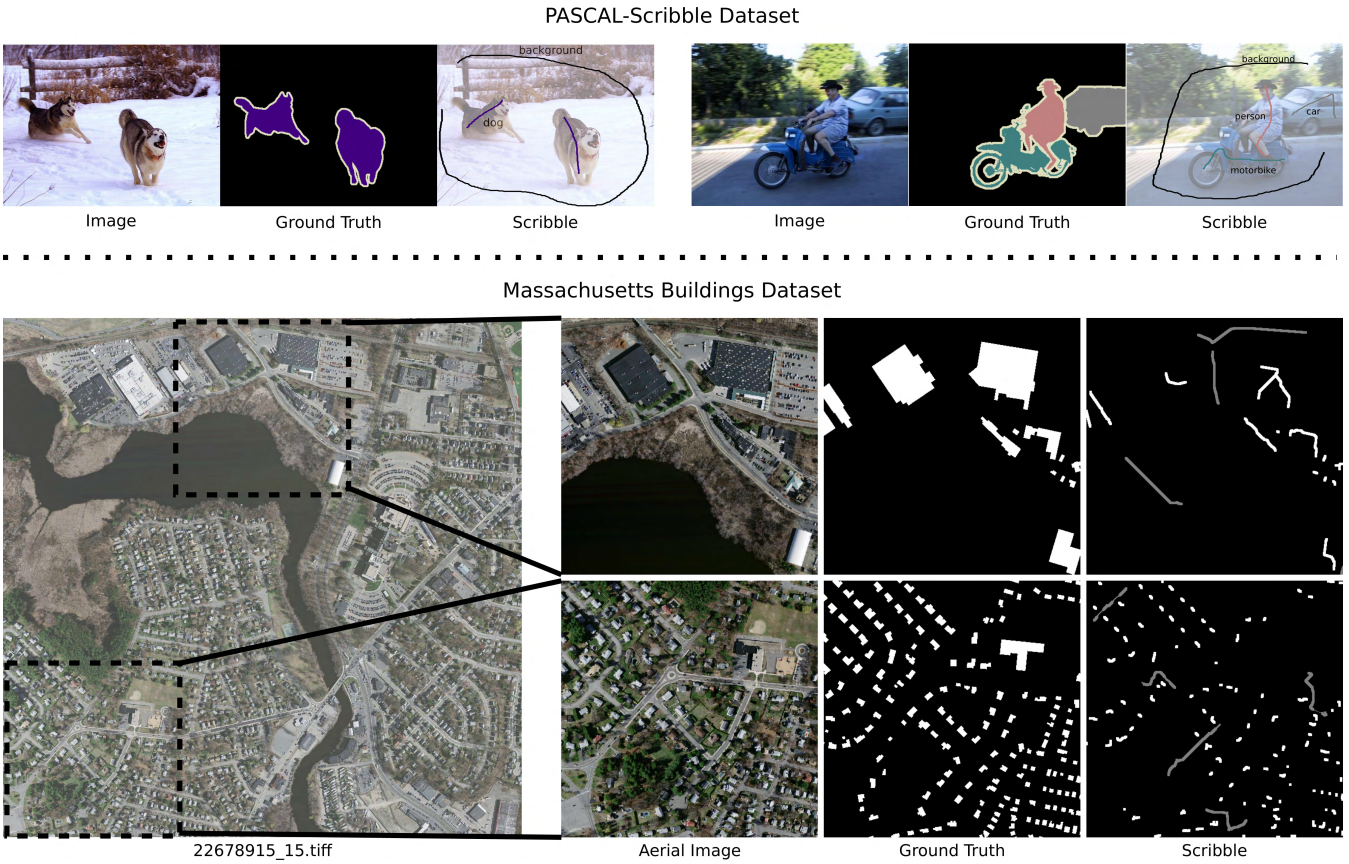


FIGURE 1. Samples from PASCAL-Scribble Dataset and Massachusetts Buildings Dataset. Each sample contains a ground truth mask and a scribble mask. Note that images from PASCAL-Scribble Dataset generally contain fewer object instances and have relatively consistent background in spatial/color feature space, while images from the Massachusetts Buildings Dataset contain a number of building instances of various sizes and shapes. White and grey scribble denote building and background, respectively.

segmentation using a type of weak supervision: scribble supervision. As shown in Figure 1, it is much more efficient to annotate images with simple scribbles than with label masks. Specifically, we develop an algorithm that exploits *only scribble annotations* to train a deep convolutional neural network for building footprint segmentation. Without object boundary outlines, it is supposed to be a very challenging task. For the rest of this paper, we address models supervised by fully annotated label masks as 'mask-supervised' models, as opposed to our 'scribble-supervised' models.

We approach scribble-supervised building footprint segmentation by dividing it into two sub-problems. For the first one, we train a fully convolutional network (FCN) to output building label predictions. This is similar to the mask-supervised model, except that we only back-propagate cross-entropy losses from *pixels that are covered by scribbles* since they are the only correct labels we have obtained. This is addressed as a partial cross-entropy (pCE) baseline. We observe this simple but practically effective method causes building instance separation issue, i.e. the model cannot separate building instances from each other, leaving them to stick together and form building blobs, as shown in Figure 2. For the second sub-problem, we introduce an adversarial learning architecture by considering the pCE

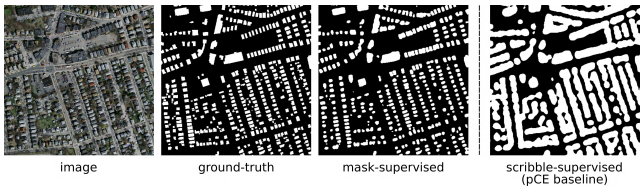


FIGURE 2. A test sample from the Massachusetts Buildings Dataset. The third and fourth column display segmentations from a mask-supervised model and a pCE baseline model. Observe that predictions of pCE baseline can achieve a reasonably high Interaction-over-Union (IoU) score but has severe instance separation issue with multiple large building blobs.

baseline as a *generator*, which keeps generating building label predictions. Following [9], we design a second convolutional neural network, the *discriminator*, whose mission is to distinguish 'fake' generator outputs from 'real' building masks. During adversarial learning, the generator and discriminator keep playing a min-max game, which eventually cause the discriminator to fail its mission, i.e. the generator has successfully produced outputs that resemble 'real' building masks to some extent. This architecture enables us to incorporate shape priors to regularize behaviors of the generator. Note that since we do not have any 'real' building

mask after all, we propose a simple yet effective pipeline to synthesize *obb-scribble* masks from scribble annotations by fitting oriented bounding boxes around scribbles to mimic how 'real' building masks should look like. Motivated by scribble-supervised methods as well as the generative adversarial network (GAN) [9], we address the proposed model as *ScrGAN*.

We use the Massachusetts Buildings Dataset [1] to evaluate our method. To further investigate ScrGAN's sensitivity towards scribble instance numbers, we propose an automatic scribble generation method that allows us to flexibly control how many scribbles to draw within each aerial image. Experimental results show that ScrGAN outperforms all the baseline models including the current state-of-the-art method using various evaluation metrics. It shows reasonable degradation compared to its mask-supervised counterpart. Moreover, ScrGAN alleviates building instance separation issue and appears robust towards various scribble instance levels. These results suggest that scribble-supervised methods have many desirable aspects as well as the potential to be applicable in other aerial image interpretation tasks.

The rest of this paper is arranged as follows: In Section II, related work is reviewed with focus on deep-learning-based strongly and weakly supervised image segmentation. In Section III, the proposed scribble-based aerial image segmentation method is introduced. The corresponding experimental results and discussion are presented in Section IV, followed by Section V, which concludes the paper.

II. RELATED WORK

A. FULLY SUPERVISED AERIAL IMAGE SEGMENTATION

Since manual examination can be both expensive and time-consuming, attempts have been made to develop semi- or fully-automatic systems for aerial image interpretation since 1970s [10]. The recent rapid growth of heterogeneous data, including the availability of high-resolution aerial imagery in the areas of urban planning [11], path planning [12], crop management [13] etc., underlines the need to develop fully automatic tools with satisfying levels of accuracy and efficiency. To this end, supervised machine learning models have found numerous applications in aerial image interpretation [14], [15].

In machine learning and computer vision applications, the task of semantic segmentation is usually formulated as a pixel-wise classification problem, in which pre-defined semantic labels are manually assigned to all pixels within each image in the training set. Learning algorithms then perform parameter updating on given models according to customized objective functions. In order to achieve a good generalization ability, i.e. performing accurate pixel-wise classification on data outside the training set, regularization terms such as L2 norm are incorporated in the objective functions. Around a decade ago, successful semantic segmentation models relied on hand-crafted local features and flat statistical learners [16], [17]. Applications of these models in aerial image segmentation led to better performance especially in high-resolution imagery. For instance, Porway *et al.* [18] developed a boosting-based hierarchical model to parse aerial images with a pre-defined label category containing car, road, tree roof and parking lot. Kluckner *et al.* [19] developed covariance-based feature representations for aerial imagery and learnt through multi-class random forests and conditional random fields (CRF). Substantial performance improvements were made by enriching the representative ability of local features with high-order context information and structured predictions [20], [21].

Recent years have witnessed a series of revolutionary progresses in image segmentation due to the emergence of deep learning. By cascading non-linear mapping that transforms the representation at one level (starting from the raw input) into a higher and more abstract level, deep learning methods can thus learn complex features that are otherwise non-trivial to obtain in previous hand-crafted methods. One of the key aspects of deep learning is that features from all layers are learnt in a fully data-driven fashion [22]. Recently, a fully convolutional network (FCN) architecture was introduced for semantic segmentation [23]. In FCN setting, fully connected layers are replaced by the more efficient and spatially informative convolutional layers, and therefore avoid redundant computations in overlapping patches. Since then, FCN-based models have achieved the state-of-the-art performances in several large-scale segmentation challenges such as PASCAL VOC. Using up-sampling operations such as bi-linear interpolation and transposed convolution [23], [24], the resolution of outputs can be restored to the same as inputs. Other techniques such as skip connections [25], dilated convolutions [26], [27] and CRF post-processing [28] have been shown to marginally improve segmentation performance.

Automatic aerial image interpretation has benefited from the development of deep learning methods. Among early works, Mnih *et al.* proposed deep neural networks for road detection [29] and map annotation [15]. More recently, FCN models have been successfully used in aerial image segmentation [5], [30], [31]. Marmanis *et al.* ensembled two structurally identical FCNs for very high resolution (VHR) aerial image segmentation in an end-to-end way. They cast deep supervision at multiple scales to form a joint objective function with the purpose to adjust intermediate features for more efficient learning [32]. Bischke *et al.* proposed a multi-task learning approach to train a VGG-16 based encoder-decoder FCN. Besides a cross-entropy loss for standard segmentation task, they added a second output to regress the distance transformation of the segmentation mask in order to preserve roof boundaries in high-resolution aerial imagery [31]. All these methods focused on designing more effective network architectures given a fully annotated dataset. However, a more realistic issue, arising with the popularity of deep learning, is the increasing demand of annotated data. Semantic segmentation, among common computer vision tasks, is arguably the most *data hungry* one due to the fact that every pixel within an image needs annotating, which can be both expensive and

time-consuming. This makes huge amounts of fully annotated training data generally rare for image segmentation tasks. Recently, Kaiser *et al.* [5] proposed to extract segmentation masks directly from noisy crowd-sourced online maps and trained FCNs on them accordingly. The idea of generating training labels automatically inspired us to investigate the use of unconventional yet less expensive supervision signals for aerial image segmentation, which corresponds to the emerging field of weakly supervised learning.

B. WEAKLY SUPERVISED IMAGE SEGMENTATION

In a recent review [33], Zhou categorized weakly supervision signals into three types: incomplete supervision, where only a subset of training data are given strong labels; inexact supervision, where only coarse-grained labels are provided; and inaccurate supervision, where the given labels are not always ground-truth. Incomplete supervision usually relies on active learning or semi-supervised learning techniques, which is beyond the scope of this paper. Inexact supervision mainly deals with class-level labels. It is not suitable for complex high resolution aerial imagery, where pixel-level spatial information rather than class-level information is of the primary concern. Inaccurate supervision for image segmentation frees annotators from careful outline of object boundaries by sacrificing performance. Annotations for inaccurate supervision is usually much cheaper. Meanwhile it restores spatial information to some extent, which has intriguing potentials for image segmentation.

Training image segmentation models with (weak) inaccurate supervision has caught wide interest recently. Various forms of supervision signals were exploited, e.g. scribbles [8], [34], points [35], [36] and bounding boxes [37], [38]. In this work, we investigate training aerial image segmentation models using scribbles as supervision signals. Previously, scribbles were widely used in interactive image segmentation [39]–[41] and were recognized as being much more user-friendly than pixel-level manual segmentation. Driven by the popularity of weakly supervised learning, recent work considered directly using scribbles (instead of fully-annotated masks) to train image segmentation models. Lin *et al.* [8] developed an alternating training scheme: use superpixel-based graph cut to generate segmentation proposals from scribbles, train an FCN using these proposals, refine the unary terms with FCN predictions. By iterating this process, the FCN were gradually fed with more reliable proposals and thus propagated more accurate labels. They also introduced the PASCAL-Scribble Dataset, which is shown in Figure 1. However, inaccuracies of the generated segmentation proposals can propagate errors thus inevitably sabotage FCN's performance. Tang *et al.* [34] proposed a solution to this problem. Instead of alternating between FCN and graphical models, they trained a single FCN via a joint loss function with two terms: one is a partial cross entropy loss for scribbles only, the other is a relaxed normalized-cut regularizer that implicitly propagated true labels to unknown pixels during training.

Although the aforementioned learning algorithms brought the performance of scribble-supervised models closer to their mask-supervised counterparts on datasets such as PASCAL VOC, there is no guarantee that they would generalize well to aerial image segmentation. As shown in Figure 1, objects in PASCAL-Scribble Dataset tend to appear near the centre of the image with clear boundaries and reasonably large size. Stuff (water, sky, road, etc.) in the dataset tend to be consistent in both spatial and color domain. On the contrary, building instances in the Massachusetts Buildings Dataset, for instance, appear to be randomly distributed with varying sizes and shapes. Existence of trees and shadows may exert negative influence on image interpretation. Therefore, it is not clear whether the performance of weakly-supervised learning algorithms on aerial imagery would be consistent with those on PASCAL. So far, many efforts have been made to utilize weakly supervised learning methods on aerial image interpretation. Han *et al.* [42] proposed to use saliency detection and class-level annotations for object detection in optical remote sensing images. Zhou *et al.* [43] presented a deep learning approach via transfer learning and negative bootstrapping to detect targets in aerial imagery. To the best of our knowledge, no work has made any attempt to investigate the performance of scribble-based weak supervision for aerial image segmentation or propose any modification to make it well-suited for aerial imagery.

C. GENERATIVE ADVERSARIAL NETWORKS

The recent success of Generative Adversarial Networks (GANs) opens up intriguing possibilities for many research areas. In the original work, Goodfellow *et al.* [9] proposed an adversarial architecture to train deep generative models. A typical GAN consists of two sub-networks: a generator and a discriminator, where two networks compete with each other in a min-max game during training. The adopted *adversarial loss* regulates the behaviors of two sub-networks. It is expected that the generator would learn a mapping which makes its output indistinguishable from samples drawn from a target domain. GANs have achieved impressive progresses in image generation [44], unsupervised image-to-image translation [45] and representation learning [44], [46]. Recent work begins to apply adversarial learning as a new type of structured loss that enforces extra regularizations upon the existing model, which is otherwise non-trivial to define in closed-form objective functions. In this setting, the existing model is usually treated as a generator conditioned on a given input, while a discriminator is optimized to distinguish the model's output from ground truths. For instance, Luc *et al.* [47] trained an FCN (generator) to perform semantic segmentation and a discriminator to distinguish ground truth masks from generator's outputs. The motivation is to correct higher-order inconsistencies between ground truth masks and model predictions via the adversarial loss. Pan *et al.* [48] adopted a similar architecture for saliency prediction in order to produce saliency maps that resembled the ground truth. Hung *et al.* [49] exploited GANs

TABLE 1. Annotation efficiency test. Columns represents the form of inaccurate supervision signals. Note that test images are of size 500×500 px, cropped from raw aerial images. The unit is second per image.

	Scribble	Point	Bounding Box
Massachusetts Building [1]	55.5	49.8	108.5

to facilitate semi-supervised learning and achieved better performances on several semantic segmentation datasets. More recently, GANs have shown great potential in improving weakly supervised learning. Shen *et al.* [50] speeded up online weakly supervised object detectors by introducing an adversarial paradigm that trained class-level labels. Remez *et al.* [51] reformulated the box-supervised segmentation task using an adversarial cut-and-paste operation. It remains an open question how to exploit adversarial learning in under-constrained settings such as scribble-supervised image segmentation, where ground-truth masks are not available during training.

III. METHODS

A. SCRIBBLES AS SUPERVISION SIGNALS

As shown in Figure 1, a scribble is a set of pixels with a semantic label. Compared to mask annotation, which requires annotators to categorize all pixels into corresponding semantic labels, scribble annotation tends to be sparse, with the process being more intuitive and thus less time-consuming. Pixels that are not annotated with scribbles fall into an 'unknown' category and technically should not be punished or rewarded during learning. Without using any full mask annotation, our algorithm exploits only scribbles to supervise deep convolutional neural networks for aerial building footprint segmentation. As mentioned in II-B, inaccurate supervision signals may take various forms such as scribbles, points and bounding boxes. A point annotation [35], [36] can be viewed as a degenerate case of a scribble annotation, where annotators simply click at the centre of the object and store the point coordinate and class label. Unlike points and scribbles that lie within objects, a bounding box annotation aims to determinate the spatial bound of an object by excluding all irrelevant pixels outside the box. It is a stronger type of supervision signal. One obvious drawback of bounding box annotation, besides lower annotation efficiency, is that it is not suitable for annotating non-object regions (or 'stuff') such as sky, water and the generic 'background' in our case.

In Table 1, we present results of a preliminary annotation efficiency test among scribble, point and bounding box annotations on the Massachusetts Buildings Dataset. Following annotation criteria in [8], [36], and [51], we randomly sample 50 images and annotate *all* building instances with scribbles, points and bounding boxes respectively. Each image is cropped from the raw aerial maps to have width and height of 500 pixels. Details will be covered in Section IV-A. As shown in Table 1, it appears that scribble and point annotation share a similar level of annotation efficiency. We argue this results from the fact that objects such as buildings in aerial imagery usually have many instances and reasonably small sizes, degenerating scribbles towards points. Since a scribble annotation covers more pixels (i.e. creates more training samples) than a point annotation, this implies that scribble-supervised learning may be more promising than point-supervised learning in aerial image interpretation, as an alternative to mask-supervised learning. To evaluate scribble-based learning under different circumstances, we propose to generate scribbles automatically from ground-truth masks. Note that this is simply a flexible and time-saving alternative to drawing scribble manually since our method generates scribbles that visually resemble those from manual annotation. Moreover, our scribble generation method involves randomness, as opposed to human annotation which is inclined to subjectivity. This is by all means different from using ground-truth masks directly to train segmentation models, but rather a flexible use of ground-truth masks in a self-supervised fashion, i.e. all supervision signals come from the dataset itself and no extra human input is needed. We discuss and verify the effectiveness of our scribble generation method in Section IV by comparing against scribbles drawn by two human annotators under a specific annotation protocol.

Given a ground-truth mask with '1' representing building and '0' being background, our scribble generation method accounts for two tasks. For the first one, we isolate each object instance based on connectivity and then run a fast skeletonization algorithm [52] to reduce the instance to a 1-pixel-width representation without breaking its internal connectivity. Empirically, we find the resulting skeletons qualify as scribbles for small object instances with regular shapes such as buildings. We further augment the skeletons by incorporating neighbors that are within a distance of 4 pixels to thicken the skeleton. In Figure 3(a), we illustrate the simple pipeline for the first task, which is referred to as *foreground scribble generation*.

For the second task *background scribble generation*, we start by inverting the mask and perform skeletonization on it to obtain a connected skeleton of the background. The key part is to extract visually acceptable scribbles from the skeleton to resemble those drawn by human annotators, which tend to be smooth rather than zigzag. To achieve this, we propose an effective random walk (RW) searching algorithm to collect a set of spatially connected coordinates from the skeleton, following a cumulative direction sampling mechanism. As shown in Algorithm 1, the search starts by randomly sampling a coordinate c on the skeleton $\mathbf{S}$. Our random walker moves according to a compass vector $\mathbf{p}$, denoting the probabilistic distribution of choosing one out of eight based on 8-neighbor connectivity. Every direction in $\mathbf{p}$ is initialized to be 0.125. We assign an offset vector for each direction based on 2D image coordinate system. For example, north corresponds to $[0, -1]$ and southwest $[-1, 1]$. Our RW moves according to this compass vector. For each 'Move' operation, a random direction was drawn by following a categorical distribution on $\mathbf{p}$. If the resulting position is also on the

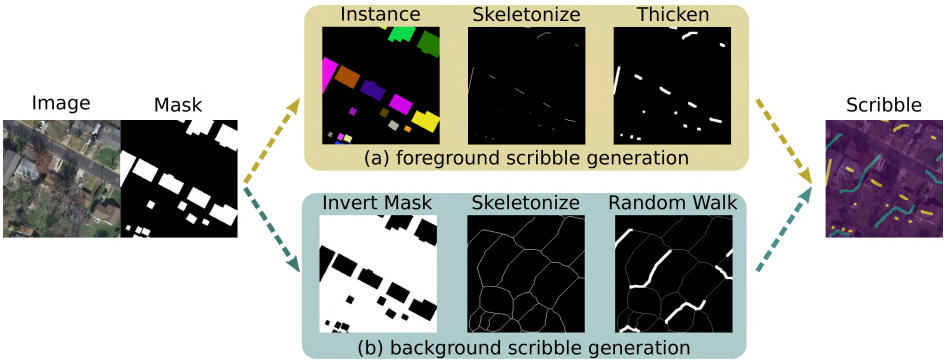


FIGURE 3. Automatic scribble generation pipeline.

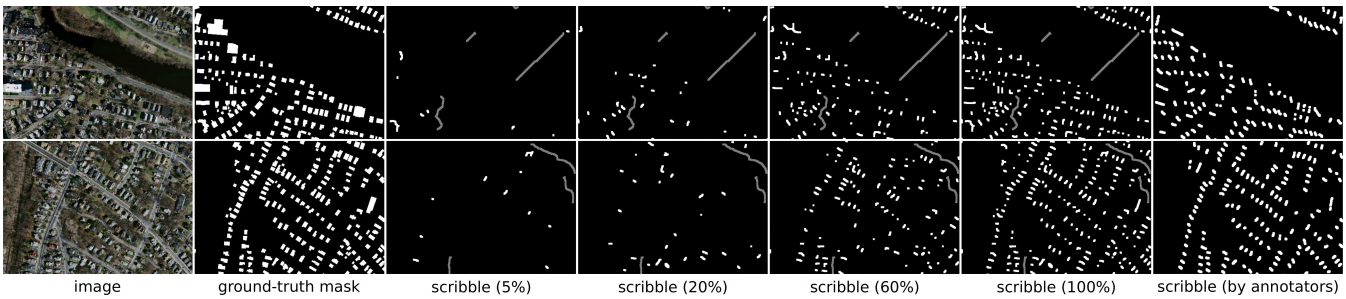


FIGURE 4. Scribble generation examples with different scribble instance rates.

Algorithm 1 Scribble Generation via Random Walks

procedure SEARCH(T)
  $\mathbf{p} = [\frac{1}{8}, \frac{1}{8}, \frac{1}{8}, \frac{1}{8}, \frac{1}{8}, \frac{1}{8}, \frac{1}{8}, \frac{1}{8}]$, $\mathbf{S} = \{\text{Skeleton}\}$
  $\mathbf{D} = \{\text{8-neighbor offset vector}\}$, $\mathbf{L} = \emptyset$
  Randomly sample a coordinate c from $\mathbf{S}$
  $\mathbf{L} = \mathbf{L} \cup c$
  for t in $\{1 \cdots T\}$
    MOVE(c, $\mathbf{p}$, $\mathbf{L}$)
  return $\mathbf{L}$ ▷ scribble coordinates

procedure MOVE(c, $\mathbf{p}$, $\mathbf{L}$)
  $\gamma = 0.05$
  $c_d \sim Cat(\mathbf{p})$ ▷ $c_d \in \mathbf{D}$, sample a direction
  if $(c_d + c) \in \mathbf{S}$
    $c = c_d + c$ ▷ move to the new position
    if $c \notin \mathbf{L}$
      $\mathbf{L} = \mathbf{L} \cup c$
      $\mathbf{p}_d = \mathbf{p}_d + \gamma$
      $\mathbf{p}_{\text{neighbour}(d)} = \mathbf{p}_{\text{neighbour}(d)} + 0.5\gamma$
      Normalize $\mathbf{p}$
  else
    $\mathbf{p}_d = \mathbf{p}_d - \gamma$
    Normalize $\mathbf{p}$

skeleton and not visited before, we reward this direction with a constant γ and its two neighboring directions with 0.5γ. The neighboring directions of any given direction refer to its two closest ordinal directions (e.g. north-east and north-west for north, south and west for south-west). The rewarding mechanism encourages the walker to move along a particular direction rather than jump randomly. Similarly, we punish the direction which would take the walker off the skeleton. After each reward or punishment, we re-normalize $\mathbf{p}$ to sum up to one. Trace of the random walker is recorded as the scribble. We empirically set $\gamma = 0.05$. This simple RW algorithm is quite effective in our experiments. A visually acceptable scribble can be generated within 150 moves. For each training sample, we randomly sample 5∼7 background scribbles. Using the proposed automatic scribble generation pipeline, we can also investigate the dynamics between scribble numbers and segmentation performance, as shown in Figure 4.

B. OBB-SCRIBBLE FOR INSTANCE SEPARATION

As shown in Figure 2, predictions made by the pCE baseline do not preserve well-separated boundaries: many building instances are merged into a large blob. We argue this is due to the fact that no structured loss is applied to enforce separation. With a limited amount of training examples, models fail to learn this important property automatically, leading to a reasonable IoU score with a rather inferior visual outcome. To overcome this limitation, we present an adversarial architecture that learns a structured loss function to capture special characteristics of one image domain and translate into the other domain in the absence of paired training examples, much like the unpaired image-to-image

FIGURE 5. Generate obb-scribble masks via OBB fitting.

translation task [45]. In aerial building footprint segmentation, buildings have straight boundaries in nature. The idea is to generate masks that resemble the ground-truths in terms of *boundary straightness* and *instance separation*, then use them to regularize the training so that model predictions would have straighter boundaries, separated instances and better IoU performance, if possible. Given input images and their scribbles from training set, we translate scribble annotations to corresponding *obb-scribble* masks, where superpixels[1] connected by each scribble are enclosed by an oriented bounding box (OBB). Unlike the axis-aligned bounding box used in object detection, an OBB is not necessarily parallel to the axes, just like how buildings normally distribute in aerial images, as illustrated in Figure 5. Given a 2D or 3D point set, an OBB can be easily generated using the principal component analysis (PCA) algorithm. In our 2D case, we use PCA to find two orthogonal axes such that the sum of Euclidean distances from one axis is minimized. We project points onto each axis to determine OBB's center and form a rectangle that encloses the point set, as shown in Figure 5 and Figure 6. Please refer to [53] for more topics on OBB. An obb-scribble mask is generated by fitting OBBs for each scribble instance respectively. Although obb-scribble masks do not highly resemble how buildings are arranged in ground-truth masks, they preserve two appealing visual properties: boundary straightness and instance separation. To this end, we introduce an adversarial loss to make our model prediction resemble obb-scribble masks.

C. OBJECTIVE FUNCTIONS

From the perspective of domain transfer, our goal is to learn a mapping function between aerial image domain X and building footprint mask domain Y given training samples $\{\mathbf{x}_i, \mathbf{s}_i, \mathbf{o}_i\}_{i=1}^N$. Here $\mathbf{x}_i \in X$ is the input image, $\mathbf{s}_i$ is the scribble map and $\mathbf{o}_i$ is the obb-scribble mask. Our model includes a generator $G : X \to Y$ and a discriminator D that aims to distinguish between obb-scribble maps $\{\mathbf{o}_i\}$ and model predictions $\{G(\mathbf{x}_i)\}$ in order to separate building instances. We denote these data distributions as $\mathbf{x} \sim p(X)$, $\mathbf{y} \sim p(Y)$ and $\mathbf{o} \sim p(O)$. Our objective function contains two types of losses: (1) a partial cross entropy (pCE) loss to reduce errors between model predictions and scribbles; 2) adversarial losses to regularize the data distribution of model predictions to match $p(O)$.

1) PARTIAL CROSS ENTROPY LOSS

By definition, a scribble annotation is a subset of the corresponding mask annotation. Therefore, $\mathbf{s}_i$ can be viewed as a set of samples drawn directly from $\mathbf{y}_i \in Y$, which provides us with a partition of ground truth samples. Given input image $\mathbf{x}_i$, we denote the model prediction $G(\mathbf{x}_i)$. In [34], Tang *et al.* proposed to use a partial cross entropy (pCE) loss:

$$\mathcal{L}_p(\mathbf{x}_i, \mathbf{s}_i) = \sum_{x \in \mathbf{x}_i} -u^x [\mathbf{s}_i^x \log G(\mathbf{x})_i^x + (1 - \mathbf{s}_i^x) \log(1 - G(\mathbf{x})_i^x)] \quad (1)$$

where the superscript x denotes a pixel in $\mathbf{x}_i$. u^x is 1 if pixel x is covered by $\mathbf{s}_i$ and 0 otherwise. A pCE loss only back-propagates gradients for pixels that are annotated by scribbles. This turns out to be a simple but effective approach which essentially chooses correctness over sample size, while the earlier approach [8] did the opposite by training directly through the entire (inaccurate) segmentation proposal.

2) ADVERSARIAL LOSS

We apply the adversarial loss [9] to both the generator G and the discriminator D. The objective is defined as:

$$\mathcal{L}_{GAN}(G, D, X, O) = \mathbb{E}_{\mathbf{x} \sim p(X)}[\log(1 - D(G(\mathbf{x})))] + \mathbb{E}_{\mathbf{o} \sim p(O)}[\log D(\mathbf{o})] \quad (2)$$

where D aims to distinguish model predictions $G(\mathbf{x})$ from obb-scribbles, while G aims to map input images $\mathbf{x}$ to look more similar to obb-scribbles in terms of boundary straightness, hopefully. The process can be summarized as a min/max game: $\min_G \max_D \mathcal{L}_{GAN}(G, D, X, O)$.

3) FULL LOSS OF ScrGAN

The full objective function is defined by:

$$\mathcal{L} = \mathbb{E}_{\mathbf{x} \sim p(X)}[\mathcal{L}_p(\mathbf{x}, \mathbf{s})] + \lambda \mathcal{L}_{GAN}(G, D, X, O) \quad (3)$$

where λ sets the relative importance of two types of loss functions. The optimal solution is $G^* = \arg\min_G \max_D \mathcal{L}$, which is searched via gradient descent optimization. From a high-level point of view, the discriminator can be seen as a learnable structured loss function for the generator, unlike the traditional closed-form losses such as cross entropy or mean square errors [54]. During training, this structured loss is also updating its parameters to deal with generator's adaptation.

D. NETWORK ARCHITECTURE AND TRAINING DETAILS

ScrGAN is illustrated in Figure 6. The dash line marks steps that are needed during training. For inference, we only need the generator network and the post-processing module. For the generative network G, we adapt the U-Net architecture to have VGG-11's first eight convolutional layers as the encoder [25], [55]. For the decoder, we interleave

[1] In practice, we find using superpixels does not help improve segmentation performance significantly, compared to fitting oriented bounding box (OBB) on scribbles directly. So we make it an optional module and report results without it. We argue it will be more useful in segmenting larger building instances.

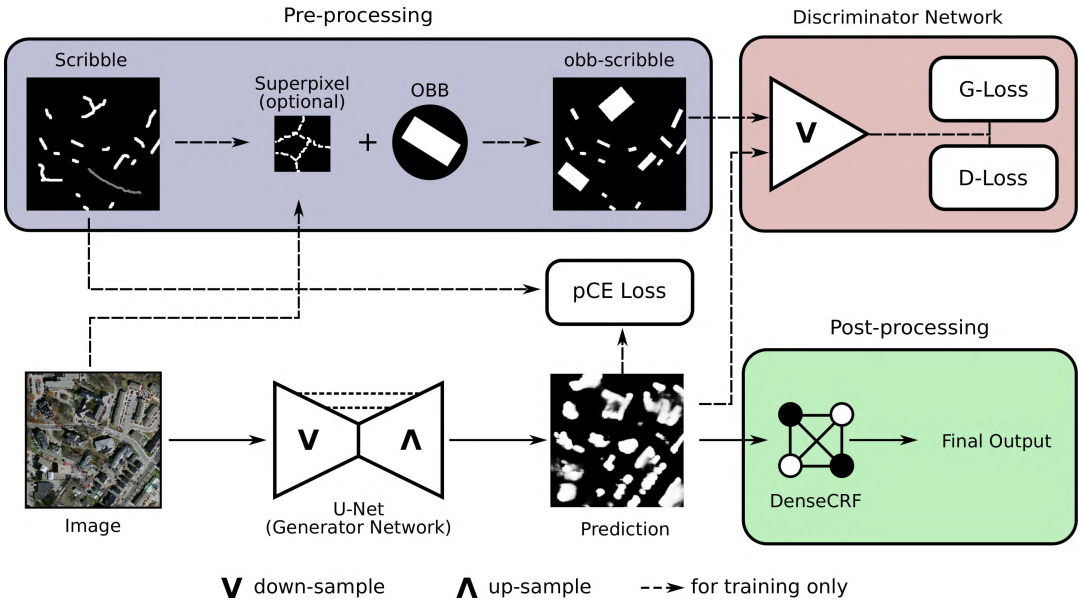


FIGURE 6. ScrGAN takes image-scribble pairs and learns to segment building footprints via the joint supervision of pCE loss and adversarial losses.

five transposed convolutional layers and six convolutional layers. The entire architecture of G is similar to Iglovikov's TernausNet [56], except that we replace all ReLUs with Leaky ReLUs (with a negative slope of 0.2), which gives a better performance in our experiments. We used their Kaggle Carvana[2] pre-trained model to initialize the generator network's encoder part. For the decoder part, weights of the convolutional layers are initialized by sampling from a normal distribution $\mathcal{N}(0, 0.02)$ and batch-norm layers from $\mathcal{N}(1, 0.02)$, as suggested in [44]. For the discriminator network D (Table 2), we use four stride-2 convolutional layers, one average pooling layer and two fully-connected layers to build up a simple binary classifier. Following [44], we use leaky ReLUs with a negative slope of 0.2 and batch normalization in D. A dropout layer with $p = 0.2$ is added to fight against over-fitting [57]. For all experiments, we set $\lambda = 0.25$ in Equation 3 as we find it gives the best segmentation performance. We use the Adam optimizer with a batch size of 2 [58]. The initial learning rate is set to 0.0001 and begins to linearly decay after 25 epochs until it reaches zero after another 25 epochs. All major hyper-parameters are chosen via grid search. Our implementation is based on PyTorch on a Linux environment with one 12GB NVIDIA GPU card.

IV. EXPERIMENTS

A. DATASETS

The Massachusetts Buildings Dataset consists of 151 aerial images of the Boston area. Each image is of size 1500×1500 px and covers an area of 2.25 square kilometers. The entire dataset covers roughly 340 square kilometers. The ground-truth masks were obtained by rasterizing building footprints from the OpenStreetMap project.[3] As shown in Figure 1, the dataset covers buildings of various sizes and shapes in urban and suburban regions. There are two semantic classes: building and non-building (background). We use the official split of this dataset: 137 images for training, 4 for validation and 10 for testing. We further crop each image into 9 non-overlapping patches of 500×500 px, augmenting each set by 9 times in size. During training, we randomly crop a patch of 384×384 px from the input image and ground-truth mask, respectively. For testing, we resize the input to 448×448 px.

TABLE 2. Architecture of the discriminator network D.

Layer	Depth	Kernel	Stride	Padding	Activation
conv_1	1→64	4×4	2	1	Leaky ReLU
conv_2	64→64	4×4	2	1	BN+Leaky ReLU
conv_3	64→64	4×4	2	1	BN+Leaky ReLU
conv_4	64→64	4×4	2	1	BN+Leaky ReLU
avg_p	64	24	-	-	-
fc_1	64→16	-	-	-	Tanh
dropout	-	-	-	-	-
fc_2	16→1	-	-	-	Sigmoid

B. EVALUATION METRICS

In our experiments, we evaluate models using the following well-known metrics: 1) intersection over union (IoU) of a class; 2) precision: the fraction of true positives among all predictions of a class; 3) recall: the fraction of true positives over the ground truth of a class; 4) F1 score: the harmonic average of the precision and recall; 5) pixel accuracy (Acc.): the fraction of correctly classified pixels over all pixels. For all evaluation metrics except Acc., we report the individual score of each class as well as the mean score over two classes.

[2]https://www.kaggle.com/c/carvana-image-masking-challenge

[3]https://www.openstreetmap.org/

C. COMPARISON AGAINST BASELINES

We train several baseline models and evaluate their performances using the aforementioned metrics. Some of the baselines come from previous literatures while the rest are variants of ScrGAN by removing certain modules.

1) FULL MASK

Without changing network architecture, we train the U-Net with the ground-truth masks. Results of this standard approach can be viewed as the upper-bound of weakly-supervised learning models. We expect certain degree of performance degradation when replacing the full masks with scribbles. Lowering performance gaps between models trained on these two types of supervision signals is the major concern of almost all weakly-supervised learning algorithms.

2) SEGMENTATION PROPOSAL

Using scribbles as seeds to generate segmentation proposals for FCN training is a natural strategy. Following [8], we use the well-known GrabCut [40] interactive segmentation algorithm to generate masks and train our U-Net. Note that these segmentation proposals usually contain a large amount of erroneous labels that may potentially lead to inferior segmentation performance. For simplicity, we address this model as *Grab-cut*.

3) NORMALIZED CUT LOSS [34]

This algorithm achieves the state-of-the-art performance on PASCAL-Scribble Dataset [8]. Besides pCE loss, they introduced a relaxed normalized cut loss that aimed to lower normalized cut energies of the model outputs based on the observation that lower energies typically correspond to better semantic segmentations. However, this observation may not hold very well in aerial image segmentation, as normalized cut has the bias towards equal segments. Moreover, approximating normalized cut energies in [34] involved computing fast bilateral filtering using permutohedral lattice [59], which can be inefficient. In our experiments, we use a GPU version of the permutohedral lattice approximation algorithm to accelerate training processes. In [34], Tang *et al.* first trained an FCN with pCE loss only and then fine-tuned the model by adding the normalized cut loss. This strategy led to a better segmentation performance. For simplicity, we address this model as *Norm-cut*, with three variants: (1) apply pCE and normalized cut loss when the training begins (*no wait*); (2) apply pCE only and then add normalized cut loss after 2 epochs[4] (*wait 2 epochs*); (3) add extra adversarial loss on *no wait* (*+GAN*). In addition, we train a UNet with the pCE loss alone, i.e. the pCE baseline mentioned in Section I and report the results (*pCE only*).

4) CRF POST-PROCESSING

Dense CRF [28] has become one of the most successful post-processing techniques for semantic segmentation. We report evaluation metrics before and after CRF post-processing.

[4]We find 2 epochs yields the best performance.

D. SENSITIVITIES TO SCRIBBLE INSTANCE NUMBER

Scribble annotation is subject to a lot of factors, such as annotators' experience, scribble length and number. Lin *et al.* [8] investigated the sensitivity of their method to scribble quality by reducing scribble length towards 0 (point). They reported a decreasing mean IoU score as the length was reduced. Regarding scribble number, we notice all object instances, however small or big, need to be annotated with at least one scribble according to their annotation protocol [8]. In aerial image interpretation, where objects tend to be small and densely distributed, is it necessary to annotate all building instances within a map to obtain quality segmentation? We find this to be an interesting topic that no one looked into previously. Using the scribble generation approach, we investigate how sensitive our model is to the scribble instance number, by controlling the number of building instances from which our algorithm generates scribbles. For the Massachusetts Buildings Dataset, where the number of building instances within an image can easily exceed 100, we set seven levels of scribble number: 5%, 10%, 20%, 40%, 60%, 80% and 100%, as shown in Figure 4. We train the baselines and ScrGAN variants on these levels of scribble annotations respectively.

As a reference, we also manually annotate this dataset with scribbles. Our scribble annotations are labeled by 2 annotators. Each image is first labeled by one annotator, and then examined by the other. Similar to [8], the annotators are asked to draw scribbles on the regions that they find confident; building boundaries do not need to be annotated. As opposed to [8], our annotation follows several subjective requirements: (1) annotators are allowed to adjust scribble width up to 5 pixels if needed; (2) annotators can stop annotating immediately after she/he feels confident that the existing scribbles would suffice to make good segmentation, which means *not all* building instances in the aerial images need to be annotated; (3) background is not annotated. These rules are designed to simulate annotations made by non-experts, as we are interested in investigating whether our model can take a perceptually acceptable annotation by non-experts and still produce reasonably quality segmentation. To make a fair comparison, we use the same sets of automatically generated background scribbles for all trainings since we find background annotation is much more inclined to subjectivity than building annotation due to its diversity.

E. ANALYSIS OF EXPERIMENTAL RESULTS

Table 3 shows the IoU and Acc. of all models on the Massachusetts Buildings Dataset. Models trained with Grab-cut segmentation proposals does not show good performance, especially in low scribble instance levels (5% and 10%) where the building IoU is too low to be reliable at all. This is due to the fact that Grab-cut segmentation proposals contain numerous incorrect labels from computations of low-level features without considering contexts. This drawback is enlarged in aerial image segmentation, where high-level

semantics is extremely useful to distinguish numerous non-overlapping buildings from the background. As an ablation study, we investigate loss terms in [34], including pCE loss, normalized cut loss, and our adversarial loss. Using pCE loss alone leads to a strong baseline: starting from 5% instance level (IoU 69.1%), the performance drops as instance level increases. Remember that we fix background scribble numbers (5∼7 background scribbles per image) and vary only the building scribble instance level from 5% to 100%. We argue the performance drop is caused by data imbalance since semantic segmentation via FCN is essentially a classification task. The data is most well-balanced at 5% instance level, where the size of building class and the background class is closest to each other. As we increase scribble instance level, the building class gradually dominates, causing the performance drop. Therefore in this case, it is not always appreciated to incorporate more scribbles for one class alone. A well-balanced scribble dataset is expected to yield better performance than the imbalanced ones. On average, our proposed ScrGAN surpasses baseline *pCE only* by a margin of 4.9% IoU. This is achieved by incorporating the obb-scribble adversarial loss term in Equation 2, which validates the effectiveness of introducing adversarial learning.

The normalized cut loss, on the other hand, does not help improve the performance in our task. Observe the IoU score drops 6.1% in *no wait* and 5.5% in *wait 2 epochs* on average when applying normalized cut loss. In Figure 7, we demonstrate the training process of one of our baseline model *Norm-cut (wait 2 epochs)*. From the top row, we show that the normalized cut energy decreases as training goes, while the mean IoU does not improve substantially. Meanwhile in the bottom row, we present the model prediction of one randomly chosen training sample. During the first 2 epochs, pCE loss controls the training and behaves reasonably well (Iteration 1000). However, after the normalized cut loss is introduced, it gradually dominates the training and produces fuzzy boundaries, preventing pCE loss from going down. We argue this is because the normalized cut term treats each pixel as a vertex and tries to reconcile global cut, which inevitably counteracts pCE loss, causing inconsistency near the boundaries. We observe similar behaviors on all *Norm-cut* variants, no matter how we adjust the weight of normalized cut loss term. To mitigate this issue, a potential modification is to instead use superpixels as vertices. This is beyond the scope of this work and will be investigated in the future work. Finally, we add the proposed adversarial loss term to the *Norm-cut (no wait)* model and observe a minor performance increase overall. The regularization effect of adversarial loss appears to mitigate the fuzziness issue to some extent. We further compare the precision, recall and F1 scores between ScrGAN and *no wait* and display the results in Figure 8. It shows the superior performance of ScrGAN, especially in terms of recall.

The proposed ScrGAN model surpasses all baseline models in IoU scores at all scribble instance levels. The best one achieves a mean IoU of 73.2% and Acc. of 90.4%

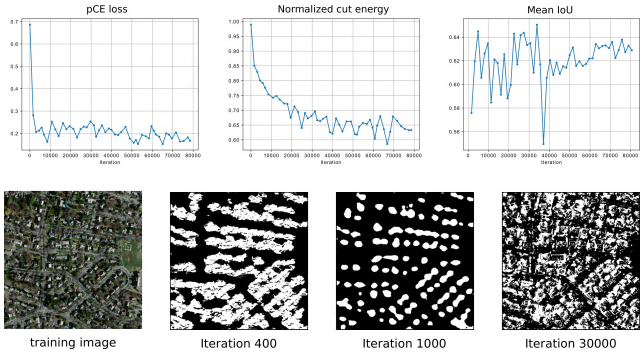


FIGURE 7. Training process of the baseline model *Norm-cut (wait 2 epochs)* [34].

after CRF post-processing, which is lower than the mask-supervised model by only 5.2% in IoU and 2% in Acc. With only 5% scribble instances, our model still achieves a mean IoU of 70.8%. We believe this is a reasonable gap between mask-supervised and scribble-supervised models. Compared with all scribble-supervised baselines including the current state-of-the-art [34], our ScrGAN not only has better quantitative performance, but also undermines building instance separation issues to a large extent (bottom row in Figure 9). Moreover, ScrGAN performs steadily under different scribble instance levels, suggesting its strong robustness towards data imbalance.

In Table 3, we report experimental results before and after applying CRF post-processing [28]. Unlike PASCAL, building footprint segmentation on the Massachusetts Buildings Dataset does not benefit from this technique considerably. On average, the performance is boosted by the margin less than 1% in IoU and less than 0.5% in Acc. This suggests using low-level color/spatial features alone may not suffice in complex aerial imagery settings.

Table 3 presents results of models trained with human scribble annotations in the *manual* column. In general, we find these models resemble those trained with higher scribble instance levels (60%, 80%, 100%) and suffer from data imbalance as well. It indicates our annotators behave conservatively when it comes to terminating the annotation. ScrGAN achieves a 70.4% mean IoU using human scribble annotations, suggesting our model has the potential for real-world aerial image interpretation tasks.

F. LIMITATIONS AND DISCUSSIONS

In ScrGAN, we introduce the OBB-based shape prior to a U-Net segmentation model via adversarial learning. The shape is a simple rectangle, which is easy to generate yet somewhat unrealistic since building footprints can take various geometric shapes. The flexibility of adversarial learning enables us to incorporate a structured loss via the discriminator network. This loss is implicitly enforced to make generator's predictions resemble obb-scribble masks. As shown in Figure 9, our ScrGAN alleviates boundary

TABLE 3. Experimental results on the Massachusetts Buildings Dataset. For each cell, the first row is in the format of IoU [Acc.] and the second row (background IoU, building IoU), all in percentage.

Model	CRF	Scribble instance level							
		5%	10%	20%	40%	60%	80%	100%	manual
Grab-cut [8]	before	39.3 [73.9] (73.5, 5.0)	42.8 [78.9] (78.5, 7.1)	37.0 [56.0] (48.1, 25.8)	25.1 [40.2] (27.4, 22.8)	22.6 [36.9] (22.8, 22.4)	39.3 [59.0] (51.7, 27.0)	34.6 [52.6] (43.1, 26.0)	19.6 [32.8] (17.5, 21.7)
	after	38.6 [73.9] (73.7, 3.4)	42.6 [79.6] (79.3, 5.8)	36.1 [54.7] (46.2, 25.9)	24.5 [39.4] (26.3, 22.6)	22.4 [36.7] (22.6, 22.3)	38.9 [58.4] (50.8, 27.1)	34.1 [51.9] (42.0, 26.2)	19.3 [32.4] (16.9, 21.6)
Norm-cut [34] (*no wait*)	before	63.5 [83.8] (81.2, 45.8)	63.4 [83.9] (81.4, 45.5)	62.6 [82.4] (79.3, 45.8)	60.8 [81.0] (77.7, 43.9)	59.7 [80.8] (77.8, 41.5)	58.2 [78.0] (73.6, 42.8)	56.8 [76.8] (72.3, 41.3)	53.9 [73.8] (68.5, 39.3)
	after	63.5 [83.7] (81.1, 45.9)	63.3 [83.8] (81.3, 45.4)	62.7 [82.4] (79.3, 46.0)	60.8 [81.0] (77.7, 44.0)	59.7 [80.8] (77.7, 41.7)	58.4 [78.1] (73.8, 43.0)	56.9 [76.9] (72.3, 41.5)	54.0 [73.9] (68.6, 39.5)
Norm-cut [34] (*wait 2 epochs*)	before	62.0 [82.9] (80.2, 43.7)	61.4 [83.0] (80.5, 42.2)	64.0 [83.2] (80.2, 47.9)	63.1 [82.7] (79.6, 46.6)	59.8 [80.1] (76.6, 43.0)	59.3 [78.9] (74.8, 43.8)	59.3 [79.1] (75.1, 43.6)	57.8 [78.0] (73.9, 41.8)
	after	61.9 [82.7] (80.1, 43.6)	61.2 [82.9] (80.4, 42.0)	64.0 [83.2] (80.1, 47.9)	63.6 [83.0] (79.9, 47.3)	60.0 [80.2] (74.9, 44.0)	59.5 [79.0] (74.9, 44.0)	59.6 [79.2] (75.2, 43.9)	58.4 [78.5] (74.4, 42.4)
Norm-cut [34] (+ *GAN*)	before	65.4 [85.7] (83.6, 47.3)	65.2 [85.6] (83.5, 46.9)	64.0 [83.3] (80.3, 47.8)	65.7 [85.1] (82.6, 48.8)	54.6 [76.3] (72.6, 36.7)	58.9 [78.8] (74.7, 43.1)	61.6 [83.8] (81.7, 41.4)	54.6 [74.8] (69.9, 39.4)
	after	66.4 [86.2] (84.2, 48.6)	65.9 [86.1] (84.0, 47.8)	64.6 [83.7] (80.8, 48.5)	66.5 [85.6] (83.2, 49.8)	54.7 [76.4] (72.6, 36.8)	59.2 [78.9] (74.8, 43.5)	61.6 [83.8] (81.7, 41.5)	55.0 [75.0] (70.1, 39.8)
pCE only [34]	before	68.4 [87.4] (85.4, 51.3)	67.9 [86.5] (84.2, 51.5)	68.0 [86.6] (84.3, 51.6)	68.1 [86.1] (83.6, 52.6)	66.8 [85.1] (82.3, 51.3)	64.1 [82.5] (79.0, 49.2)	64.3 [83.0] (79.7, 48.9)	64.8 [83.6] (80.6, 49.1)
	after	69.1 [87.8] (86.0, 52.2)	68.4 [86.8] (84.6, 52.3)	68.5 [86.9] (84.8, 52.2)	68.6 [86.4] (84.0, 53.1)	67.5 [85.6] (82.9, 52.1)	64.4 [82.8] (79.2, 49.6)	64.6 [83.2] (80.0, 49.3)	65.1 [83.8] (80.8, 49.5)
ScrGAN (ours)	before	70.3 [88.7] (87.0, 53.7)	70.4 [88.0] (85.9, 55.0)	71.6 [88.7] (86.7, 56.5)	72.4 [89.7] (88.1, 56.7)	71.9 [89.4] (87.8, 56.0)	72.9 [90.3] (88.9, 57.0)	72.8 [88.8] (87.7, 58.0)	69.8 [89.1] (87.6, 52.0)
	after	**70.8** [89.0] (87.3, **54.2**)	**71.0** [88.3] (86.3, **55.6**)	**72.2** [89.0] (87.2, **57.3**)	**72.8** [89.9] (88.3, **57.3**)	**72.1** [89.5] (87.9, **56.4**)	**73.2** [90.4] (89.0, **57.3**)	**73.1** [88.9] (87.9, **58.4**)	**70.4** [89.3] (87.9, **52.9**)
Full mask (for reference)	before	78.4 [92.4] (91.1, 65.7)							
	after	78.4 [92.4] (91.1, 65.8)							

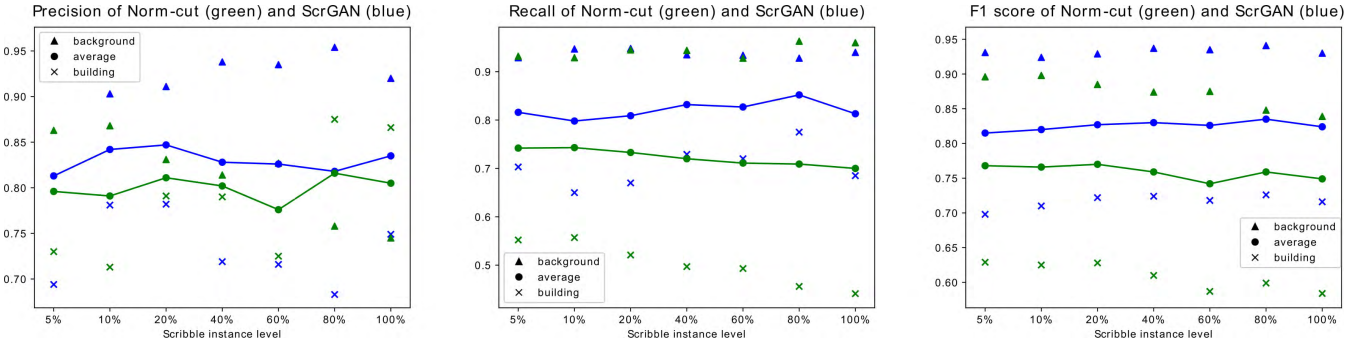


FIGURE 8. Precision, recall and F1 scores between *no wait* and ScrGAN.

separation issue significantly and outperforms all baselines in terms of evaluation metrics such as IoU. However, boundary straightness characteristics of obb-scribble masks is not well preserved in our predictions. The resulting segmentation masks have blobby shapes. We argue it is more challenging to preserve straight boundaries in this case, even for the strongly-supervised method (3^{rd} column in Figure 9). An intuitive extension of ScrGAN is to consider more advanced shape priors such as convex hull or even customized prior in a semi-supervised learning setting. Moreover, note that our generator is a simplified version of the original U-Net. This suggests it can be easily upgraded into more powerful architectures such as DeepLab [27] and deep residual networks [60], thus yield better segmentation performance without bells and whistles. We leave this to the future work. In the current python implementation, where all reported models use the same U-Net architecture, we achieve a GPU inference time of 130 ms per image, compared to a CPU inference time of 3500 ms per image on a commercial-level laptop with an Intel Core i7-4700HQ CPU, 16 GB RAM and a 4GB NVIDIA GTX 860M GPU. This indicates that a GPU implementation has an edge over the CPU one in terms of testing speed. In Figure 10, we show several failure cases of ScrGAN, where dot noise (also known as checkboard artefacts [61]) is somehow introduced into the model prediction. We argue this may result from the use of transposed convolutional layers in our architectures, causing the instability of adversarial learning. In future work, we plan to examine

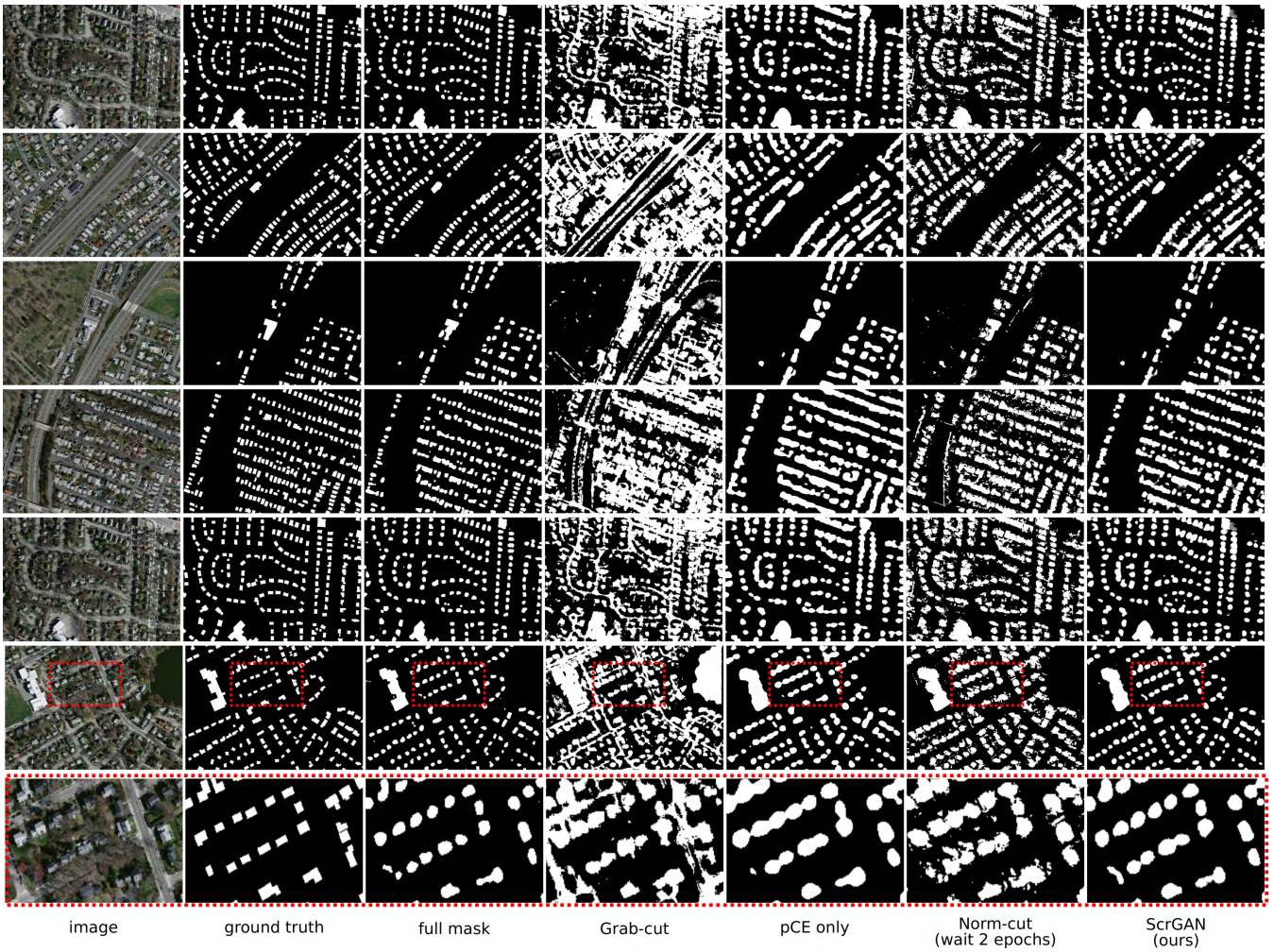


FIGURE 9. Our results on the Massachusetts Buildings Dataset test set illustrates ScrGAN helps alleviate building instance separation issue, thus achieves better segmentations. The bottom row is a zoomed-in look of the red boxes in the row above. See also Table 3 for quantitative results.

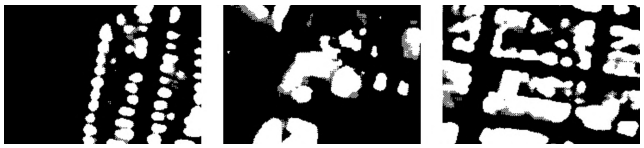


FIGURE 10. Failure cases of ScrGAN.

alternative architectures and techniques such as using nearest-neighbor upsampling layers and PatchGAN [45], [54] to resolve this issue.

V. CONCLUSION

We have presented a weakly-supervised learning method for aerial building footprint segmentation based only on scribble annotations. Traditional strongly-supervised methods substantially rely on costly and time-consuming pixel-level mask annotation of the entire set of training images. In this work, we have explored a simple alternative: annotate images with scribbles and train a deep neural network to predict building footprint segmentation directly. We observe that previous scribble-supervised methods are likely to cause building instance separation issue, i.e. model predictions contain large building blobs instead of separated building instances. We propose an adversarial learning architecture to overcome this issue. First, we generate obb-scribble masks (with negligible computational overhead) given image-scribble pairs. It has promising characteristics including boundary straightness and instance separation, characteristics that we would like the model prediction to possess. Second, we take a U-Net as a generator network and design a discriminative network to regularize the generator so that its output gradually resemeble obb-scribble masks. We have conducted a number of experiments to verify the effectiveness of our ScrGAN model, including sensitivities toward scribble instance number. Our model is proved to have better segmentation performance compared with a variety of baselines including the state-of-the-art scribble-supervised algorithms on PASCAL Scribble Dataset. We believe scribble-based methods can substantially shorten annotation time and have the potential to be

applicable for many other types of aerial image interpretation tasks such as aerial object detection and crop management.

ACKNOWLEDGMENT

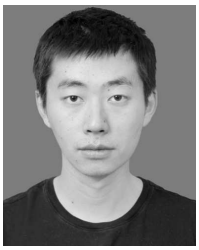
(Weimin Wu and Huan Qi contributed equally to this work.)

REFERENCES

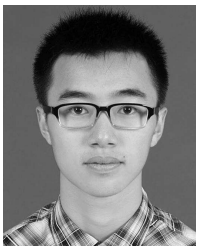
[1] V. Mnih, "Machine learning for aerial image labeling," Ph.D. dissertation, Univ. Toronto, Toronto, ON, Canada, 2013.
[2] X. X. Zhu *et al.*, "Deep learning in remote sensing: A comprehensive review and list of resources," *IEEE Geosci. Remote Sens. Mag.*, vol. 5, no. 4, pp. 8–36, Dec. 2017.
[3] Y. Ma *et al.*, "Remote sensing big data computing: Challenges and opportunities," *Future Generat. Comput. Syst.*, vol. 51, pp. 47–60, Oct. 2015.
[4] L. Zhang, L. Zhang, and B. Du, "Deep learning for remote sensing data: A technical tutorial on the state of the art," *IEEE Geosci. Remote Sens. Mag.*, vol. 4, no. 2, pp. 22–40, Jun. 2016.
[5] P. Kaiser, J. Wegner, A. Lucchi, M. Jaggi, T. Hofmann, and K. Schindler, "Learning aerial image segmentation from online maps," *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 11, pp. 6054–6068, Nov. 2017.
[6] L. Castrejón, K. Kundu, R. Urtasun, and S. Fidler, "Annotating object instances with a polygon-RNN," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 5230–5238.
[7] S. Bell, P. Upchurch, N. Snavely, and K. Bala, "OpenSurfaces: A richly annotated catalog of surface appearance," *ACM Trans. Graph.*, vol. 32, no. 4, 2013, Art. no. 111.
[8] D. Lin, J. Dai, J. Jia, K. He, and J. Sun, "ScribbleSup: Scribble-supervised convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 3159–3167.
[9] I. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.
[10] R. Bajcsy and M. Tavakoli, "Computer recognition of roads from satellite pictures," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-6, no. 9, pp. 623–637, Sep. 1976.
[11] M. M. Rathore, A. Ahmad, A. Paul, and S. Rho, "Urban planning and building smart cities based on the Internet of Things using big data analytics," *Comput. Netw.*, vol. 101, pp. 63–80, Jun. 2016.
[12] L. Liu, H. Qi, and W. Wu, "A remote sensing map based solution to path planning problem," in *Proc. IEEE Int. Conf. Autom. Logistics (ICAL)*, Aug. 2012, pp. 173–178.
[13] J. M. Peña, P. A. Gutiérrez, C. Hervás-Martínez, J. Six, R. E. Plant, and F. López-Granados, "Object-based image classification of summer crops with machine learning methods," *Remote Sens.*, vol. 6, no. 6, pp. 5019–5041, 2014.
[14] G. Mountrakis, J. Im, and C. Ogole, "Support vector machines in remote sensing: A review," *ISPRS J. Photogramm. Remote Sens.*, vol. 66, no. 3, pp. 247–259, 2011.
[15] V. Mnih and G. E. Hinton, "Learning to label aerial images from noisy data," in *Proc. 29th Int. Conf. Mach. Learn. (ICML)*, 2012, pp. 567–574.
[16] J. Shotton, J. Winn, C. Rother, and A. Criminisi, "TextonBoost for image understanding: Multi-class object recognition and segmentation by jointly modeling texture, layout, and context," *Int. J. Comput. Vis.*, vol. 81, no. 1, pp. 2–23, Dec. 2007.
[17] B. Fulkerson, A. Vedaldi, and S. Soatto, "Class segmentation and object localization with superpixel neighborhoods," in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, Sep./Oct. 2009, pp. 670–677.
[18] J. Porway, K. Wang, B. Yao, and S. C. Zhu, "A hierarchical and contextual model for aerial image understanding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2008, pp. 1–8.
[19] S. Kluckner, T. Mauthner, P. M. Roth, and H. Bischof, "Semantic classification in aerial imagery by integrating appearance and height information," in *Proc. Asian Conf. Comput. Vis.* Berlin, Germany: Springer, 2009, pp. 477–488.
[20] J. Carreira and C. Sminchisescu, "CPMC: Automatic object segmentation using constrained parametric min-cuts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 7, pp. 1312–1328, Jul. 2011.
[21] J. Carreira, R. Caseiro, J. Batista, and C. Sminchisescu, "Semantic segmentation with second-order pooling," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, 2012, pp. 430–443.
[22] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, p. 436, 2015.
[23] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 3431–3440.
[24] H. Noh, S. Hong, and B. Han, "Learning deconvolution network for semantic segmentation," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 1520–1528.
[25] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assisted Intervent.* Cham, Switzerland: Springer, 2015, pp. 234–241.
[26] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," in *Proc. Int. Conf. Learn. Represent.*, 2015, pp. 1–13.
[27] 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. 2017.
[28] P. Krähenbühl and V. Koltun, "Efficient inference in fully connected CRFs with Gaussian edge potentials," in *Proc. Adv. Neural Inf. Process. Syst.*, 2011, pp. 109–117.
[29] V. Mnih and G. E. Hinton, "Learning to detect roads in high-resolution aerial images," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, 2010, pp. 210–223.
[30] D. Marmanis, J. D. Wegner, S. Galliani, K. Schindler, M. Datcu, and U. Stilla, "Semantic segmentation of aerial images with an ensemble of CNNs," *ISPRS Ann. Photogramm., Remote Sens. Spatial Inf. Sci.*, vol. 3, pp. 473–480, Jun. 2016.
[31] B. Bischke, P. Helber, J. Folz, D. Borth, and A. Dengel. (Sep. 2017). "Multi-task learning for segmentation of building footprints with deep neural networks." [Online]. Available: https://arxiv.org/abs/1709.05932
[32] C.-Y. Lee, S. Xie, P. Gallagher, Z. Zhang, and Z. Tu, "Deeply-supervised nets," in *Proc. 18th Int. Conf. Artif. Intell. Statist. (AISTATS)*, San Diego, CA, USA, 2015.
[33] Z.-H. Zhou, "A brief introduction to weakly supervised learning," *Nat. Sci. Rev.*, vol. 5, no. 1, pp. 44–53, 2017.
[34] M. Tang, A. Djelouah, F. Perazzi, Y. Boykov, and C. Schroers, "Normalized cut loss for weakly-supervised CNN segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1818–1827.
[35] K.-K. Maninis, S. Caelles, J. Pont-Tuset, and L. Van Gool, "Deep extreme cut: From extreme points to object segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Nov. 2018, pp. 616–625.
[36] A. Bearman, O. Russakovsky, V. Ferrari, and L. Fei-Fei, "What's the point: Semantic segmentation with point supervision," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 549–565.
[37] A. Khoreva, R. Benenson, J. Hosang, M. Hein, and B. Schiele, "Simple does it: Weakly supervised instance and semantic segmentation," in *Proc. CVPR*, 2017, vol. 1, no. 2, pp. 876–885.
[38] J. Dai, K. He, and J. Sun, "BoxSup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 1635–1643.
[39] Y. Y. Boykov and M.-P. Jolly, "Interactive graph cuts for optimal boundary & region segmentation of objects in N-D images," in *Proc. 8th IEEE Int. Conf. Comput. Vis. (ICCV)*, vol. 1, Jul. 2001, pp. 105–112.
[40] C. Rother, V. Kolmogorov, and A. Blake, "'GrabCut': Interactive foreground extraction using iterated graph cuts," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 309–314, 2004.
[41] L. Grady, "Random walks for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 11, pp. 1768–1783, Nov. 2006.
[42] J. Han, D. Zhang, G. Cheng, L. Guo, and J. Ren, "Object detection in optical remote sensing images based on weakly supervised learning and high-level feature learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 6, pp. 3325–3337, Jun. 2015.
[43] P. Zhou, G. Cheng, Z. Liu, S. Bu, and X. Hu, "Weakly supervised target detection in remote sensing images based on transferred deep features and negative bootstrapping," *Multidimensional Syst. Signal Process.*, vol. 27, no. 4, pp. 925–944, 2016.
[44] A. Radford, L. Metz, and S. Chintala. (Nov. 2015). "Unsupervised representation learning with deep convolutional generative adversarial networks." [Online]. Available: https://arxiv.org/abs/1511.06434
[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)*, Venice, Italy, 2017, pp. 2242–2251.
[46] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training GANs," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 2234–2242.

[47] P. Luc, C. Couprie, S. Chintala, and J. Verbeek. (Nov. 2016). "Semantic segmentation using adversarial networks." [Online]. Available: https://arxiv.org/abs/1611.08408
[48] J. Pan *et al.* (Jan. 2017). "SalGAN: Visual saliency prediction with generative adversarial networks." [Online]. Available: https://arxiv.org/abs/1701.01081
[49] W.-C. Hung, Y.-H. Tsai, Y.-T. Liou, Y.-Y. Lin, and M.-H. Yang. (Jul. 2018). "Adversarial learning for semi-supervised semantic segmentation." [Online]. Available: https://arxiv.org/abs/1802.07934
[50] Y. Shen, R. Ji, S. Zhang, W. Zuo, and Y. Wang, "Generative adversarial learning towards fast weakly supervised detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5764–5773.
[51] T. Remez, J. Huang, and M. Brown. (Mar. 2018). "Learning to segment via cut-and-paste." [Online]. Available: https://arxiv.org/abs/1803.06414
[52] T. C. Lee, R. L. Kashyap, and C. N. Chu, "Building skeleton models via 3-D medial surface axis thinning algorithms," *Graph. Models Image Process.*, vol. 56, no. 6, pp. 462–478, 1994.
[53] S. Gottschalk, "Collision queries using oriented bounding boxes," Ph.D. dissertation, Dept. Comput. Sci., Univ. North Carolina Chapel Hill, Chapel Hill, NC, USA, 2000.
[54] 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)*, Honolulu, HI, USA, 2017, pp. 5967–5976.
[55] K. Simonyan and A. Zisserman. (Apr. 2015). "Very deep convolutional networks for large-scale image recognition." [Online]. Available: https://arxiv.org/abs/1409.1556
[56] V. Iglovikov and A. Shvets. (Jan. 2018). "TernausNet: U-net with VGG11 encoder pre-trained on imagenet for image segmentation." [Online]. Available: https://arxiv.org/abs/1801.05746
[57] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.
[58] D. P. Kingma and J. Ba. (2014). "Adam: A method for stochastic optimization." [Online]. Available: https://arxiv.org/abs/1412.6980
[59] A. Adams, J. Baek, and M. Davis, "Fast high-dimensional filtering using the permutohedral lattice," *Comput. Graph. Forum*, vol. 29, no. 2, pp. 753–762, 2010.
[60] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 770–778.
[61] A. Odena, V. Dumoulin, and C. Olah, "Deconvolution and checkerboard artifacts," *Distill*, Oct. 2016, doi: 10.23915/distill.00003.

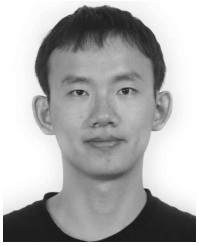
WEIMIN WU received the Ph.D. degree in control science and engineering from Zhejiang University, Hangzhou, China, in 2002. Since 2003, he has been a Faculty Member with the Department of Control Science and Engineering, Institute of Cyber-Systems and Control, Zhejiang University, where he currently is an Associate Professor. His main research interests include discrete event systems and its applications in manufacturing, transportation, self-driving, and logical automation systems.



HUAN QI was born in Hegang, Heilongjiang, China. He received the B.Eng. degree from Zhejiang University, China, and the M.A.Sc. degree in electrical and computer engineering from The University of British Columbia, Canada. He is currently pursuing the D.Phil. degree in engineering science with University of Oxford, U.K. His research interests include computer vision, machine learning, and medical image analysis.



ZHENRUI RONG is currently pursuing the B.Eng. degree in control science and engineering with Zhejiang University, Hangzhou, China. His main research interests include machine vision, intelligent transportation, and related applications.



LIANG LIU received the Ph.D. degree from the Department of Industrial and Systems Engineering, University of Southern California, Los Angeles, CA, USA. Since 2017, he has been a Software Engineer with Google Inc. His main research interest is optimization algorithm, including combinatorial optimization and stochastic optimization, and its application in transportation, machine learning, and auction.

HONGYE SU received the B.S. degree in industrial Automation from the Nanjing University of Chemical Technology, Nanjing, China, in 1990, the M.S. and Ph.D. degrees in industrial automation from Zhejiang University, Hangzhou, China, in 1993 and 1995, respectively. Since 1995, he has been a Faculty Member with the Institute of Cyber-Systems and Control, Zhejiang University, where he currently is a Professor. His recent research interests include robust control, time-delay systems, nonlinear systems, discrete event systems, and advanced process control theory and application.