How Many Annotations Do We Need for Generalizing New-Coming Shadow Images?

Wen Wu, Wenya Yang, Weiyin Ma, Member, IEEE, and Xiao-Diao Chen

Abstract—Unlabeled data is often used to improve the generalization ability of one segmentation model. However, it tends to neglect the inherent difficulty of unlabeled samples, and then produces inaccurate pseudo masks in some unseen scenes, resulting in severe confirmation bias and potential performance degradation. These motivate two unexplored questions for new-comming data: (1) How many images do we need to annotate; and (2) how to annotate them? In this paper, two kinds of shadow detectors (i.e., SDTR and SDTR+) based on the Transformer and self-training scheme are successively proposed. The main difference between them is whether weak annotations are required for partial unlabeled data. Specifically, in SDTR, we first introduce an image-level sample selection scheme to separate the unlabeled data into reliable and unreliable samples from the holistic prediction-level stability. Then, we perform selective retraining to exploit the unlabeled images progressively in a curriculum learning manner. While in SDTR+, we further provide various weak labels (i.e., point, box and scribble) for the rest unreliable samples and design corresponding loss functions. By doing this, it can achieve a better trade-off between performance improvement and annotation cost. Experimental results on public benchmarks (i.e., SBU, UCF and ISTD) show that both SDTR and SDTR+ can be favorable against state-of-the-art methods.

Index Terms—Image segmentation, shadow detection, weak label, confirmation bias, curriculum learning.

I. INTRODUCTION

SHADOW detection is essential for scene understanding [1], [2] as it can provide meaningful cues, such as light source, scene illumination, etc.. However, they also degrade the performance of many computer vision tasks [3], [4], [5], so sometimes, we need to locate [6] and remove them [7], [8]. In the past couple of years, the accuracy of shadow detectors designed only for a single domain seems to reach an upper bound. In this connection, recent works have received ever-increasing interest in improving the efficiency [9], [10] or generalization ability [11].

Traditional shadow detectors were proposed by exploring various hand-crafted features [12], [13], [14], [15]. However, these low-level features are not discriminative enough to deal with complex or web-quality shadow images. Thanks to the large amounts of labeled data, deep learning-based shadow detectors have dominated this task. That can be divided into two branches: fully supervised methods [9], [16], [17], [18], [19], [20], [21], [22] or semi-supervised methods [11]. The former usually over-fit to the training set, resulting in dramatic performance degradation on new-coming data as the domain shift [23]. Instead of fitting only one domain, a robust and practical shadow detector should be able to learn continuously with the new-coming data. For this purpose, Chen et al. [11] employ a teacher-student framework to generate pseudo masks for unlabeled data, bringing an obvious generalizability improvement. However, they treat each unlabeled sample equally and produce inaccurate pseudo masks in some unseen scenes. These improper supervisions in most unseen scenes will lead to a severe confirmation bias [24] and potential performance degradation.

A straightforward way to tackle the above problem is to fully annotate all unlabeled data. However, this pixel-wise manner is costly and laborious, especially in complex shadow scenes (e.g., soft shadow [25] and self shadow [9]). Fortunately, we found that even on a new-coming unlabeled dataset (e.g., USR [26], a shadow removal benchmark without any shadow masks), there are so many scenes that are similar to the images in the labeled dataset (e.g., SBU [16]), as shown in Fig. 1. Consequently, a teacher net well-trained by labeled data can produce relatively accurate pseudo masks for these similar images in the teacher-student setting. In contrast, they often cannot work well for the rest unseen scenes. To this end, we successively introduce two solutions to improve the quality of pseudo masks for them: (1) Exploiting the whole unlabeled data from easy to hard, which is also known as curriculum learning [27]; (2) further providing weak labels for these unseen images. These two solutions both need to separate the unlabeled data into similar images (reliable samples) and unseen images (unreliable samples), while the difference is whether the annotations are required for the unreliable samples.

To implement the first solution, we present a basic shadow detector, SDTR, in a semi-supervised learning manner. It is built on both the teacher-student framework [11], [28] and the Transformer architecture [29], [30], [31], which allow us to leverage unlabeled data and explore significant long-range dependencies, respectively. The key components in SDTR are the image-level sample selection and progressive learning scheme. Specifically, we first distinguish reliable parts from
the whole unlabeled data by measuring the stability of history predictions. Instead of identifying the uncertain pixel-level samples [32] by setting a simple threshold, our image-level manner can allow the model to focus on holistic contextual regions for more robust learning. Then, we can utilize these selected reliable samples and their corresponding pseudo masks together with labeled data to retrain a better-optimized shadow detector.

Although our SDTR can holistically improve the quality of the predicted shadow maps for unlabeled data, it still struggles in some complex regions, including tiny shadows, self-shadows and some shadow-like non-shadow (distraction) regions. Therefore, we consider annotating these regions empirically with different types of weak labels, which can be regarded as valuable cues for better shadow location. Specifically, we first annotate the USR dataset with three types of weak labels (i.e., point, box and scribble), called Mix-USR. Then, we further develop a unified architecture, SDTR+, to accommodate such various weak forms of annotations. In detail, we regard the assignment between predictions and the available weak labels as a bipartite matching problem [29], [33] and design a series of loss functions for all matched pairs. Experimental results show that we can generalize the entire USR dataset well with only 30% of weak labels.

In summary, our main contributions are as follows:

- Two solutions (i.e., SDTR and SDTR+) for shadow detection are proposed to cope with the potential performance degradation caused by incorrect pseudo masks. Experiments show that they both outperform state-of-the-art shadow detectors on public benchmarks.
- We introduce a novel image-level sample selection scheme by measuring the stability of history predictions, which allows deep model focus on the holistic contextual regions for more robust learning.
- We are the first to train a shadow detector with weak labels, especially in a mixed fashion, i.e., point, box and scribble. That enables us to achieve a better trade-off between performance improvement and annotation cost.

II. RELATED WORK

A. Shadow Detection

Traditional shadow detection methods mainly employ physical models [1], [2], [3] or machine learning classifiers based on hand-crafted features, including chromaticity [1], [2], [4], [13], [14], edge [1], [4], [12], [15], intensity [3], [12], [13], [14], [15] and texture [12], [13], [14]. However, these above methods do not work in complex shadow images since these hand-crafted features are not discriminative enough to distinguish shadow regions well from backgrounds.

In recent years, convolutional neural networks (CNNs) have achieved remarkable performance in the shadow detection community. For example, in 2016, Nguyen et al. [17] introduce a novel sensitive conditional generative adversarial network (scGAN) to extract high-level contexts better. Next, Russell et al. [6] propose a feature-based image patch approximation and multi-independent sparse representation technique to distinguish shadow points from the foreground object in many problematic situations. Le et al. [18] propose an attenuation and detection network (A+D Net) for adversarial learning, where the A-Net tries to fool the shadow predictions from the D-Net by modifying the illumination of shadow regions. Zhu et al. [19] develop a bidirectional recurrent attention residual (BDRAR) to optimize context features from deep to shallow layers. Hu et al. [20] present a direction-aware spatial context (DSC) module that uses a spatial RNN to learn the spatial context of shadows. Zheng et al. [21] propose a distraction-aware shadow detection network (DSDNet) to learn discriminative features for robust shadow detection. Recently, Chen et al. [11] introduce a multi-task mean teacher network (MTMT-Net), which uses unlabeled shadow data to further improve the generalization ability. Inoue et al. [7] extend a physically-grounded shadow illumination model and present a large-scale synthetic shadow dataset for better shadow detection. Very recently, Hu et al. [9] design a fast shadow detection network (FSDNet) by adopting the DSC [20] module. Zhu et al. [22] propose a feature decomposition and reweighting scheme (FDRNet) to mitigate the intensity bias in shadow detection.

Prevalent approaches are mainly designed for a single domain and trained in a fully-supervised manner. There is only one method (i.e., MTMT-Net [11]) to exploit the use of unlabeled data in a semi-supervised manner for generalizing other shadow scenes. However, it cannot produce accurate pseudo masks on some unseen scenes as domain shift. In this work, we aim to further improve the generalization ability of shadow detector by learning from extra training data, and then explore how to make a trade-off between the annotation cost and performance improvement.

B. Vision Transformer

Transformer, a natural language processing (NLP) model [34], has become very popular in the computer vision community. When applied to vision tasks such as image classification [35], object detection [36], and image restoration [37], it learns by exploring global interactions between different regions and focusing on important image
regions. Thanks to its excellent performance, Transformer has also been introduced for image segmentation. For example, Zheng et al. [38] treat semantic segmentation as a sequence-to-sequence prediction task and deploy a pure transformer, termed segmentation transformer (SETR). Xie et al. [30] present SegFormer, a simple yet efficient powerful semantic segmentation framework that unifies Transformers with lightweight multilayer perceptron (MLP) decoders. Lee et al. [39] propose a multi-scale patch embedding and multi-path structure scheme for Transformers (MPViT) for simultaneously representing fine and coarse features for dense prediction tasks. Considering the dependency on downstream applications, inspired by these works, we aim to design a shadow detector with a small size and low latency.

C. Self-Training

Self-training (pseudo labeling) is one of the oldest semi-supervised learning algorithms [40]. The idea is first to use labeled data to train an initial model in a fully supervised manner, then generate pseudo labels for unlabeled data, and finally retrain a new model on the union of pseudo and manually annotated labels. Recently, this technique has received attention in deep learning, such as fully-supervised image recognition [41], semi-supervised learning [42], [43] and domain adaptation [44]. However, these works tend to adopt the plainest training pipeline, while our work proposes a more advanced scheme to safely use extra training data in an easy-to-hard manner.

D. Uncertainty Estimation

The previous method [45] predicts the uncertainty of a model through Bayesian analysis. To reduce the computation cost, Gal et al. [46] propose to add a dropout layer to the neural network to approximate the probabilistic deep Gaussian process. Next, Sohn et al. [32] propose to set a confidence threshold to filter uncertain samples. Feng et al. [43] train two networks with the same structure but different initialization to highlight uncertain regions. In this work, we estimate uncertainty at the image-level by measuring the stability of history prediction, which does not require us to train other networks or manually set thresholds. In addition, it also can give us valuable cues, such as which images need to annotate.

III. PROPOSED METHODS

This section introduces two kinds of shadow detectors, SDTR and SDTR+, to reasonably utilize extra new-coming data. As shown in Fig. 2, we can observe that our two solutions both consist of a student net $S$ and a teacher net $T$. In detail, SDTR is a semi-supervised framework that aims to generalize from images with pixel-wise annotations $D_{label} = \{x, y\}$ and unlabeled images $D_{new} = \{u, \emptyset\}$. In contrast, SDTR+ is weakly supervised, in which we replace $D_{new}$ with weakly labeled dataset $D_{new} = \{u, y_w\}$ to provide student net with various weak labels.

A. Network Architecture

Considering the dependency on downstream applications, we have to design a shadow detection network with a small size and low latency, as shown in Fig. 3. The model size of our network is 24.82 M, and its inference speed can reach 212 fps.

1) Encoder: Inspired by SegFormer [30], the mix Transformer (MiT) can extract coarse-to-fine features with high efficiency, so we use MiT-B2 as the encoder of our network, as shown in Fig. 3a. Where the MiT block (Fig. 3b) can reduce the dimension of keys in multi-head self-attention (MSA) by introducing efficient self-attention. Moreover, the Mix-FFN in MiT can provide a better position representation by using convolution instead of positional encoding.

2) Decoder: To exploit global representations, we first adapt the pyramid pooling module (PPM) [45] to refine the feature maps extracted from encoder blocks. As shown in Fig. 3c, a PPM can simultaneously downsample high-level feature maps into differently scaled versions via a series
of pooling layers, then concatenate them to produce an efficient global prior representation. Moreover, inspired by UperNet [47], our decoder uses convolution blocks on multiple image scales instead of directly fusing the output of the middle and low-level blocks. Finally, we use the deep supervision scheme [48] followed by [11], [19], and [20] to guide early image scales instead of directly fusing the output of the middle UperNet [47], our decoder uses convolution blocks on multiple efficient global prior representation. Moreover, inspired of pooling layers, then concatenate them to produce an

Fig. 3. The architecture of our proposed network.

where \(\hat{y}\):

\[
L = \sum_{i=1}^{M} \frac{N_S}{N_S + N_{NS}} y_i \log \hat{y}_i + \frac{N_{NS}}{N_S + N_{NS}} (1 - y_i) \log (1 - \hat{y}_i),
\]

where \(\lambda_{\text{plain}}\) acts as a trade-off between labeled and unlabeled data.

**B. Progressive Self-Training Scheme**

Despite the above plainest self-training scheme can easily exploit unlabeled data, it treats each unlabeled sample equally without considering each independent instance’s inherent reliability and difficulty. Hence, incorrect predictions on hard examples will bring too many training noises, resulting in performance degradation. To tackle this issue, we propose a progressive self-training scheme by preferentially learning reliable data to reasonably explore the less reliable data from unlabeled samples in a curriculum learning manner [27].

The crux of our progressive self-training scheme is how to separate unlabeled data into two parts: reliable and unreliable images. Existing methods were proposed to estimate the model’s reliability or uncertainty mainly in a pixel-level manner, which can be divided into the following two types: (1) Adding dropout layers (i.e., MCDO [45], [46]) to one CNN-based model, and then retain them in the inference stage to estimate the model’s uncertainty by setting a fixed confidence threshold; (2) training two parallel networks with the same structure but different initialization, and then regard their disagreements [43] as uncertain regions. Unlike these above strategies, we wish to use only one model to estimate the reliability at the image-level without additional layers and confidence threshold. Moreover, the image-level manner allows us to learn more holistic contextual patterns.

1) Image-Level Sample Selection: We find a positive correlation between the shadow detection performance and the evolving stability of history predictions. Therefore, we can select better-predicted unlabeled images (reliable samples) based on their evolving stability in the training stage. Specifically, we save \(K\) checkpoints in the training stage and produce \(K\) corresponding pseudo masks for an unlabeled image. Furthermore, since deep networks tend to converge in the late training stage, we take the result of the last epoch as the best result. Then, we can obtain a stability score \(\text{score}_i\) for each unlabeled image \(\text{ui}\):

\[
\text{score}_i = \frac{1}{K} \sum_{j=1}^{K-1} \phi(M_i, M_{iK}),
\]

where \(\text{score}_i\) is the stability score for each unlabeled image \(\text{ui}\):

\[
L_{\text{consist}} = \text{MSE}(y_S, y_T),
\]

where \(y_S\) and \(y_T\) are the soft predictions from \(S\) and \(T\), respectively. MSE is the mean square error loss [11]. So the overall objective of the plainest self-training is formalized as follows:

\[
L_{\text{plain}} = L_{\text{ce}} + \lambda_{\text{plain}} L_{\text{consist}},
\]

where \(\lambda_{\text{plain}}\) acts as a trade-off between labeled and unlabeled data.

**B. Progressive Self-Training Scheme**

Despite the above plainest self-training scheme can easily exploit unlabeled data, it treats each unlabeled sample equally without considering each independent instance’s inherent reliability and difficulty. Hence, incorrect predictions on hard examples will bring too many training noises, resulting in performance degradation. To tackle this issue, we propose a progressive self-training scheme by preferentially learning reliable data to reasonably explore the less reliable data from unlabeled samples in a curriculum learning manner [27].

The crux of our progressive self-training scheme is how to separate unlabeled data into two parts: reliable and unreliable images. Existing methods were proposed to estimate the model’s reliability or uncertainty mainly in a pixel-level manner, which can be divided into the following two types: (1) Adding dropout layers (i.e., MCDO [45], [46]) to one CNN-based model, and then retain them in the inference stage to estimate the model’s uncertainty by setting a fixed confidence threshold; (2) training two parallel networks with the same structure but different initialization, and then regard their disagreements [43] as uncertain regions. Unlike these above strategies, we wish to use only one model to estimate the reliability at the image-level without additional layers and confidence threshold. Moreover, the image-level manner allows us to learn more holistic contextual patterns.

1) Image-Level Sample Selection: We find a positive correlation between the shadow detection performance and the evolving stability of history predictions. Therefore, we can select better-predicted unlabeled images (reliable samples) based on their evolving stability in the training stage. Specifically, we save \(K\) checkpoints in the training stage and produce \(K\) corresponding pseudo masks for an unlabeled image. Furthermore, since deep networks tend to converge in the late training stage, we take the result of the last epoch as the best result. Then, we can obtain a stability score \(\text{score}_i\) for each unlabeled image \(\text{ui}\):

\[
\text{score}_i = \frac{1}{K} \sum_{j=1}^{K-1} \phi(M_i, M_{iK}),
\]

where \(\lambda_{\text{plain}}\) acts as a trade-off between labeled and unlabeled data.

**B. Progressive Self-Training Scheme**

Despite the above plainest self-training scheme can easily exploit unlabeled data, it treats each unlabeled sample equally without considering each independent instance’s inherent reliability and difficulty. Hence, incorrect predictions on hard examples will bring too many training noises, resulting in performance degradation. To tackle this issue, we propose a progressive self-training scheme by preferentially learning reliable data to reasonably explore the less reliable data from unlabeled samples in a curriculum learning manner [27].

The crux of our progressive self-training scheme is how to separate unlabeled data into two parts: reliable and unreliable images. Existing methods were proposed to estimate the model’s reliability or uncertainty mainly in a pixel-level manner, which can be divided into the following two types: (1) Adding dropout layers (i.e., MCDO [45], [46]) to one CNN-based model, and then retain them in the inference stage to estimate the model’s uncertainty by setting a fixed confidence threshold; (2) training two parallel networks with the same structure but different initialization, and then regard their disagreements [43] as uncertain regions. Unlike these above strategies, we wish to use only one model to estimate the reliability at the image-level without additional layers and confidence threshold. Moreover, the image-level manner allows us to learn more holistic contextual patterns.

1) Image-Level Sample Selection: We find a positive correlation between the shadow detection performance and the evolving stability of history predictions. Therefore, we can select better-predicted unlabeled images (reliable samples) based on their evolving stability in the training stage. Specifically, we save \(K\) checkpoints in the training stage and produce \(K\) corresponding pseudo masks for an unlabeled image. Furthermore, since deep networks tend to converge in the late training stage, we take the result of the last epoch as the best result. Then, we can obtain a stability score \(\text{score}_i\) for each unlabeled image \(\text{ui}\):

\[
\text{score}_i = \frac{1}{K} \sum_{j=1}^{K-1} \phi(M_i, M_{iK}),
\]

where \(\lambda_{\text{plain}}\) acts as a trade-off between labeled and unlabeled data.

**B. Progressive Self-Training Scheme**

Despite the above plainest self-training scheme can easily exploit unlabeled data, it treats each unlabeled sample equally without considering each independent instance’s inherent reliability and difficulty. Hence, incorrect predictions on hard examples will bring too many training noises, resulting in performance degradation. To tackle this issue, we propose a progressive self-training scheme by preferentially learning reliable data to reasonably explore the less reliable data from unlabeled samples in a curriculum learning manner [27].

The crux of our progressive self-training scheme is how to separate unlabeled data into two parts: reliable and unreliable images. Existing methods were proposed to estimate the model’s reliability or uncertainty mainly in a pixel-level manner, which can be divided into the following two types: (1) Adding dropout layers (i.e., MCDO [45], [46]) to one CNN-based model, and then retain them in the inference stage to estimate the model’s uncertainty by setting a fixed confidence threshold; (2) training two parallel networks with the same structure but different initialization, and then regard their disagreements [43] as uncertain regions. Unlike these above strategies, we wish to use only one model to estimate the reliability at the image-level without additional layers and confidence threshold. Moreover, the image-level manner allows us to learn more holistic contextual patterns.

1) Image-Level Sample Selection: We find a positive correlation between the shadow detection performance and the evolving stability of history predictions. Therefore, we can select better-predicted unlabeled images (reliable samples) based on their evolving stability in the training stage. Specifically, we save \(K\) checkpoints in the training stage and produce \(K\) corresponding pseudo masks for an unlabeled image. Furthermore, since deep networks tend to converge in the late training stage, we take the result of the last epoch as the best result. Then, we can obtain a stability score \(\text{score}_i\) for each unlabeled image \(\text{ui}\):

\[
\text{score}_i = \frac{1}{K} \sum_{j=1}^{K-1} \phi(M_i, M_{iK}),
\]
where \( \text{score}_i \in [0, 100] \) reflects the reliability of \( \mathbf{u}_i \). To avoid the model overfitting labeled data and then leading to poor results on the extra data, we use the early stopping strategy in the stage of sample selection. That means the value of \( K \) is slightly less than the number of conventional training epochs. Moreover, considering the class imbalance in shadow images, we take both shadow and non-shadow regions into account, and define the similarity between the earlier mask \( \mathbf{M}_{ij} \) and the final \( \mathbf{M}_{iK} \) as:

\[
\phi(\mathbf{M}_{ij}, \mathbf{M}_{iK}) = \frac{1}{2} \left( \frac{TP}{S} + \frac{TN}{NS} \right),
\]

where \( TP \) and \( TN \) are the numbers of true positives, and true negatives, respectively.

2) Progressive Self-Training in SDTR: In the first training, we select the top \( R \) samples with the highest stability score. Then, we use these selected images and corresponding pseudo masks together with labeled data to retrain a stronger student model. Finally, we use the model to predict the remaining unreliable images. At this time, all unlabeled data have their corresponding pseudo masks. So we can retrain the student model on all labeled and pseudo-labeled data in a fully supervised manner. The details of our SDTR are presented in Alg. 1.

### C. Learning From Weak Labels

Although our SDTR can holistically improve the quality of the pseudo masks, it still faces many challenges in some complex image regions, including tiny shadows, self shadows and some distraction regions. From these observations, we have the idea to provide some annotations for these samples as “key tips” to supervise the deep model.

1) Data Annotation: In theory, improving the generalization ability of a deep model often requires as many image scenes as possible. The unpaired shadow removal (USR) dataset [26] meets our demands, containing 2445 shadow images cast by various objects, e.g., trees, buildings, traffic signs, persons, umbrellas, railings, etc. Unfortunately, USR was originally designed for shadow removal without any binary shadow masks, so we opt to annotate the USR to exploit it better.

Dense labeling for shadow images is extremely difficult as it requires annotators to have more expertise, e.g., the edge detection for soft shadow or the judgment of self shadow. That is time-consuming and expensive, leading to an unexplored question: How can we generalize a new-coming dataset with a bare minimum of annotations? As we all know, the most common quick annotation approaches include scribble, box and point, known as “weak labels”. To achieve a better trade-off between annotation cost and detection performance, this work introduces a flexible manner (i.e., mixed annotation) to label these rich shadow scenes from the USR dataset.

#### Algorithm 1 Algorithm of SDTR and SDTR+

**Input:** \( D_{new} = \{\mathbf{u}, \emptyset \text{ or } \mathbf{y}_w\} \) : New-coming data

**Output:** a well-trained \( S \)

1: Train \( T \) on \( D_{label} \) and save \( K \) checkpoints \( \{T_j\}_{j=1}^K \)
2: for \( \mathbf{u}_i \) in \( D_{new} \) do
3: for \( T_j \) in \( \{T_j\}_{j=1}^K \) do
4: Produce pseudo masks \( \mathbf{M}_{ij} = T_j(\mathbf{u}_i) \)
5: end for
6: Compute \( \text{score}_i \) with Eq. 4 for \( \{\mathbf{M}_{ij}\}_{j=1}^K \)
7: end for
8: Select the top \( R \) unlabeled images into \( D_{new1} \)
9: \( D_{new2} = D_{new} - D_{new1} \)
10: \( D_{new1} = \{\mathbf{u}, T(\mathbf{u})\}_R \)
11: Train \( S \) on \( (D_t \cup D_{new1}) \) with Eq. 3
12: if Do not use any annotation on \( D_{new2} \) then
13: Print(“case of SDTR”)
14: Update pseudo mask \( D_{new2} = \{\mathbf{u}, S(\mathbf{u})\}_{1-R} \)
15: Retrain \( S \) on \( (D_t \cup D_{new1} \cup D_{new2}) \) with Eq. 3
16: else
17: Print(“case of SDTR+”)
18: Retrain \( S \) on \( (D_t \cup D_{new1}) \) with Eq. 3 and \( D_{new2} = \{\mathbf{u}, \mathbf{y}_w\}_{1-R} \) with Eq. 11
19: end if
20: return \( S \)

As shown in Fig. 4, we first divide these 2445 images into two parts: the training set with 2000 images and the testing set with 445 images. Then, we mixed-annotate the...
training set (Fig. 4a) and densely annotate the testing set (Fig. 4b). Moreover, we also provide a statistic for our Mix-USR, as shown in Fig. 4c. Noting that, labeling all data does not mean we must use them entirely in the training phase. The protocols used to annotate our Mix-USR can be summarized as follows:

- When there are only a few shadow regions in an image, we opt to provide a bounding box for them;
- When there are more shadow regions in an image, we opt to give a point at the center of each shadow region;
- When the scene on an image is complex, including shadow-like materials (e.g., black objects), shadow regions with non-shadow patterns (e.g., soft shadows), or ambiguous shadow regions (e.g., self shadows), we use several scribbles with two different colors to mark shadow and non-shadow regions, respectively.

2) Box Supervision: When a sample \( u \) from Mix-USR is annotated by boxes, we first calculate a corresponding minimal up-right bounding rectangle for each individual shadow region in the prediction. Then, to alleviate the effect of the noises in predictions, we filter out the rectangle whose width and height are less than 30 (the output is \( 416 \times 416 \)).

Formally, we denote the selected unreliable image from Mix-USR as \( u^{\text{unre}} \), the set of GT boxes as \( b \), and the set of predicted shadow regions as \( \hat{b} \). We cannot directly compute the training loss without the matching relationship between the predicted boxes and ground truths. Motivated by [29] and [33], we formulate this pseudo-label assignment problem as a bipartite matching problem, as shown in Fig. 5. For example, we assume the size of \( \hat{b} \) (i.e., \( |\hat{b}| = N \)) is larger than \(|b|\), and then consider \(|b|\) is equal to \( N \) by padding with \( \emptyset \) (None). We can find an optimal matching relationship \( \sigma \in \psi \) between these two sets with the Hungarian algorithm [29], [33] by minimizing the following bipartite matching loss:

\[
\hat{\sigma}_{\text{box}} = \arg \min_{\sigma \in \psi} \sum_{i=1}^{N} \mathcal{L}_{\text{iou}}(b_i, \hat{b}_{\sigma(i)}),
\]

where \( \mathcal{L}_{\text{iou}} \) is a generalized IoU loss [50], \( b \in [0, 1]^4 \) is a vector that defines the height and width of the ground truth box relative to the image size.

Then we compute the loss \( \mathcal{L}_{\text{box}} \) for all matched box pairs:

\[
\mathcal{L}_{\text{box}} = (|b| - |\hat{b}|)^2 + \sum_{i=1}^{N} \lambda_{\text{iou}} \mathcal{L}_{\text{iou}}(b_i, \hat{b}_{\sigma(i)}) + \lambda_{L1} ||b_i - \hat{b}_{\sigma(i)}||_1,
\]

where \( \lambda_{\text{iou}} \) and \( \lambda_{L1} \) are balance factors. The first part in \( \mathcal{L}_{\text{box}} \) constrains the number of predicted shadow regions, while the other one constrains the location and size.

3) Point Supervision: For point supervision, we also calculate the bounding rectangles for predicted shadow regions and then obtain their center coordinates \( p \in [0, 1]^2 \), as shown in Fig. 6. Similar to box supervision, the matching loss can be formulated as follows:

\[
\hat{\sigma}_{\text{point}} = \arg \min_{\sigma \in \psi} \sum_{i=1}^{N} ||p_i - \hat{p}_{\sigma(i)}||_1.
\]

Next, we introduce a point loss \( \mathcal{L}_{\text{point}} \) based on the matched point pairs as:

\[
\mathcal{L}_{\text{point}} = (|p| - |\hat{p}|)^2 + \sum_{i=1}^{N} \lambda_p ||p_i - \hat{p}_{\sigma(i)}||_1,
\]

where \( \lambda_p \) is a hyperparameter, we also pose the constraints on the number and location of predicted shadow regions.

4) Scribble Supervision: For scribble supervision, we only need to treat it similarly to any fully supervised task, as shown in Fig. 7. We design a scribble loss \( \mathcal{L}_{\text{scribble}} \) on the available points as:

\[
\mathcal{L}_{\text{scribble}} = -\frac{1}{|\mathcal{V}_s|} \sum_{i \in \mathcal{V}_s} \log(\hat{y}_i) - \frac{1}{|\mathcal{V}_{\text{ns}}|} \sum_{j \in \mathcal{V}_{\text{ns}}} \log(1 - \hat{y}_j),
\]

where \( \mathcal{V}_s \) and \( \mathcal{V}_{\text{ns}} \) denote the labeled shadow and non-shadow pixels, respectively. Although this supervision is sparse, it is effective as we empirically annotate error-prone regions.

5) Learning From Weak Labels in SDTR+: For the rest of shadow images, we directly supervise student’s outputs by \( \mathcal{L}_{\text{weak}} \):

\[
\mathcal{L}_{\text{weak}} = \alpha \mathcal{L}_{\text{box}} + \beta \mathcal{L}_{\text{point}} + \gamma \mathcal{L}_{\text{scribble}},
\]
where $\alpha$, $\beta$ and $\gamma$ are balance factors, which control the relative importance of the loss functions for each type of weak label. Note that the loss magnitudes of $L_{\text{box}}$ and $L_{\text{point}}$ are typically larger than that of $L_{\text{scribble}}$ due to the different computing strategies, we aim to reduce the differences among them by adjusting the values of these three hyperparameters. This adjustment prevents the model training from being adversely affected by excessively small losses of specific loss functions. Details of the hyperparameter values can be found in Sec. IV-A. The algorithm of our SDTR+ is also presented in Alg. 1.

IV. EXPERIMENTS

This section first describes shadow detection benchmarks, evaluation metrics and implementation details (Sec. IV-A). Then, we compare the proposed SDTR and SDTR+ with the state-of-the-art shadow detectors and some relevant image segmentation models, such as salience detection and semantic segmentation (Sec. IV-B). Next, we report ablation study results for network design, sample selection, weak label learning and different weak labels (Sec. IV-C). Finally, we apply our method for video shadow detection (Sec. IV-D). Our code, model parameters and datasets will be released at https://github.com/wuwen1994/SDTR.

A. Setup

1) Dataset: We validate our proposed method on three widely-used shadow benchmark datasets, i.e., SBU [16], UCF [13] and ISTD [51]. The SBU dataset contains 4089 training images and 638 testing images. The UCF dataset consists of 145 training images and 76 testing images, which are collected for traditional hand-crafted feature-based methods as its limited size. The ISTD dataset contains 1330 training images and 540 test images. Similar to a recent semi-supervised shadow detection method (i.e., MTMT-Net), we train our model on the SBU training set and our Mix-USR dataset, and test on SBU and UCF testing set. Since ISTD only contains cast shadow images that differ from SBU images, following [11, and [21], we retrain our method on the ISTD training dataset with Mix-USR, and test on the ISTD testing set.

2) Metric: We use a widely used metric, i.e., balance error rate (BER), to evaluate all competitors:

$$\text{BER} = \left(1 - \frac{1}{2} \left(\frac{TP}{S} + \frac{TN}{NS}\right)\right) \times 100.$$  (12)

3) Implementation Details: We use PyTorch 1.7.1 to implement the proposed network structure on a single NVIDIA RTX 3090 GPU. We fine-tune the parameters in the non-overlapping process [30] to produce features with resolution $\left[\frac{H}{K} \times \frac{W}{K}, \frac{H}{K} \times \frac{W}{K}, \frac{H}{K} \times \frac{W}{K}, \frac{H}{K} \times \frac{W}{K}\right]$. We use the Adam optimizer and cosine annealing scheduler with a learning rate of 1e-4 to optimize the whole network. In the first training phase, $K$ is empirically set as 20 and 15 for SBU and ISTD, respectively. While in the retraining phase, we train the whole network with 40 and 35 epochs on the basic dataset set as SBU and ISTD, respectively. All the images are resized to 416 $\times$ 416. The weak augmentation is set by random horizontal flipping, while the strong augmentation is set by color jitter and blur. The hyperparameters $[\lambda_{\text{plain}}, \lambda_{\text{iou}}, \lambda_{L1}, \lambda_p, \alpha, \beta, \gamma]$ are set as $[0.5, 2, 4, 5, 0.2, 0.3, 0.5]$. Following recent shadow detectors [11], [22], we also use fully connected CRF [56] to optimize our prediction results.

B. Comparing With State-of-the-Art Methods

We first compare our method with nine state-of-the-art shadow detectors, i.e., FDRNet [22], MTMT-Net [11], DSD [21], DC-DSPF [52], A+D Net [18], DSC [20], BDRAR [19], scGAN [17] and stacked-CNN [16]. Furthermore, since shadow detection is a pixel-level classification task, it is similar to saliency object detection and semantic segmentation. For a more comprehensive comparison, we also compare our method with two state-of-the-art saliency object detection methods, EGNet [55], ITSD [54], and three popular semantic segmentation models, SegFormer [30], SegNeXt [53] and MPViT [39].

1) Quantitative Comparison: Tab. I shows the quantitative results of the three datasets. One can observe that our SDTR and SDTR+ can achieve the best two BER scores among all competitors. Compared to the FDRNet, our SDTR reduces the BER scores by 2.30%, 6.46% and 1.94% on SBU, UCF and ISTD, respectively. Despite MTMT-Net and our SDTR being semi-supervised and using extra unlabeled data, the quantitative comparison between them shows the superiority of our easy-to-hard curriculum learning manner. In addition, when we provide some weak labels for partial unlabeled data, compared to our SDTR, our SDTR+ can further reduce the BER value by 0.66% and 0.64% on the SBU and ISTD dataset, respectively. Since the weak labels from our Mix-USR are mainly used to improve the generalization ability of SDTR+, the improvements from our SDTR to SDTR+ on the basic dataset (i.e., SBU and ISTD) are limited. It is worth noting that the performance on the UCF dataset can indicate the generalization ability of one shadow detector, as none of the above methods have been trained on this dataset. The 6.31% improvement (from 6.81 to 6.35) on the UCF dataset can demonstrate the effectiveness of our weak annotations.

2) Qualitative Comparison: As shown in Fig. 8, we also qualitatively compare our SDTR+ with the most recent state-of-the-art shadow detection methods. One can see that our SDTR+ has the best visualization performance among all competitors. It can efficiently detect tiny shadows (first row) and soft shadows (fifth row) with lower uncertainty. Moreover, in the second and fourth rows, SDTR+ can also accurately detect the shadow regions with crisp edges, while other methods produce many noises and zigzag boundaries. The robustness in various scenes is attributed to our simple yet efficient network designing and flexible learning strategies. In addition, the weak supervision of hard examples can provide valuable cues (i.e., number of shadows, locations and ambiguous regions) for generalizing other unseen scenes.

C. Ablation Study

The main contributions of this work lie in the following: (1) a simple yet efficient network, (2) a selective retraining
TABLE I

COMPARING OUR SDTR AND SDTR+ AGAINST THE STATE-OF-THE-ART SHADOW DETECTORS. THE BEST BER VALUES OF ALL METHODS ARE MARKED IN RED, WHILE THE BEST OF PREVIOUS METHODS IS MARKED IN BLUE. WE ALSO PROVIDE THE PERCENTAGE IMPROVEMENT XX.XX% FOR OUR TWO DETECTORS COMPARED TO THE BEST ONE IN THE PREVIOUS DETECTORS. WHERE “SHADOW” AND “NON-SHADOW” DENOTES THE ERROR RATES FOR THE SHADOW AND NON-SHADOW REGIONS, RESPECTIVELY.* PRESENTS A MODEL TRAINED WITH EXTRA DATA, ** MEANS A MODEL IS TRAINED WITH EXTRA SUPERVISION FROM OTHER MODELS.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>BER ↓ Shadow ↓</td>
<td>BER ↓ Non-shadow ↓</td>
<td>BER Shadow Non-shadow</td>
</tr>
<tr>
<td>SDTR+*(Ours)</td>
<td></td>
<td></td>
<td>3.15</td>
<td>2.75</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(2.96%↑)</td>
<td></td>
</tr>
<tr>
<td>SDTR*(Ours)</td>
<td></td>
<td></td>
<td>3.24</td>
<td>2.70</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(2.30%↑)</td>
<td></td>
</tr>
<tr>
<td>FDRCN [12]</td>
<td>2021</td>
<td>3.04</td>
<td>2.91</td>
<td>3.18</td>
</tr>
<tr>
<td>MTMT* [11]</td>
<td>2020</td>
<td>3.15</td>
<td>3.73</td>
<td>2.57</td>
</tr>
<tr>
<td>DSD** [21]</td>
<td>2019</td>
<td>3.45</td>
<td>3.33</td>
<td>3.58</td>
</tr>
<tr>
<td>DDC-DSPF [52]</td>
<td>2019</td>
<td>4.90</td>
<td>4.70</td>
<td>5.10</td>
</tr>
<tr>
<td>ADNet [18]</td>
<td>2018</td>
<td>5.37</td>
<td>4.45</td>
<td>6.30</td>
</tr>
<tr>
<td>DSC [20]</td>
<td>2018</td>
<td>5.59</td>
<td>9.76</td>
<td>1.42</td>
</tr>
<tr>
<td>BDRAR [19]</td>
<td>2018</td>
<td>3.64</td>
<td>3.40</td>
<td>3.89</td>
</tr>
<tr>
<td>stacked-CNN [16]</td>
<td>2016</td>
<td>0.00</td>
<td>8.84</td>
<td>12.76</td>
</tr>
</tbody>
</table>

| MPVT [39]     | 2022 | 3.18 | 3.05 | 3.31 | 7.01 | 8.99 | 5.03 | 1.77 | 1.26 | 2.28 |
| SegNoXt [53]  | 2022 | 3.29 | 4.12 | 2.46 | 6.98 | 7.13 | 6.83 | 1.75 | 1.42 | 2.08 |
| SegFormer [30] | 2020 | 4.36 | 5.70 | 3.02 | 7.74 | 9.70 | 5.78 | 1.91 | 1.73 | 2.09 |
| TSD [54]      | 2020 | 3.00 | 8.65 | 1.36 | 10.16 | 17.13 | 3.19 | 2.73 | 2.05 | 3.40 |
| BGNet [55]    | 2019 | 4.49 | 5.23 | 3.75 | 9.20 | 11.28 | 7.12 | 1.85 | 1.75 | 1.95 |

TABLE II

ABLATION STUDY ON THE COMPONENTS OF NETWORK ARCHITECTURE AND CORRESPONDING LOSS FUNCTION.

<table>
<thead>
<tr>
<th>Network</th>
<th>PPM</th>
<th>DS</th>
<th>BER SBU</th>
<th>BER ISTD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Basic</td>
<td>×</td>
<td>×</td>
<td>3.99</td>
<td>2.09</td>
</tr>
<tr>
<td>Basic+DS</td>
<td>×</td>
<td>✓</td>
<td>3.28</td>
<td>1.87</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(17.79%↑)</td>
<td>(10.52%↑)</td>
</tr>
<tr>
<td>Basic+PPM</td>
<td>✓</td>
<td>×</td>
<td>3.45</td>
<td>1.92</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(13.53%↑)</td>
<td>(8.13%↑)</td>
</tr>
<tr>
<td>Basic+PPM+DS</td>
<td>✓</td>
<td>✓</td>
<td>3.17</td>
<td>1.81</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>(20.55%↑)</td>
<td>(13.40%↑)</td>
</tr>
</tbody>
</table>

Fig. 8. Qualitative comparison of our SDTR+ with the most recent state-of-the-art methods.

scheme, and (3) a weak label learning strategy. In this section, we verify the actual effectiveness of the three parts in detail.

1) Effectiveness of Network Components: To validate the effectiveness of each element in the proposed network, we compare our detection part (Sec. III-A) with the following configurations trained on a single domain:

- basic: Removing the PPM module and deep supervision into a “basic” model;
- basic+DS: Appending deep supervision into “basic”;
- basic+PPM: Adding PPM to the “basic”.

One can observe that from Tab. II: (1) These three detectors can all achieve state-of-the-art results on two benchmarks, which is mainly attributed to the recent progress in Transformer architecture; (2) “basic+DS” have superior BER values over “basic”, indicating the deep supervision scheme can guide the deep model to extract more “task-relevant” features in the early training phase. It improves the BER value from 3.99 to 3.28 and 2.09 to 1.87 on the SBU and ISTD, respectively; (3) “basic+PPM” also has better performance than “basic”,

Authorized licensed use limited to: HANGZHOU DIANZI UNIVERSITY. Downloaded on October 28, 2023 at 07:47:38 UTC from IEEE Xplore. Restrictions apply.
TABLE III

PERFORMANCE OF SDTR WITH DIFFERENT SETTINGS ON THE PROPORTION $R$, WHERE “SBU + USR → MIX-USR” DENOTE THAT TRAINING ON SBU AND USR, THEN TESTING ON THE MIX-USR TESTING SET

<table>
<thead>
<tr>
<th>$R$ for SDTR</th>
<th>10%</th>
<th>20%</th>
<th>50%</th>
<th>70%</th>
<th>90%</th>
</tr>
</thead>
<tbody>
<tr>
<td>SBU + USR → Mix-USR</td>
<td>5.58</td>
<td>5.36</td>
<td>5.13</td>
<td>5.01</td>
<td>5.03</td>
</tr>
<tr>
<td>ISTD + USR → Mix-USR</td>
<td>8.57</td>
<td>8.14</td>
<td>8.15</td>
<td>8.34</td>
<td>8.55</td>
</tr>
</tbody>
</table>

Fig. 9. Quality of pseudo masks with and without selective retraining.

which demonstrates that the effective global contextual priors extracted from PPM are useful for our shadow detection; (4) by combining these two parts into “basic”, we can achieve 20.55% and 13.40% BER value improvement than “basic” on SBU and ISTD, respectively.

2) Effectiveness of Selective Retraining Scheme: To verify the effectiveness of our selective retraining scheme in the SDTR, we first conduct ablation studies on the value of $R$ (the ratio of reliable samples in USR). For training a student model in SDTR, the value of $R$ means that the $R$ images in USR are supervised directly by the old teacher model, and the rest $1-R$ images are supervised by the retrained teacher. Specifically, we evaluate the performance of the well-trained student from SDTR on the testing set of Mix-USR. From Tab. III, one can observe that the performance degrades gradually when $R$ exceeds 70% and 20% for the labeled dataset is set to SBU and ISTD, respectively.

2a) Fig. 9. Quality of pseudo masks with and without selective retraining.

3) Effectiveness of Weak Label Learning: To verify the effectiveness of weak labels, we first calculate the BER value using the $R$ well-defined in Sec. IV-C.2. For training a student model, the value of $R$ means the $R$ images in USR are supervised by the old teacher model, and the rest $1-R$ are supervised by the weak labels from Mix-USR. As shown in Tab. IV, one can observe that, compared with SDTR (Tab. III), SDTR+ can improve the BER value from 5.01 to 4.80 and 8.14 to 7.64, respectively.

Moreover, we also conduct ablation studies with different $R$ to explore how many annotations can help generalize USR best. One can observe that: (1) The value of $R$ for obtaining the best BER value is equal to SDTR; (2) we cannot get better results when using more labels as these weak supervisions may not be suitable for reliable samples compared to the pseudo masks generated from the teacher model.

We also provide some visualization results to show the superiority of our SDTR+, as shown in Fig. 10. Compared with baseline and SDTR, SDTR+ can significantly improve the quality of the pseudo mask in the hard case.

4) Effectiveness of Different Types of Annotations: To demonstrate the superiority of our mixed annotation, we also annotate the USR dataset with three types of single annotation (i.e., scribble, box and point). Then, we use the optimal setting $R = 70\%$ and $R = 20\%$ from Sec. IV-C.2 and further conduct some comparisons in annotation cost and detection performance for the total four annotation strategies in Tab. V.
In principle, scribble annotation does not require us to find and label all the shadow regions and leads to the highest efficiency, while box annotation will result in a booming annotation cost when there is an increase in the number of shadow regions. It is worth noting that the mixed annotation tends to help us to annotate images faster, although it requires us to take extra time to judge which annotation type is more suitable for the current image. One can also observe that: (1) The burden of the mixture annotation (5.8s) is larger than scribble (3.5s) and point annotation (4.2s) but lower than box annotation (8.3s); (2) compared to other annotations, the mixed manner can lead to the highest BER values for SDTR+; (3) point annotation cannot lead to a satisfactory performance due to the inappropriate usage in some shadow scenes will produce severe label noises, potentially hampering the detection performance of deep models. For example, as shown in Fig. 11, point annotation is not suitable for shadows with winding shape (Fig. 11a): one of the calculated points (Fig. 11b) is far away from its corresponding GT point, bringing a big training loss. Therefore, we should opt to label this image via scribble or box.

In summary, the advantages of mixed annotation lie in the following two aspects: (1) This manner tends to find a better assignment between the shadow image and weak label, allowing SDTR+ to pay attention to the distraction region in the image, as well as the quantity, size and location of the complex shadow regions; (2) it can bring fewer label noises than the single annotation. To this end, mixed annotation can make SDTR+ achieve a better trade-off between the annotation cost and generalization ability.

### D. Applications

#### 1) Video Shadow Detection

Thanks to our simple yet efficient designs on network architecture, our SDTR+ can produce a shadow mask in real-time. To this end, we apply it to handle shadow videos from ViSha dataset [57] by processing each frame in order. One can observe from Fig. 12, our SDTR+ can also work well on some videos despite not training on any video data.

![Video 1](image1.png) ![Video 2](image2.png)

Fig. 12. Application of our SDTR+ on video shadow detection.

V. CONCLUSION

This paper introduces a robust semi-supervised shadow detector SDTR by designing a simple yet efficient architecture and an image-level reliable sample selection scheme. Moreover, we also propose SDTR+, a unified framework for weakly supervised shadow detection, which can adopt different types of weak annotations for unreliable samples. Under this guidance, we can achieve a better trade-off between annotation cost and performance improvement. To this end, for new-coming shadow images, one can first identify unreliable samples via our reliable sample selection scheme and then choose to (1) retrain the reliable samples, re-inference for unreliable samples, and obtain relatively accurate pseudo masks; or (2) annotate unreliable samples in a flexible way (i.e., box, point and scribble), and then learn from them to further improve the generalization ability of deep models. In the future, we will explore how to generalize new-coming data using fewer annotations.

### REFERENCES


Wen Wu received the B.Sc. degree in electronic and information engineering from Wuhan College of Arts and Science in 2016 and the M.Sc. degree in computer applied technology from Hubei University in 2019. He is currently pursuing the Ph.D. degree in computer science and technology with Hangzhou Dianzi University. From 2019 to 2022, he was a Lecturer in computer science with the Xinjiang Institute of Technology. His research interests include deep learning and computer vision.

Wenya Yang received the B.Sc. degree from the Zhengzhou University of Aeronautics in 2017 and the M.Sc. degree from the Zhengzhou University of Light Industry in 2020. She is currently pursuing the Ph.D. degree with the School of Computer Science, Hangzhou Dianzi University. Her research interests include computer vision and image processing.

Weiyin Ma (Member, IEEE) is currently a Faculty Member with the Department of Mechanical Engineering, City University of Hong Kong (CityU), lecturing in areas of geometric design, CAD/CAM, finite element analysis, and rapid prototyping technologies. Before joining CityU in 1995, he was a Research Fellow with Materialise N.V., a rapid prototyping firm in Belgium, where he developed software modules on medical CAD modeling from computed tomography (CT) data. His current research interests include computer aided geometric design, CAD, 3D printing, and rapid manufacturing.

Xiao-Diao Chen received the bachelor’s degree from Zhejiang University and the master’s and Ph.D. degrees from Tsinghua University. He is currently a Faculty Member of the School of Computer, Hangzhou Dianzi University. His research interests include approximation and interpolation methods applied in computer graphics, edge detection, and shadow detection in image processing.