Semantic-aware Transformer for shadow detection
Kai Zhou, Jing-Long Fang, Wen Wu, Yan-Li Shao, Xing-Qi Wang, Dan Wei

A R T I C L E I N F O
Communicated by Alessio Del Bue
MSC:
41A05
41A10
65D05
65D17
Keywords:
Image processing
Shadow detection
Transformer
Semantic-aware
Multi-task learning

A B S T R A C T
Shadow detection is significant for scene understanding. Ambiguities in a shadow image, such as shadow-like non-shadow regions and shadow regions with non-shadow patterns, are still very challenging for prevalent CNN-based methods. This work attempts to alleviate this problem from a new perspective of shape semantics, and then proposes a Semantic-aware Transformer (SaT) in a multi-task learning manner. Concretely, we first propose a shadow detection network based on the recent progress of Transformer architecture, allowing us to capture significant global interactions between contexts. Next, we design a multi-task learning framework, combining shadow supervision and semantic supervision to perform a semantic-aware shadow detection. Finally, we introduce a simple yet effective information buffer unit to overcome the gradient signal conflict from multi-task learning. Experimental results on three public benchmark datasets (i.e., ISTD, SBU, and UCF) show that our SaT can effectively detect ambiguous cases and achieve state-of-the-art results.

1. Introduction
Shadows are common in real-world scenes, caused by occluders blocking light sources. In some vision tasks, the presence of shadows will lead a performance degradation, so we are urgent to locate and then remove them. For example, detecting and removing shadows from text images (Wang and Chuang, 2020) and remote sensing images (Silva et al., 2018) can enhance the readability and identifiability, respectively. In some other tasks, shadows are prone to bring ambiguity and are often misrecognized as objects in image segmentation (Eiens et al., 2014), object detection (Cucchiara et al., 2003), and visual tracking (Nadimi and Bhanu, 2004). In addition, shadows can provide valuable cues for scene understanding, i.e., light direction (Lalonde et al., 2012), object geometry (Karch et al., 2011), and camera parameters (Wu et al., 2010). Hence, accurate shadow detection is vital to its subsequent tasks.

Traditional methods are mainly proposed to detect shadows based on hand-crafted features, e.g., illumination (Zhang et al., 2015), color (Salvador et al., 2004), and others (Lalonde et al., 2010; Zhu et al., 2010). In recent years, convolutional neural networks (CNNs) have been successfully applied in various vision tasks (Tian et al., 2021; Zhou et al., 2022; Mozdehi and Medeiros, 2022) as their powerful feature representation capabilities. Currently, CNN-based methods (Hu et al., 2018; Le et al., 2018; Zhu et al., 2018; Zheng et al., 2019; Chen et al., 2020; Hu et al., 2021) have become the mainstream for shadow detection, achieving impressive performance. They generally employ two strategies: combining contextual information (Hu et al., 2018; Zhu et al., 2018; Zheng et al., 2019), and enlarging training data (Le et al., 2018; Chen et al., 2020). These methods are mainly trained on the ISTD (Wang et al., 2018b) and SBU (Vicente et al., 2016) datasets. We find that the misdetected samples are mostly ambiguous cases: (1) Shadow-like regions have similar colors to shadows (i.e., shadow-like non-shadow regions), which are often misrecognized as shadows; (2) some heterogeneous backgrounds exist in the shadow regions, forming relatively bright regions (i.e., shadow regions with non-shadow patterns), weakening the shadow color and makes the result incomplete.

To tackle these issues, some works (Chen et al., 2020; Hu et al., 2021) employ extra training data to improve the model’s performance. However, these models still struggle with ambiguous cases. There may be two reasons for these ambiguities: (1) The essence of shadow detection is to perform binary classification of pixels. Ground truth (GT) is only presented in the form of a shadow mask, which is insufficient to adapt to ambiguous scenes as the absence of more prior knowledge (e.g., occluder category); (2) due to the lack of semantic interaction in spatial information extracted by convolution operations,
CNN-based methods have limitations in modeling long-range dependencies. Hence, these methods usually show weak performance when significant variations in the target regions’ shape, size, and texture.

We observe that humans can accurately identify shadow regions in natural scenes as we typically leverage global-to-local prior knowledge of shadows. Specifically, we first capture the regions that may be shadows (including shadow-like regions) in the global image according to the color prior, then further select/complete shadow regions (local) according to the shadow shape prior (e.g., person, animal, or object), thereby accurately locating the shadow position. Therefore, we consider incorporating shadow shape semantics (i.e., occluder category information) in supervision to assist shadow detection. For this purpose, we first append extra semantic masks to the original GTs to obtain shadow semantic labels, and then employ semantic priors to identify ambiguous regions, as shown in Fig. 1. This strategy can be viewed as enhancing shadow detection via shadow semantic segmentation. Note that the proposed shadow semantic segmentation aims to segment shadow regions with different categories, which is different from image semantic segmentation, that is, to segment all objects in the image.

To overcome the limitations of CNN, some works (Zheng et al., 2019; Wang et al., 2018a) initially propose to introduce a self-attention mechanism in CNN to enhance the global information interaction ability. Next, Transformer (Vaswani et al., 2017) is proposed as alternative architecture, since it completely relies on the self-attention mechanism, enabling modeling in the global context. Recently, Swin Transformer (Liu et al., 2021) is proposed to solve the problems of scale and computational complexity existing in vision Transformer (ViT) (Dosovitskiy et al., 2020), which performs local self-attention calculation by shifting window scheme. Compared with traditional Transformer, it can process high-resolution images efficiently and model flexibly on multiple scales, making it more suitable for dense prediction tasks. In particular, the recently proposed Swin-Unet (Cao et al., 2022) based on the Swin Transformer has been applied to medical image segmentation, and achieved a significant performance improvement. Inspired by Swin-Unet, we use Swin Transformer as the backbone of the shadow detector to process more complex shadow images.

Based on the above analysis, we propose a Semantic-aware Transformer (SaT) for shadow detection. Specifically, we first transform the training task into two supervised tasks, i.e., shadow supervision and semantic supervision. Then, we build a multi-task learning framework based on Swin Transformer and deep supervision (Lee et al., 2015). Since the low-level features contain rich shadow details, we use category-agnostic shadow supervision, while the high-level features contain rich shadow semantics, we use category-aware shadow semantic supervision. However, this different supervision manner will cause gradient signal conflict during training, affecting the network’s performance. To this end, we design an information buffer unit to cope with signal conflict. Extensive experiments demonstrate that the proposed method outperforms state-of-the-art methods, and improves the BER values by 11.05%, 4.13%, and 3.88% on ISTD, SBU, and UCF datasets, respectively.

Our main contributions are as follows:

- We are the first to propose a shadow detection method using shape semantics. With the help of shadow semantic information, our model can effectively avoid the interference of ambiguous regions. To this end, we also create two semantic label sets (i.e., Sem-ISTD and Sem-SBU) for semantic learning.
- We develop an efficient detection network with 24.37M parameters based on Swin Transformer, which fuses shadow multi-scale predictions to make the shadow detection results more complete and fine-grained.
- To deal with ambiguous cases better, we build a multi-task learning framework combining shadow supervision and semantic supervision, and design a simple yet effective information buffer unit to coordinate different supervised tasks.

## 2. Related works

### 2.1. Shadow detection

Shadow detection can help in visual scene understanding. Existing works are mainly divided into two categories, i.e., traditional and deep learning-based methods. The former detects shadows by building physical models and machine learning models. They rely on various hand-craft features such as geometrical properties (Salvador et al., 2004; Panagopoulos et al., 2011), illumination (Zhang et al., 2015; Tian et al., 2016), color (Salvador et al., 2004; Finlayson et al., 2005), edge (Lalonde et al., 2010; Huang et al., 2011), and texture (Zhu et al., 2010; Guo et al., 2011; Vicente et al., 2017). However, these methods usually suffer from performance degradation in real-world scenes as the hand-crafted features are not discriminative enough.

In recent years, deep learning has achieved excellent performance in shadow detection. For example, Khan et al. (2014) use CNNs to learn shadow features automatically, and then leverage conditional random field to generate shadow contours. Vicente et al. (2016) propose a stacked-CNN framework to learn shadow detection from a collection of noisy annotations. Nguyen et al. (2017) present a novel GAN network with a tunable sensitivity parameter based on the conditional generative adversarial network (cGAN). Wang et al. (2018b) design a stacked cGAN that can learn shadow detection and removal simultaneously. Hu et al. (2018) embed a direction-aware spatial context module in each CNN layer for learning global semantics. Le et al. (2018) jointly train the shadow attenuation network and the shadow detection network to form adversarial learning. Zhu et al. (2018) use the recurrent attention residual module to extract CNN’s global and local context information to aggregate shadow context features. Zheng et al. (2019) introduce a distraction-aware shadow module to correct false positive and false negative regions in shadow detection.

Recently, Wang et al. (2020) propose instance shadow detection, which aims to pair object instances and shadow instances. It can accurately locate shadows, but it is difficult to generalize to shadow images without object instances. Chen et al. (2020) propose a multi-task mean teacher network (MTMT-Net), which combines the three detection tasks of shadow region, shadow edge, and shadow count to significantly improve inference accuracy. Hu et al. (2021) propose a fast shadow detection network (FSDNet) with only 4.4M parameters, which greatly improves the detection efficiency. Recent shadow detection methods mainly focus on inference accuracy and computational efficiency, as their subsequent tasks (e.g., shadow removal and shadow editing (Li et al., 2022)) are highly dependent on shadow detection. However, these methods struggle to work when the shadow color is similar to the background (i.e., ambiguous scenes). Shadows are ubiquitous in various visual scenes, so it is necessary to develop a robust shadow detector for ambiguous cases.
Recently, Swin-Unet (Cao et al., 2022) based on the Swin Transformer overcomes the above problems, as it enables global semantic information interaction and long-range dependency modeling by the shifted window scheme. Following the Swin-Unet, we use Swin Transformer Blocks (STBs) as the backbone network. Shadow detection should be efficient as the dependencies of its downstream tasks. However, Swin-Unet obeys the U-shaped design, so the main computational cost is from the decoder caused by skip connections. To this end, we only retain the encoder of Swin-Unet and directly compress and upsample multi-scale feature maps to obtain the final shadow detection result, as shown in Fig. 3. Our detection network mainly consists of an encoder and a decoder, which enables efficient shadow detection.

3.1. Network architecture

Encoder. In the encoder, the inputs are fed into the STBs to perform representation learning. Given a shadow image \( I \in \mathbb{R}^{H \times W \times 3} \) as input, the image is first split into many non-overlapping patches (tokens) by the patch partition layer. We set the patch size as \( 2 \times 2 \times 2 \), and its corresponding feature dimension is \( 2 \times 2 \times 3 = 12 \). By doing this, the input is transformed into sequence embeddings. Then, the feature dimension is transformed by the linear embedding layer, from 12 to \( C \), where \( C \) represents any dimension. The patch tokens pass through four stages (Stage1–Stage4) to generate multi-scale feature maps. In each stage, the task of the STB is feature transformation, and the task of the patch merging layer is downsampling.

Hierarchical feature maps constructed by Swin Transformer (Liu et al., 2021) can help shadow detection task perform dense prediction at pixel level. The key to its success lies in the Multi-head Self Attention (MSA) design based on the shifted window. As shown in Fig. 4, a Window-based MSA (W-MSA) is used in the first STB, and a Shifted Window-based MSA (SW-MSA) is used in the second STB. In both blocks, other layers are the same as the traditional Transformer, including LayerNorm (LN) layer and Multi-Layer Perceptron (MLP) layer. In our encoder, each stage uses two successive Swin Transformer Blocks (STB×2).

Decoder. To improve detection efficiency, instead of designing the decoder like Swin-Unet (Cao et al., 2022), we build our decoder using the side outputs of the encoder. To be specific, we first connect an information buffer (IB) unit after each side (side1–side4) to output multi-scale feature maps, and then compress and upsample \((x2, 4, 8, \text{and} 16)\) them to the same resolution for concatenation and output. Our decoder differs from that of Swin-Unet in three main aspects: (1) Swin-Unet is a symmetric U-shaped architecture with a heavy decoder. Our decoder consists mainly of four simple IBs, which significantly reduces the model’s parameters, resulting in lower computational costs. (2) Swin-Unet produces only one prediction at the last layer of its encoder, while our decoder can produce multi-scale predictions for fusion by deep supervision, which makes our prediction results more detailed and complete than Swin-Unet’s. (3) Furthermore, we combine shadow supervision and semantic supervision in the decoder to build a multi-task learning framework (Section 3.3), enabling our network to handle ambiguous cases better.

3.1.2. Network parameter analysis

Efficient shadow detection is beneficial for its subsequent tasks, such as shadow removal and shadow editing. Hence, we must consider the goal of effectively reducing the model’s parameters when designing a shadow detection network. Traditional Transformer (Vaswani et al., 2017) has been widely used in computer vision, but its processing of visual tasks is based on global self-attention computing. The global computational complexity is quadratic of the image size, which is unsuitable for the dense prediction task in shadow detection. However, Swin Transformer (Liu et al., 2021) optimizes this calculation. Its self-attention calculation is implemented in non-overlapping local windows, and the computational complexity has a linear relationship with the image size. Given an image with \( h \times w \) patches, assuming that each
window is set to $M\times M$ patches, the computational complexity of MSA and W-MSA can be expressed as:

$$\Omega(\text{MSA}) = 4hwC^2 + 2(hw)^2 C, \tag{1}$$

$$\Omega(\text{W-MSA}) = 4hwC^2 + 2M^2 hwC, \tag{2}$$

where $\Omega(\text{MSA})$ is quadratic to $hw$, and $\Omega(\text{W-MSA})$ and $hw$ are linear. One can see that Swin Transformer has higher computational efficiency. In the experiment, we design our network based on Swin Transformer with only 24.37M parameters, which can perform the shadow detection task efficiently.

### 3.2. Semantic annotation for GTs

Existing GTs (i.e., shadow masks) (Wang et al., 2018b; Vicente et al., 2016) only present shadow and non-shadow regions, and it is difficult for the trained model to identify the ambiguous regions further. To enable the model to learn shadow knowledge sufficiently, we perform semantic annotation on the original GTs to help shadow detection. In detail, we add semantic masks to the GTs in the ISTD (Wang et al., 2018b) and SBU (Vicente et al., 2016) dataset to obtain our semantic label sets, named Sem-ISTD and Sem-SBU.

Fig. 5 shows our semantic annotation details. We first divide shadows into several categories according to occluders, and then use different colors to represent these shadow categories. The protocols used to annotate GTs can be summarized as follows:

- Each image may contain one or more categories of shadows, and when there are connected categories, we need to delineate category boundaries according to occluder priors.
- For categories with the same shape, such as rectangular boards of different sizes in ISTD, we group them into the same category.
- In SBU, we group the shadows with similar occluders into the same category, such as motorcycles and bicycles are classified as "cycle."

To further illustrate our semantic label sets, we perform statistics and analysis on them in Fig. 6. (a) and (b) show the ratio distribution for each category. The ratio represents the proportion of the number of images containing the same category to that of the entire dataset. (c) and (d) show the mutual dependencies among shadow categories. One can observe that: (1) Sem-SBU has more categories than Sem-ISTD, and (2) Sem-SBU has more complex category dependencies than Sem-ISTD. Therefore, the semantic learning of Sem-SBU is more challenging than Sem-ISTD.

### 3.3. Multi-task learning

According to previous methods (Hu et al., 2018; Zhu et al., 2018; Zheng et al., 2019; Chen et al., 2020; Hu et al., 2021), high-level features contain abstract semantic information, while low-level features contain rich details but less semantic information. These methods usually use high-level semantics to predict shadows without recognizing shadow categories, and utilize low-level features to supplement details. They tend to misdetect ambiguous regions as their models treat all detection cases equally.

To overcome the interference of ambiguity, we further improve the above strategy: (1) adding a semantic supervision task to shadow detection (i.e., shadow category recognition), and (2) using deep supervision to improve low-level and high-level predictions. Instead of directly performing semantic supervision to each layer (i.e., the original deep supervision manner (Lee et al., 2015)), we only employ this scheme for the top layer as low-level features contain rich details but less semantic information. These methods usually use high-level semantics to predict shadows without recognizing shadow categories, and utilize low-level features to supplement details. They tend to misdetect ambiguous regions as their models treat all detection cases equally.
yet effective information buffer (IB) unit to ensure consistent gradient signals. Considering that residual network (He et al., 2016) is easier to optimize than ordinary network, we use two residual convolution modules to form the IB, as shown in Fig. 7. The IB can transform the representation of the side output into a more proper form. In this way, the gradients generated by the side supervision can be avoided from being directly propagated into the backbone network. So the backbone network can generate a unified update signal and optimize for the same goal. In general, adding a few extra convolutional layers will introduce more network parameters and does not necessarily improve network performance. Although IB also makes the network deeper, it is used to mitigate the gradient signal conflict caused by different supervision (multi-task learning), which can effectively improve the network performance.

**Multi-task learning framework.** For shadow supervision, we use a single-channel $1 \times 1$ conv at side1~side3 to generate shadow maps $S = \{S^1, S^2, S^3\}$. For semantic supervision, we use a $K$-channel $1 \times 1$ conv at side4 to generate semantic shadow maps $A^k = \{A^k_1, A^k_2, \ldots, A^k_4\}$, where $K$ represents the number of shadow categories. Fig. 8 shows how the proposed framework concatenates and fuses these predicted maps. We first copy $S$ and each channel of $A^k$ for shared concatenation (SC) to get the stacked shadow activation map:

$$S^f = \{S, A^1_1, S, A^2_1, \ldots, S, A^K_1\}.$$  \hspace{1cm} (3)

and then use $K$ $1 \times 1$ convs to fuse $S^f$ into a semantic shadow map with $K$ channels, which is finally binarized into a shadow mask.

3.4. **Loss function**

To achieve the multi-task learning, we compute the multi-task supervision loss by fusing the shadow supervision loss and the semantic supervision loss. Let $W$ denotes the parameters of all network layers in SaT. For a shadow image $I$, suppose its shadow label (i.e., GT) is $Y = \{y_i : i = 1, 2, \ldots, |I|\}$, and its semantic label is $\{C^1, C^2, \ldots, C^K\}$. Where $C^k = \{C^k_i : i = 1, 2, \ldots, |I|\}$ is the shadow map of the $k$th category. We design shadow supervision loss for side1~side3 based on cross-entropy (Zheng et al., 2019), which can be expressed as:

$$L^n_{sid}(W) = \sum_{i \in I} [−y_i \cdot \log(P(S^n_i; W))]$$

$$-(1−y_i) \cdot \log(1−P(S^n_i; W))),$$

where $m = 1, 2, 3$ represents the sequence number of the side, $S_i^n$ represents the activation function value at pixel $i$, and $P_i$ denotes the activation function (i.e., Sigmoid).

Further, we design semantic supervision loss for side4:

$$L^k_{sid}(W) = \sum_{i \in I} [−C^k_i \cdot \log(P(A^k_i; W))$$

$$-(1−C^k_i) \cdot \log(1−P(A^k_i; W))),$$

where $A^k_{i,j}$ represents the activation function value at pixel $i$ and belongs to category $k$.

Similarly, we can get the semantic supervision loss of the fused semantic shadow map:

$$L_{fus}(W) = \sum_{k \in I} [-C^k_i \cdot \log(P(S_f^k; W))]$$

$$-(1−C^k_i) \cdot \log(1−P(S_f^k; W))].$$

5
BER = error rate (BER), which is defined as:

\[ \text{BER} = \left( \frac{TP}{P} + \frac{TN}{N} \right) \times 100 \]  

where \( TP, TN, P \) and \( N \) are the number of true positives, true negatives, shadow, and non-shadow pixels, respectively. "S" and "NS" denote the pixel error rate for shadow and non-shadow, respectively. * indicates that the model is trained with extra data. The best performance is set to bold red, and the second one is set to bold blue.

### Table 1

Performance comparison of different methods. Where "FPS" represents "frames per second," "Para" represents the number of the model's parameters, and "SaT\textbackslash SeSup" denotes SaT without semantic supervision. "S" and "NS" denote the pixel error rate for shadow and non-shadow, respectively. * indicates that the model is trained with extra data. The best performance is set to bold red, and the second one is set to bold blue.

<table>
<thead>
<tr>
<th>Method</th>
<th>Year</th>
<th>FPS</th>
<th>Para(M)</th>
<th>ISTD BER ↓</th>
<th>SBU BER ↓</th>
<th>UCF BER ↓</th>
</tr>
</thead>
<tbody>
<tr>
<td>SeGAN</td>
<td>2017</td>
<td>64.06</td>
<td>28.24</td>
<td>4.70</td>
<td>3.22</td>
<td>6.18</td>
</tr>
<tr>
<td>A+D Net</td>
<td>2018</td>
<td>8.91</td>
<td>79.03</td>
<td>3.42</td>
<td>3.85</td>
<td>3.00</td>
</tr>
<tr>
<td>BDRAR</td>
<td>2019</td>
<td>39.53</td>
<td>58.16</td>
<td>2.17</td>
<td>1.36</td>
<td>2.98</td>
</tr>
<tr>
<td>MTMT-Net*</td>
<td>2020</td>
<td>21.88</td>
<td>89.62</td>
<td>1.72</td>
<td>1.36</td>
<td>2.08</td>
</tr>
<tr>
<td>DSDNet</td>
<td>2021</td>
<td>114.62</td>
<td>4.40</td>
<td>2.66</td>
<td>2.08</td>
<td>3.24</td>
</tr>
<tr>
<td>Swin-Unet</td>
<td>2022</td>
<td>37.09</td>
<td>59.74</td>
<td>2.23</td>
<td>2.14</td>
<td>2.32</td>
</tr>
<tr>
<td>SaT\textbackslash SeSup (ours)</td>
<td>–</td>
<td>87.54</td>
<td>21.16</td>
<td>1.82</td>
<td>1.53</td>
<td>2.11</td>
</tr>
<tr>
<td>SaT (ours)</td>
<td>–</td>
<td>76.23</td>
<td>24.37</td>
<td>1.53</td>
<td>1.24</td>
<td>1.82</td>
</tr>
</tbody>
</table>

Table 1 reports the quantitative comparison results of these methods. One can observe that our SaT obtains the best detection performance across the three datasets. Although Swin-Unet is also based on the light-weight Swin Transformer, its model’s parameters reach 59,74M due to its U-shaped design. Unlike Swin-Unet, our SaT decoder is cleverly designed as a multi-task learning framework, which not only greatly reduces the model’s parameters (from 59.74M to 24.37M), but also achieves a significant performance improvement. Both MTMT-Net and our SaT improve performance by multi-task learning. Compared with MTMT-Net, our SaT combines semantic supervision task reduces the BER scores by 11.05%, 4.13%, and 3.88% on ISTD, SBU, and UCF, respectively. Our SaT\textbackslash SeSup achieves comparable performance to

**Implementation details.** The proposed SaT is achieved based on PyTorch 1.7.0 and Python 3.6. We train our network on a GeForce RTX 3090 GPU with 24 GB memory. During training, the input image size is set to 256 × 256, the patch size is set to 4, and data augmentation is performed by random horizontal flipping, colorjitter, and blur to increase the data diversity. We use stochastic gradient descent (SGD) to optimize all parameters of the network, the batch size is set to 16, the learning rate is set to 0.001, and the momentum and weight decay are set to 0.9 and 1e-4, respectively. The training epochs on the ISTD and SBU datasets are set to 40 and 60, respectively.

### 4.2. Comparisons with state-of-the-art methods

We compare proposed SaT with eight state-of-the-art methods, i.e., SeGAN (Nguyen et al., 2017), DSC (Hu et al., 2018), A+D Net (Le et al., 2018), BDRAR (Zhu et al., 2018), DSDNet (Zheng et al., 2019), MTMT-Net (Chen et al., 2020), FSDNet (Hu et al., 2021), and a similar task Swin-Unet (Cao et al., 2022). We analyze all methods on three benchmark datasets, i.e., ISTD, SBU, and UCF.
MTMT-Net by deeply supervised Swin Transformer and fusing multi-scale predictions. FSDNet has the least model parameters among all methods, but at the expense of inference accuracy. Although our SaT has more parameters than FSDNet, it also can achieve efficient shadow detection with a 76.23 FPS. In addition, the gradual improvements of SaT\SeSup and SaT on the UCF dataset demonstrate that our models can generalize well to unseen scenes, benefiting from the robust shadow detection network.

Fig. 9 further shows some visual comparisons. One can see that our SaT’s predictions are the closest to GTs. For ambiguous scenes, none of the previous methods can overcome the ambiguous regions’ interference, i.e., shadow-like regions (e.g., the 2nd and 3rd rows) and heterogeneous background regions (e.g., the 1st, 3rd, and 5th rows). Although MTMT-Net can filter most shadow-like regions using its multi-task detection strategy, some shadow-like materials connected to the shadow regions still cannot be identified, such as in the 2nd and 3rd rows. Compared to MTMT-Net, our method overcomes this issue well with shadow shape semantics. DSDNet is a CNN-based network designed for ambiguous cases. Still, this approach performs poorly when the shadow color is similar to the background, such as in the 3rd, 4th, and 5th rows, because it is difficult for CNN to capture global and long-range information interactions. Compared to DSDNet, our shadow detection network is designed based on Swin Transformer, tackling this problem effectively. Moreover, in scenes with weak light intensity, other methods usually cannot handle shadow details well, while SaT can enhance shadow details by fusing multi-scale prediction results, such as in the 6th row.

4.3. Ablation study

To verify the effectiveness of the proposed model, we conduct an ablation study on each component of SaT. To this end, we compared SaT with its five variants:

- **Basic**, which uses a conventional shadow detection manner. We connect a single-channel 1 x 1 conv after side4 to generate shadow detection results, and use the basic binary cross-entropy (BCE) loss function to train the model.
- **Basic+LowFea**, which adds multi-scale feature fusion to the “Basic” model. We connect a single-channel 1 x 1 conv after each side (side1~side4) to generate multi-scale feature maps, and fuse these feature maps to obtain the final shadow mask.
- **Basic+DeSup**, which adds deep supervision to the “Basic+Low Fea” model. We perform shadow supervision on side1~side4, and then fuse the prediction maps generated by these sides to obtain the final shadow mask. Note that we optimize the model using a deep supervision loss, which is obtained by adding the BCE losses of side1~side4.
- **SaT\IB**, which does not use the information buffer unit. We remove all IBs in SaT and use different supervised tasks directly for the detection network. Like SaT, we optimize the model using the loss \( L(W) \) (Eq. (7)).
- **SaT\DeSup**, which does not use deep supervision. We remove the shadow supervision of side1~side3, but retain multi-scale feature fusion. We perform semantic supervision on side4 and the final fused semantic shadow.

We evaluate these models on the ISTD and SBD datasets, and the quantitative comparison results are summarized in Table 2. One can observe that the Basic model has the worst performance on both datasets. Compared with Basic, Basic+LowFea improves detection performance by adding multi-scale feature fusion, a common detail enhancement strategy that helps the model detect tiny shadows. Basic+DeSup utilizes deep supervision to intervene in early predictions, making the final fused shadow mask more accurate and improving detection performance further. We design SaT based on Basic+DeSup, and use shadow semantic supervision to help the model achieve the best performance. Fig. 10 shows the training details of SaT on the two datasets. One can see that the model can bring a lower BER and loss when combined with semantic supervision. The improvement in the ISTD dataset is more obvious since it contains fewer shadow categories and simpler category mutual dependencies than SBU.

Moreover, SaT\DeSup obtains a poor BER value (lower than SaT) after removing the deep supervision of the bottom layers, because the feature learning in the bottom layers lacks more guidance. In addition, the SaT\IB’s performance is lower than that of SaT\DeSup, and the BER values are relatively reduced by 11.06% (ISTD) and 4.55%
Table 2
Performance comparison of different network designs. Where “LowFea” denotes low-level feature fusion, “DeSup” denotes deep supervision, and “SeSup” denotes semantic supervision.

<table>
<thead>
<tr>
<th>Method</th>
<th>Network structure</th>
<th>Loss function</th>
<th>BER1</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Backbone</td>
<td>LowFea</td>
<td>IB</td>
</tr>
<tr>
<td>Basic</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Basic+LowFea</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>SaT(IB)</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>SaT(DeSup)</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>SaT (ours)</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

Fig. 10. Training analysis on two benchmark datasets.

Fig. 11. Advantage analysis of SaT.

Fig. 12 visualizes how the proposed SaT uses the loss functions to improve shadow detection gradually. For ambiguous cases, the background is usually the main factor affecting the accuracy of shadow detection. One can observe that the semantic supervision loss and the deep supervision loss can help effectively overcome the background influence. For example, when combined with the semantic supervision loss, the semantic-aware model can easily identify shadows and shadow-likes in the 1st row. When combined with deep supervision loss further, more complete and fine-grained shadow regions can be obtained, as shown in the 2nd row.

Discussion about the semantic supervision. Compared with conventional shadow detection, our SaT can better infer shadow structures and details by incorporating shadow semantic supervision, as shown in Fig. 13. This work aims to claim the importance of semantic information for shadow detection, rather than focusing on improving the performance of shadow semantic segmentation. Therefore, for a new scenario, the defined categories can provide semantic information to improve shadow detection. Despite inaccurate shadow semantic segmentation results from undefined categories, this does not affect the performance of the final shadow detection. For example, rows 4–6 in Fig. 13 show the visual results of the generalization experiment. Although the “board” and “stone” shadows (undefined) in rows 4 and 6 are misidentified as “building” and “artifact” shadows, the defined “plant” category can improve detection results. Furthermore, we find that semantic supervision is beneficial for detecting challenging self shadows in some cases (when the occluder has both cast and self shadows), as shown in rows 2 and 5.
4.4. Analysis of different IB structures

IB is designed to alleviate network gradient signal conflict in multi-task learning. Considering the number of the model’s parameters, instead of designing an overly complex structure, we design a simple yet effective IB with two successive residual convolutions (Res-conv) modules. To verify the effectiveness of IB, we also compare it with three other information buffers:

- IB\residual, which directly connects all convolutional layers in IB without using residual connections.
- 1 Res-conv, which uses 1 Res-conv as an information buffer.
- 3 Res-conv, which uses 3 Res-conv as an information buffer.

Table 3 presents the quantitative evaluation results of these four information buffers. One can observe that using 3 Res-conv or IB can obtain better performance. Compared to IB, the performance of using 3 Res-conv is only slightly improved. We adopt IB with 2 Res-conv after a trade-off between the model's parameters and performance.

4.5. Applications

4.5.1. Applying shadow detection to shadow removal

Accurate shadow detection is crucial for its subsequent shadow editing. In this section, we present an application for shadow removal. As shown in Fig. 14, some overhead images contain rich ground object information (Fig. 14a), but shadows weaken them. If we detect these shadows accurately (Fig. 14b), we can perform the histogram equalization operation on these shadow regions to obtain the shadow removal results (Fig. 14c). One can see that these images can clearly present the contours of the ground objects by shadow detection and removal, which is helpful for humans to collect image information.
designing a multi-task learning framework combined with semantic supervision. The experimental results demonstrate that our method can overcome the interference of ambiguous regions and improve the performance of shadow detection. Although our SaT achieves superior performance on several benchmark datasets, in cross-dataset experiments, the detection performance of our trained model may not improve significantly for new scenes with many unseen categories. In future work, we will define more common shadow categories to further enhance the model’s generalization ability.

Ethical standards

The work follows appropriate ethical standards. The proposed model trained with publicly available data, for which no ethical approval was required.

CRediT authorship contribution statement


Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

The authors appreciate the support from the Zhejiang Provincial Natural Science Foundation of China (LY20F020015, LY21F020015, LY21F020016), Zhejiang Province Key R&D Project (2021C02012) and the National Science Foundation of China (61902345, 61972121, 61902099).

References


