Full Length Article

Shadow detection via multi-scale feature fusion and unsupervised domain adaptation

Kai Zhou, Wen Wu, Yan-Li Shao *, Jing-Long Fang, Xing-Qi Wang, Dan Wei

Key Laboratory of Complex Systems Modeling and Simulation, School of Computer Science and Technology, Hangzhou Dianzi University, Hangzhou, 310018, PR China

A R T I C L E   I N F O

Keywords:
Image processing
Shadow detection
Unsupervised domain adaptation
Multi-scale feature fusion

A B S T R A C T

Shadow detection is significant for scene understanding. As a common scenario, soft shadows have more ambiguous boundaries than hard shadows. However, they are rarely present in the available benchmarks since annotating for them is time-consuming and needs expert help. This paper discusses how to transfer the shadow detection capability from available shadow data to soft shadow data and proposes a novel shadow detection framework (MUSD) based on multi-scale feature fusion and unsupervised domain adaptation. Firstly, we set the existing labeled shadow dataset (i.e., SBU) as the source domain and collect an unlabeled soft shadow dataset (SSD) as the target domain to formulate an unsupervised domain adaptation problem. Next, we design an efficient shadow detection network based on the double attention module and multi-scale feature fusion. Then, we use the global-local feature alignment strategy to align the task-related feature distributions between the source and target domains. This allows us to obtain a robust model and achieve domain adaptation effectively. Extensive experimental results show that our method can detect soft shadows more accurately than existing state-of-the-art methods.

1. Introduction

Shadows are caused by objects blocking light and are common in images and videos. Different distances between the occluder and the ground produce different shadows [1,2]. Shadows with clear boundaries are called hard shadows. Shadows with ambiguous boundaries (i.e., wider penumbra regions [3]) are called soft shadows. Shadow knowledge can provide valuable clues for scene understanding, such as light source direction [4,5], scene geometry description [6,7], and acquisition device parameters [8,9]. In addition, shadows may degrade the model’s performance in some vision tasks, such as target detection and recognition [10–12] and image segmentation [13,14]. Hence, accurate shadow detection in these scenes is crucial.

Existing shadow detection methods are mainly divided into traditional and deep learning-based methods. Traditional methods usually locate shadow regions by color [15,16], illumination [17,18], or hand-crafted features [19–21]. Deep learning-based methods [22–32] benefit from large-scale training data or deeper network structures and have become the mainstream methods for shadow detection. They achieve better detection performance than traditional methods. Although these mainstream methods can accurately detect hard shadows, they generally perform poorly in detecting soft shadows, such as incomplete shadow regions or missing details (Fig. 1a). However, soft shadows are also common in real-world scenes, even more than hard shadows, and it is urgent to develop a general method that can detect soft shadows accurately.

There may be two reasons for the above limitations: (1) Since most of the existing training data (i.e., SBU [23] and ISTD [33]) are labeled hard shadows, the existing shadow detection methods are mainly for hard shadows. The lack of labeled soft shadow data for training limits the model’s performance. The existing training data contains fewer soft shadows because it is more difficult to annotate soft shadows. Hard shadows have clear boundaries, and we can get shadow masks (labels) with a simple filtering process. Soft shadows have ambiguous boundaries and are difficult to identify. Annotating soft shadows is time-consuming and expensive (see Section 3.1 for details); (2) Existing shadow detection networks [27–29,31] are not sensitive to soft shadow details. In the detection, they usually cannot accurately identify shadow boundary, resulting in the loss of some soft shadow regions.

To tackle the first problem, we consider combining a domain adaptation method for the following reasons: (1) There are many similarities between hard and soft shadow images, such as lighting and color (Fig. 1b); (2) Compare with other benchmark datasets, although hard

This paper has been recommended for acceptance by Zicheng Liu.

* Corresponding author.
E-mail addresses: zhoukai.hdu.cs@gmail.com (K. Zhou), shaoyanli@hdu.edu.cn (Y.-L. Shao).

https://doi.org/10.1016/j.jvcir.2022.103596
Received 4 February 2022; Received in revised form 9 June 2022; Accepted 23 July 2022
Available online 6 August 2022
1047-3203/© 2022 Elsevier Inc. All rights reserved.
shadow data dominate SBU [23] dataset, it also contains about 15% soft shadow data. Therefore, we aim to develop an unsupervised domain adaptation (UDA) method to adapt shadow detection capability from the existing shadow dataset (source domain) to a new soft shadow dataset (target domain), as shown in Fig. 1b. However, implementing cross-domain shadow detection requires addressing the domain shift problem [34]. The shadow domain discrepancy mainly comes from different acquisition equipment and environment. For UDA, feature distribution alignment-based methods [35–40] are widely used to solve domain shift problem. In particular, some feature alignment methods [41–43] based on adversarial learning are applied to image segmentation, and the pixel-level domain adaptation is successfully achieved. Inspired by these works, we design a novel feature alignment method for the shadow detection task (see Section 3.2 for details), employing adversarial learning in the input and output spaces to efficiently extract domain-invariant features.

To enhance shadow details, MTMT-Net [30] directly fuses feature maps at the last four layers of CNN in the shadow region detection task. FSDNet [31] introduces a detail enhancement module to obtain shadow details from the low-level feature maps. Although they can significantly enhance shadow details, the detection performance degrades when detecting soft shadows due to weaker boundaries. To make the model more sensitive to soft shadow details, we introduce a scale feature refinement (SFR) module based on the double attention module [44]. Considering that different CNN layers contain complementary information, we refine features using SFR at each network layer. Then, the extracted multi-scale feature maps are fused to obtain the predicted shadow mask. Furthermore, unlike some shadow detection methods [24,26,30,33], which adopt encoder–decoder architecture to output by gradually up-sampling, we only retain encoder and directly up-sampling multi-layer feature maps to output. By doing this, we can reduce the model’s parameters and improve shadow detection efficiency.

From the above observations, we first collect a soft shadow dataset (SSD) for UDA and then propose a global–local feature alignment (GLA) strategy for domain adaptation to extract domain-invariant features accurately. Specifically, we utilize GLA in the UDA framework to transfer shadow detection knowledge from the existing shadow dataset to the SSD. Moreover, we introduce a detail enhancement module SFR into the shadow detection network to effectively avoid the prediction interference of false negatives and false positives. The main contributions of this paper are as follows:

- We propose a novel unsupervised domain adaptation framework (MUSD) to detect more challenging soft shadows. In detail, we introduce a global–local feature alignment strategy in the domain adaptation framework, which can effectively adapt the shadow detection capability from traditional shadow scenes to soft shadow scenes. To this end, we also collect a pure soft shadow dataset.

- We develop an efficient shadow detection network, which is more sensitive to shadow details. Specifically, we introduce a scale feature refinement module into the network to refine and compress shadow multi-scale feature maps to enhance shadow details and reduce model parameters.

- A robust shadow detection model can be obtained by applying the proposed shadow detection network to MUSD. This model can detect both soft and hard shadows, which improves generalization ability and inference accuracy.

2. Related works

2.1. Shadow detection

Shadow detection is a fundamental and challenging problem in computer vision. Existing traditional methods usually detect shadow by using unannotated shadow data for semi-supervised learning. However, these methods are only suitable for hard shadows with clear outlines and require high image quality. Hand-crafted feature-based methods [19–21] obtain shadow feature data by manually annotating shadows and then use these data to train a classifier. These methods can detect shadows accurately, but over-reliance on custom features and have poor generalization ability.

In recent years, the application of deep learning has significantly improved shadow detection performance. Khan et al. [22] and Vicente et al. [23] utilize convolutional neural networks (CNNs) to learn latent features for shadows, enabling fully supervised learning on labeled datasets. Nguyen et al. [24] proposed a novel conditional generative adversarial network with adjustable sensitivity parameters, which can further improve shadow detection accuracy. Hu et al. [25] introduce an attention mechanism in the recurrent neural network (RNN) and use direction-aware to acquire spatial context knowledge. Zhu et al. [26] explore CNN’s shallow and deep features and propose a recurrent attention residual (RAR) module, which can combine the global and local context to enhance shadow details. Le et al. [27] combine a shadow attenuation network (A-Net) and a detection network (D-Net) for adversarial learning, which improves the performance of the GAN framework. Zheng et al. [28] introduce a distraction-aware shadow (DS) module in the shadow detection framework to effectively avoid the prediction interference of false negatives and false positives.

Recently, Wang et al. [29] design an end-to-end framework for instance shadow detection to pair object instances and shadow instances. To address the limitations of limited labeled shadow data, Chen et al. [30] design a multi-task mean teacher (MTMT-Net) model for shadow detection by using unannotated shadow data for semi-supervised learning. However, the model’s performance degrades significantly in unseen shadow scenes. Hu et al. [31] create a shadow dataset in complex scenes and propose a fast shadow detection network.
We compose our SSD by collecting and shooting 2K soft shadow images in different scenes. The SSD dataset contains images with shadows, which are used as the source domain and soft shadow data as the target domain. To transfer the shadow detection knowledge from the source domain to the target domain, we use the GLA strategy (Section 3.3) to align the feature distributions in input and output spaces. We can successfully predict the target soft shadow mask by shadow detection knowledge transfer (domain adaptation).

Specifically, we first optimize the detection network G using the SSD loss and then predict the output of \( y_t \). Let \( f_1 \) and \( f_2 \) represent the features extracted by G in the source and target domains. Let \( o_1 \) and \( o_2 \) represent the logit maps generated by G in the source and target domains. We use the domain discriminator \( D_i \) (i.e., \( D_1 \) and \( D_2 \)) to distinguish whether the \( f/o \) is from the source or target domain during the training process. At this point, the detection network G and the discriminator \( D_i \) form an adversarial relationship. The gradient is back-propagated to G, which finally makes G generate a prediction distribution in the target domain similar to the source domain.

3.3. A new dataset for soft shadow - SSD

Labeled datasets (i.e., SBU [23] and ISTD [33]) have been created successively, resulting in gradual improvements in shadow detection performance. We refer to these two datasets as traditional shadow data in this paper. Considering that SBU contains about 15% soft shadow data, which is beneficial to domain knowledge transfer, we use SBU as the source domain. To transfer the shadow detection knowledge to soft shadow scenes, we create a new soft shadow dataset (SSD) as the target domain.

We compose our SSD by collecting and shooting 2K soft shadow images in different scenes. The differences between the SSD dataset and the existing datasets are as follows:

- **New scenes**: The SSD dataset contains images from new scenes that are different from the SBU and ISTD datasets.
- **Soft shadows**: The SSD dataset contains images with soft shadows, which are not present in the SBU and ISTD datasets.
- **Label distribution**: The SSD dataset contains images with soft shadows that are positively labeled, which is different from the SBU and ISTD datasets.
- **Shadow detection**: The SSD dataset contains images with soft shadows that are detected using the shadow detection algorithms, which is different from the SBU and ISTD datasets.

To generalize the model to soft shadow scenes, we propose a novel MUSD framework, as shown in Fig. 2. We first set the traditional shadow data \( \{x_s,y_s\} \) (with labels) as the source domain and soft shadow data \( \{x_t,y_t\} \) as the target domain. During training, we take \( \{x_s,y_s\} \) and \( x_t \) together as the input of the shadow detection network G (Section 3.2), and utilize the GLA strategy (Section 3.3) to transfer the shadow detection knowledge from the source domain to the target domain. GLA strategy uses adversarial learning to align feature distribution in input space and output space. We can successfully predict the target soft shadow mask by shadow detection knowledge transfer (domain adaptation).
### Table 1

Comparing with existing shadow detection datasets. Where “PSoft” represents the proportion of soft shadows in the dataset.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Content of images</th>
<th>Amount</th>
<th>PSoft</th>
</tr>
</thead>
<tbody>
<tr>
<td>SBU</td>
<td>Shadow/shadow mask</td>
<td>4727</td>
<td>&lt;15%</td>
</tr>
<tr>
<td>ISTD</td>
<td>Shadow/shadow mask/shadow-free</td>
<td>1870</td>
<td>&lt;2%</td>
</tr>
<tr>
<td>SSD</td>
<td>Training set shadow</td>
<td>1600</td>
<td>100%</td>
</tr>
<tr>
<td></td>
<td>Testing set Shadow/shadow mask</td>
<td>400</td>
<td>100%</td>
</tr>
</tbody>
</table>

Annotating soft shadows is tedious and time-consuming, and we show our approach to annotating soft shadows. We first take a soft shadow image with a fixed camera (Fig. 4a) and then quickly remove the occluder to take the corresponding non-shadow image (Fig. 4b). Secondly, we perform an image subtraction operation on Fig. 4a and b to obtain image differences (Fig. 4c). Then, we perform a binarization operation on Fig. 4c to highlight the contour of the shadow (Fig. 4d). Finally, we perform morphological filtering and manual adjustment on Fig. 4d to obtain the final soft shadow mask (Fig. 4e). Fig. 4 shows the annotation process for the SSD testing set.

### 3.2. Shadow detection network

To improve the soft shadow detection accuracy, we need to design a detection network that is more sensitive to shadow details. Existing methods [24,26,30,33] mainly use the encoder–decoder framework to design networks. These methods improve performance from two points: (1) low-level features are used to obtain image detail information, which helps to detect tiny shadows and shadow boundaries, and (2) high-level features are used to obtain image semantic information, which helps to distinguish shadows and backgrounds. However, to improve the inference accuracy, they usually design the network deeper or wider, making the model parameters gradually increase. More parameters will increase the computational complexity, which is not conducive to the subsequent tasks of shadow detection.

To address above issues, as shown in Fig. 5, we introduce four scale feature refinement (SFR) module into the network to compress the feature maps extracted by the encoder (compressed to 16 channels). Then, these feature maps are transformed to the same size by upsampling. Finally, they are fused from low-level to high-level to generate a detection result (shadow mask). SFR is designed to reduce the model's complexity and consists of a double attention module (channel attention and spatial attention) [44] and a convolutional layer. We use four SFRs in the network, making the network more sensitive to shadow details by collecting multi-scale features. Experiments show that our network can accurately detect shadow regions with only 11.34M parameters, which is faster than most state-of-the-art methods. Note that in the MUSD framework, the detection network G adopts the proposed shadow detection network, and the network backbone adopts ResNet16.

For domain adaptation training, we first introduce the detection loss used to optimize the detection network G in the source domain. Let $x_i = \{x_1, x_2, \ldots, x_N\}$ represent the shadow images and $y_i = \{y_1, y_2, \ldots, y_N\}$ represent the corresponding labels. The network G can learn the mapping between $x_i$ and $y_i$. Let $p_i$ represent the shadow detection result (prediction map). In our task, binary cross-entropy (BCE) is used as the detection loss $L_d$, and we optimize G by minimizing $L_d$:

$$L_d = \frac{1}{N} \sum_{i=1}^{N} \Phi_{\text{BCE}}(p_i, y_i),$$

where $\Phi_{\text{BCE}}(p_i, y_i) = -y_i \log(p_i) + (1 - y_i) \log(1 - p_i)$.

#### 3.3. Global–local feature alignment strategy (GLA)

To implement UDA, we need to learn domain invariant features (i.e., shadow shared features) [37] from source and target domains. Domain invariant features extracted at feature-level can be successfully applied in image classification [40,47]. Unlike image classification, shadow detection is pixel-level adaptation and needs to encode more visual cues such as appearance, light intensity and context. Some pixel-level adaptations, such as semantic segmentation [41,43,48], can segment most objects in the target image but with low segmentation accuracy. However, shadow detection requires higher inference accuracy to detect shadow regions accurately. Therefore, it is urgent to develop an effective shadow detection domain adaptation method.

In neural networks, low-level features contain rich details, such as color, illumination, noise, and resolution; high-level features contain abstract semantics, such as semantic structure and deformation. In this paper, the low-level features are called global features (i.e., $f_1$ and $f_2$); the high-level features are related to tasks and are called local features (i.e., $a_1$ and $a_2$). Fig. 6 shows some feature maps generated in shadow detection. It can be observed that low-level feature maps contain many task-unrelated redundant features, such as color features of shadow-like regions (shadow-like materials). During the domain adaptation process, if redundant features are extracted for shadow feature learning, it will degrade the detection performance of the target domain. The high-level feature maps contain many task-related features and are more suitable
and \( P \) becomes smaller and smaller, which makes the detection distribution more difficult to determine whether they are aligned. We pass the features \( f_1 \) and \( f_2 \) through G to output logit maps \( o_1 \) and \( o_2 \), and use D_2 to estimate the distance between the \( o_1 \) and \( o_2 \) distributions. In adversarial learning, the task of the discriminator \( D_2 \) is to distinguish which domain the feature/logits come from, and the task of the generator G is to make the feature/logits distribution of the target task close to the source domain. As the weights of G are updated, the Wasserstein distance becomes smaller and smaller, which makes the detection distribution generated by G in the target domain more and more similar to the source domain. The Wasserstein distance is expressed as:

\[
W(P_1, P_2) = \inf_{\gamma} \int \| x - y \| d\gamma(x, y),
\]

where \( P_1 \) and \( P_2 \) represent the feature/logits distribution, \( \gamma(x, y) \) represents the joint distribution, and \( \int (P_1, P_2) \) represents the boundaries of \( \gamma \) are \( P_1 \) and \( P_2 \), respectively.

For shadow detection domain adaptation, considering shadow details and semantic knowledge transfer, we propose a global–local feature alignment strategy (GLA) based on GAN. As shown in Fig. 2, we first use discriminator \( D_1 \) to align the global feature distributions and then use discriminator \( D_2 \) to align the local feature distributions. Since the logit map \( o_1 \) and \( o_2 \) contains rich semantic information, the latter is also known as semantic-aware adversarial learning. In GLA, global feature alignment aims to learn shadow detail features such as soft shadow edges and tiny shadows, and local feature alignment aims to learn task-related features such as shadow contours. By doing this, we can extract accurate domain invariant features for shadow detection domain adaptation.

For adversarial learning, we first use \( D_1 \) to implicitly estimate the Wasserstein distance [49] between features \( f_1 \) and \( f_2 \) distributions to determine whether they are aligned. We pass the features \( f_1 \) and \( f_2 \) through G to output logit maps \( o_1 \) and \( o_2 \), and use \( D_2 \) to estimate the distance between the \( o_1 \) and \( o_2 \) distributions. In adversarial learning, the task of the discriminator \( D_1 \) is to distinguish which domain the features/logits come from, and the task of the generator G is to make the feature/logits distribution of the target task close to the source domain. As the weights of G are updated, the Wasserstein distance becomes smaller and smaller, which makes the detection distribution generated by G in the target domain more and more similar to the source domain. The Wasserstein distance is expressed as:

\[
W(P_1, P_2) = \inf_{\gamma} \mathbb{E}_{(x,y)\sim P_1} \| x - y \|_\gamma.
\]

The mappings \( x_i \rightarrow f_1, x_i \rightarrow f_2, x_i \rightarrow o_1, \) and \( x_i \rightarrow o_2 \) are represented by \( z_1, z_2, u_1, \) and \( u_2, \) respectively. The distributions of features \( f_1 \) and \( f_2 \) are represented by \( f_1(z_1(x_i)) \sim P_{f_1} \) and \( f_2(z_2(x_i)) \sim P_{f_2}, \) respectively. The distributions of logits \( o_1 \) and \( o_2 \) are represented by \( o_1 = u_1(x_i) \sim P_{o_1} \) and \( o_2 = u_2(x_i) \sim P_{o_2}, \) respectively. The mappings \( f_1 \rightarrow D_1 \) (or \( f_2 \rightarrow D_1 \)) and \( o_1 \rightarrow D_2 \) (or \( o_2 \rightarrow D_2 \)) are represented by \( c_1 \) and \( c_2, \) respectively. Given the source data \( (x_i, y_i) \) and the target data \( x_i \), in the global feature alignment, the losses of optimizing G and \( D_2 \) are expressed as:

\[
\min_{z_2} L_{c_2} = -\mathbb{E}_{z_2(x_i)\sim P_{z_2}} [c_2(z_2(x_i))],
\]

\[
\min_{z_1} L_{c_1} = \mathbb{E}_{z_1(x_i)\sim P_{z_1}} [c_1(z_1(x_i))]
\]

\[
- \mathbb{E}_{z_2(x_i)\sim P_{z_2}} [c_2(z_2(x_i))], s.t. \|c_1\|_\mathcal{K},
\]

where \( \mathcal{K} \) is a constant under the Lipschitz constraint.

Similarly, in the local feature alignment, the losses of optimizing G and \( D_2 \) are expressed as:

\[
\min_{z_2} L_{c_2} = -\mathbb{E}_{z_2(x_i)\sim P_{z_2}} [c_2(z_2(x))] ,
\]

\[
\min_{z_1} L_{c_1} = \mathbb{E}_{z_1(x_i)\sim P_{z_1}} [c_2(z_2(x))],
\]

\[
- \mathbb{E}_{z_2(x)\sim P_{z_2}} [c_2(z_2(x))], s.t. \|c_2\|_\mathcal{K},
\]

The overall loss is obtained by adding all of the above losses:

\[
L = \lambda_1 L_{d} + \lambda_2 (L_{c_1} + L_{c_2}) + \lambda_3 (L_{c_2} + L_{u_2}),
\]

where \( \lambda \) represents the weight.

4. Experiments

In this section, we first introduce the relevant settings of the experiment (Section 4.1.1). Then, we compare our method with state-of-the-art
optimize the domain discriminator, with a learning rate of 0.0002. The learning rate is set to 0.0001, and the momentum and weight decay are set to 0.9 and 5 × 10⁻⁵, respectively. The LeakyReLU slope is set to 0.25. During training, we set training epochs to 35, 2, and the learning rate decrease strategy mentioned in [51] is adopted. We use Adam to optimize the domain discriminator, with a learning rate of 10⁻⁴, and the same decrease strategy as the detection network.

4.1. Experimental settings

Experimental datasets. We use the benchmark dataset SBU as the source domain and the proposed dataset SSD as the target domain. In traditional shadow detection, we evaluate MUSD's performance on benchmark datasets SBU and ISTD, respectively. Among them, (1) SBU contains 4727 shadow images, all with corresponding labels (shadow masks); (2) ISTD contains 1870 shadow images, all with corresponding labels; (3) SSD contains 2000 shadow images, and only 400 testing images have corresponding labels.

Evaluation metrics. For comparison with state-of-the-art methods, we use the balance error rate (BER) as the evaluation metric:

\[
BER = 1 - \frac{1}{2} \left( \frac{TP}{TP + FN} + \frac{TN}{TN + FP} \right),
\]

where \(TP\), \(TN\), \(FP\), and \(FN\) are the number of true positive, true negative, false positive, and false negative pixels, respectively. In experiments, a lower BER value indicates a better shadow detection performance. We use “overall” to represent the overall pixel error rate, and “shadow” and “non-shadow” to represent the pixel error rate of each class (shadow and non-shadow).

Implementation details. In the proposed MUSD, we train the detection network and two domain discriminators simultaneously. For each training batch, we first optimize the detection loss \(L_d\) of the detection network \(G\) with the source data \(\{x_s, y_s\}\), and then use \(G\) to make predictions on the target data \(x_t\). During training, we pass the features \((f_d, f_s)\) and the logits \((o_d, o_s)\) to discriminators \(D_1\) and \(D_2\), respectively, to optimize \(L_{d1}\) and \(L_{d2}\), and then we compute the adversarial losses \(L_{c1}\) and \(L_{c2}\).

Our MUSD network framework is built on Python 3.6 and TensorFlow 1.9.0, and the hardware platform is Intel i9 10900K, GeForce RTX 3080 GPU, 32 GB RAM. The detection network \(G\) in the MUSD framework uses the proposed shadow detection network. Following by CGAN [50] implements two discriminator networks (i.e., \(D_1\) and \(D_2\)) with 5 convolutional blocks. In each convolutional block, the convolutional layer is preceded by the activation function LeakyReLU, followed by batch normalization, and the last layer is the Sigmoid function. LeakyReLU slope is set to 0.25. During training, we set training epochs to 35, \(\lambda_1\) to 1, \(\lambda_2\) to 0.5, and \(\lambda_3\) to 1 in Eq. (7). The input image size is set to 256 x 256, and the batch size is set to 48. We use the stochastic gradient descent (SGD) to optimize the detection network, and the momentum and weight decay are set to 0.9 and 5 x 10⁻⁵, respectively. The learning rate is set to 2.5 x 10⁻⁴, and the learning rate decrease strategy mentioned in [51] is adopted. We use Adam to optimize the domain discriminator, with a learning rate of 10⁻⁴, and the same decrease strategy as the detection network.

4.2. Shadow detection

4.2.1. Soft shadow detection

Comparison with existing soft shadow detection (or removal) methods. There are few existing soft shadow detection methods, so we compare the proposed MUSD with three related methods, i.e., Gryka et al. [1], DC-ShadowNet [52] and Soft-DA [53]. Table 2 shows the quantitative evaluation results of several methods. It can be seen that our method has the lowest BER value. Gryka et al. [1] and DC-ShadowNet [52] are soft shadow removal methods. We perform image subtraction between the shadow-removed image and the shadow image to obtain the image difference and then obtain the shadow mask by binarization. This indirect solution manner makes the obtained shadow mask more uncertain. Soft-DA [53] is a method for hard and soft shadow detection, which requires annotating part of the training data and uses different detectors to detect soft and hard shadows. The semi-supervised training makes the model still rely on label data and has weak generalization ability. Our method utilizes the UDA method and the detail enhancement strategy to adapt the shadow detection capability to the unseen soft shadow scene, and obtains the best soft shadow mask, as shown in Fig. 7.

Comparison with state-of-the-art methods (for soft shadow detection). We compare the proposed MUSD with seven state-of-the-art methods: (1) four traditional shadow detection methods, i.e., BDRAR [26], DSDNet [28], MMTT-Net [30] and FDRNet [32], and (2) three UDA-based methods are applied to soft shadow detection, i.e., Saito et al. [41], DSFN [42] and PCS [40].

Table 2 presents the quantification results of all methods. We can conclude by observation: (1) UDA-based methods have better performance than traditional shadow detection methods. Traditional methods cannot be adapted to soft shadow detection due to the limitation of datasets. However, UDA-based methods have the generalization ability to detect soft shadows further accurately. (2) Our method has the best detection performance among UDA-based methods. Compared to Saito et al. [41] and DSFN [42], our method efficiently utilizes the GLA strategy to align task-related feature distributions. Compared to PCS [40], our method explores pixel-level domain adaptation with better detection accuracy.

To further compare all methods, we also show the visualization results in Fig. 8. Overall, UDA-based methods (Fig. 8f–h) have a better effect than traditional methods (Fig. 8c–e). Compared with other methods, our results are the closest to ground truths that can detect the shadow regions completely. For soft shadows with ambiguous boundaries, such as the 1st, 2nd, and 3rd rows, none of the existing methods can overcome the uncertainty of shadow boundary. However, our method can handle boundaries better, and the detected boundaries are more fine-grained. This is because the shadow detection network
Table 2
Comparing with state-of-the-art methods (for soft shadow detection).

<table>
<thead>
<tr>
<th>Methods</th>
<th>Source/Year</th>
<th>BER&lt;sub&gt;Overall&lt;/sub&gt;</th>
<th>BER&lt;sub&gt;Shadow&lt;/sub&gt;</th>
<th>BER&lt;sub&gt;Non-shadow&lt;/sub&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Soft shadow detection</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Gryka et al. [1]</td>
<td>TOG/2015</td>
<td>16.28</td>
<td>15.02</td>
<td>17.54</td>
</tr>
<tr>
<td><strong>Traditional shadow detection</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BDRAR [26]</td>
<td>ECCV/2018</td>
<td>15.23</td>
<td>15.84</td>
<td>14.61</td>
</tr>
<tr>
<td><strong>UDA-based</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Saito et al. [41]</td>
<td>CVPR/2018</td>
<td>11.87</td>
<td>11.58</td>
<td>12.17</td>
</tr>
<tr>
<td>PCS [40]</td>
<td>CVPR/2021</td>
<td>10.14</td>
<td>9.78</td>
<td>10.50</td>
</tr>
<tr>
<td>MUSD (ours)</td>
<td>–</td>
<td>8.63</td>
<td>7.57</td>
<td>9.69</td>
</tr>
</tbody>
</table>

Fig. 8. Comparison of soft shadow detection results. (a) is input image, (b) is ground truth, (c) is predicted mask of DSDNet, (d) is predicted mask of MTMT-Net, (e) is predicted mask of FDRNet, (f) is predicted mask of Saito et al., (g) is predicted mask of DSNF, (h) is predicted mask of PCS, (i) is predicted mask of our MUSD.

we designed is more sensitive to shadow details. For soft shadows with lighter colors, such as the 4th row, none of the existing methods can distinguish shadow and background, but our method can accurately distinguish them.

4.2.2. Traditional shadow detection

For traditional shadow detection, we compare our method with several state-of-the-art methods, including Stacked-CNN [23], ScGAN [24], DSC [25], A+D Net [27], DSDNet [28], MTMT-Net [30] and FSDNet [31]. We analyze all methods on the benchmark datasets SBU and ISTD, respectively.

The quantification results are shown in Table 3, where “FPS” represents the number of frames per second processed images (we evaluate all methods on GeForce RTX 3080 GPU), and “P” represents the number of the model’s parameters. MTMT-Net [30] is the most accurate method in previous work among the several methods. Compared with MTMT-Net, our method can detect soft shadows in SBU dataset more accurately, with an overall BER improvement of 3.17%. Since there are very few soft shadow data (less than 2%) in the ISTD dataset, the advantages of our method cannot be highlighted in ISTD. In addition, both FSDNet [31] and our method improve the detection efficiency by reducing the model’s parameters. Although our parameters are more than FSDNet, FSDNet comes at the expense of inference accuracy. Therefore, overall, our method achieves the best performance.

Figs. 9 and 10 show some visualization results. We can see that our method performs on par with state-of-the-art methods for traditional shadow detection. When different materials appear in the background, our method can detect shadows completely, while some other methods are affected by the background, such as the 2nd row in Fig. 9 and the 1st and 3rd rows in Fig. 10. For well-structured shadows, most methods can detect shadow regions completely, and our method can detect more continuous and fine-grained boundaries than some other methods, such as the 2nd and 4th rows in Fig. 10. In addition, our method can also better detect tiny shadows, such as the 1st and 3rd rows in Fig. 9.

4.3. Ablation study

4.3.1. Ablation study for GLA strategy

To verify the effectiveness of the proposed domain adaptation method, we conduct an ablation study on our GLA strategy.

- First, we extract the detection network G as a Basic model, which does not use any domain adaptation method.
- Then, we add the discriminator D₁ to the Basic model, using G-D₁ adversarial learning to align the global feature distributions, named Basic+Global.
- Further, we add the discriminator D₂ to the Basic model, using G-D₂ adversarial learning (semantic-aware adversarial learning) to align the local feature distributions, named Basic+Local.
- Next, we add the discriminators D₁ and D₂ to the Basic model to form our global–local feature alignment strategy (GLA), i.e., our MUSD.

Table 4 summarizes the quantification results. The Basic model does not use any domain adaptation methods, so it does not generalize well to soft shadow detection and performs the worst. Domain adaptation based on three feature alignment strategies can improve soft shadow detection performance. The Basic+Global model aligns the global feature distribution of the source and target domains without considering the specific detection tasks. Still, it promotes domain adaptation, and the BER value is improved by 16.39% compared to Basic. The Basic+Local uses semantic-aware learning to align task-related feature distributions, and the BER value is improved by 27.89% compared to Basic, which proves that this strategy is effective for shadow detection domain adaptation. Following this line of thought,
Table 3
Comparing with state-of-the-art methods (for traditional shadow detection).

<table>
<thead>
<tr>
<th>Methods</th>
<th>Source/Year</th>
<th>FPS</th>
<th>P(M)</th>
<th>BER(SBU)</th>
<th>BER(ISTD)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Overall</td>
<td>Shadow</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Non-shadow</td>
<td>Overall</td>
</tr>
<tr>
<td>A+D Net [27]</td>
<td>ECCV/2018</td>
<td>68.02</td>
<td>54.41</td>
<td>5.37</td>
<td>4.45</td>
</tr>
<tr>
<td>MTMT-Net [30]</td>
<td>CVPR/2020</td>
<td>21.88</td>
<td>89.62</td>
<td>3.15</td>
<td>3.73</td>
</tr>
<tr>
<td>FSDNet [31]</td>
<td>TIP/2021</td>
<td>124.62</td>
<td>4.40</td>
<td>4.37</td>
<td>4.05</td>
</tr>
<tr>
<td>MUSD (ours)</td>
<td></td>
<td>98.46</td>
<td>11.34</td>
<td>3.05</td>
<td>3.38</td>
</tr>
</tbody>
</table>

Fig. 9. Comparison of traditional shadow detection results on SBU dataset. (a) is input image. (b) is ground truth. (c) is predicted mask of Stacked-CNN. (d) is predicted mask of ScGAN. (e) is predicted mask of DSC. (f) is predicted mask of DSDNet. (g) is predicted mask of MTMT-Net. (h) is predicted mask of FSDNet. (i) is predicted mask of our MUSD.

Fig. 10. Comparison of traditional shadow detection results on ISTD dataset. (a) is input image. (b) is ground truth. (c) is predicted mask of Stacked-CNN. (d) is predicted mask of ScGAN. (e) is predicted mask of DSC. (f) is predicted mask of DSDNet. (g) is predicted mask of MTMT-Net. (h) is predicted mask of FSDNet. (i) is predicted mask of our MUSD.

Fig. 11. Comparison of soft shadow detection results.
we combine global and local feature alignment strategies, which is the proposed GLA strategy. Experiments show that GLA achieves better domain adaptation results, and the BER value is improved by 35.98% compared to Basic. To sum up, we can conclude that GLA aligns the feature distribution related to the detection task from input to output space, ensuring the correct learning direction.

Fig. 11 shows some visualization results. It can be observed that the Basic model is difficult to detect soft shadows without using domain adaptation, and the detection results of the three feature alignment strategies are gradually improved.

### 4.3.2. Ablation study for shadow detection network

To verify the effectiveness of the proposed shadow detection network, we conduct an ablation study on the network structure.

- Firstly, we directly output shadow detection results at the top layer of the Backbone network.
- Then, we add multi-scale feature fusion (MFF) on the basis of Backbone to form a Backbone+MFF model, which is to fuse the feature maps generated from the bottom layer to the top layer to obtain the detection result.
- Finally, we connect SFR to each layer on the basis of Backbone+MFF, which is our strategy.

As shown in Table 5, we can observe that combining multi-scale feature fusion (Backbone+MFF) can improve the model’s performance compared to conventional shadow detection (Backbone). Further, we can get the best performance by embedding SFR modules in the network.

### 4.4. Application

This section presents an application to demonstrate that accurate shadow detection is necessary. As shown in Fig. 12, we show a subsequent task for shadow detection: shadow removal. In this application, the images contain important text or mark information, weakened by shadows. However, these images can be enhanced for viewing after shadow detection and removal. In real scenes, shadows are common in these images, and a general shadow detection method is crucial, so we propose MUSD. Our subsequent work will accurately remove shadows in these images.

### 5. Conclusion

In this paper, we explore general shadow detection methods and propose MUSD, which aims to generalize shadow detection to soft shadow scenes. Soft shadows have more ambiguous boundaries than hard shadows, making detecting soft shadows more challenging than hard shadows. To address this problem, we propose two strategies: (1) We introduce the idea of unsupervised domain adaptation (UDA) and propose a global–local alignment strategy for UDA to adapt the shadow detection capability from traditional shadow scenes to soft shadow scenes; (2) We design a detail-sensitive shadow detection network based on a double attention module to adapt to soft shadow detection. Combining the two strategies can obtain a robust shadow detector that can detect both soft and hard shadows. Finally, we also show our subsequent work in application: shadow removal.

### 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

No data was used for the research described in the article.

### Acknowledgments

This work was supported by the Zhejiang Provincial Natural Science Foundation of China (LY20F020015, LY21F020015); the National Science Foundation of China (61902345, 61972121, 61902099); and the Open Project Program of the State Key Lab of CAD & CG (Grant No. 2109), Zhejiang University.
The work follows appropriate ethical standards. The proposed model was trained with publicly available data, for which no ethical approval was required.

References