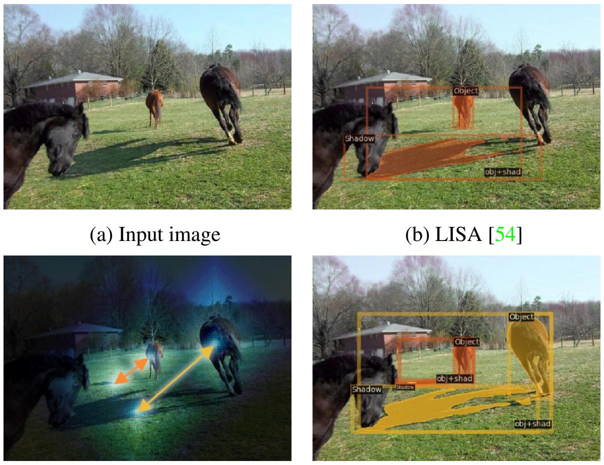
Single-Stage Instance Shadow Detection with Bidirectional Relation Learning

Tianyu Wang[1,*], Xiaowei Hu[1,*,†], Chi-Wing Fu[1], and Pheng-Ann Heng[1,2]
[1] Department of Computer Science and Engineering, The Chinese University of Hong Kong
[2] Guangdong-Hong Kong-Macao Joint Laboratory of Human-Machine Intelligence-Synergy System,
Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences

Abstract

Instance shadow detection aims to find shadow instances paired with the objects that cast the shadows. The previous work adopts a two-stage framework to first predict shadow instances, object instances, and shadow-object associations from the region proposals, then leverage a post-processing to match the predictions to form the final shadow-object pairs. In this paper, we present a new single-stage fully-convolutional network architecture with a bidirectional relation learning module to directly learn the relations of shadow and object instances in an end-to-end manner. Compared with the prior work, our method actively explores the internal relationship between shadows and objects to learn a better pairing between them, thus improving the overall performance for instance shadow detection. We evaluate our method on the benchmark dataset for instance shadow detection, both quantitatively and visually. The experimental results demonstrate that our method clearly outperforms the state-of-the-art method.



(a) Input image (b) LISA [54]

(c) Our *learned* paired locations (d) Ours

Figure 1: Instance shadow detection results produced by (b) LISA [54] and (d) our method, where LISA mismatches the large shadow with another horse. Our method learns the paired locations (c) for a better shadow-object pairing.

1. Introduction

Research on shadows has been a fundamental problem in computer vision. Prior works focus mainly on the shadow detection and shadow removal tasks, while a recent work [54] proposes a new task called *instance shadow detection*, which aims to detect shadow instances paired with the associated objects that cast the shadows. Overall, instance shadow detection benefits many vision applications, such as light direction estimation and photo editing.

To approach instance shadow detection, Wang and Hu *et al*. [54] built a dataset called SOBA, each with labeled masks for shadow instances, object instances, and shadow-object associations. To detect the shadow instances together with their associated objects, they designed a two-stage framework to first generate region proposals that have high probabilities of containing shadow instances, object instances, and shadow-object associations. Then, in the second stage, for each proposal, regions-of-interest (ROIs) are cropped from the feature maps, and the two-stage framework makes predictions of the masks and boxes of the shadow instances, object instances, and shadow-object associations from each ROI. Lastly, they formulate a strategy to pair up the shadow and object instances with the shadow-object associations predicted from a deep network.

After revisiting the task, we identified various limitations in [54]. First, it considers the shadow-object association as a single category and predicts a bounding box for each shadow-object association. However, the appearance of shadow and object instances are very different, so shadow-object associations could easily be missed; see Figure 1 (b) for an example. Second, this method generates region

[*] Joint first authors
[†] Corresponding author (xwhu@cse.cuhk.edu.hk)

proposals for shadow/object instances and shadow-object associations using two separate branches, and leverages a post-processing to produce the final shadow-object associations. Errors could accumulate through the two-stage deep network and the post-processing, thus leading to large performance degradation. Third, the ROIs employed in [54] represent feature regions using rectangular shapes, but the shapes of the shadow instances and shadow-object associations are usually irregular, and the cropped ROIs of rectangular shapes could include many irrelevant image contents, such as other object and shadow instances.

In this paper, we present a single-stage deep network for instance shadow detection by *directly* learning to find the relation between shadow instances and object instances in an end-to-end manner. Our new approach includes *only* fully convolutional operations and is able to handle shadow/object instances and shadow-object associations of *arbitrary shape*. Specifically, we jointly optimize our network model to find shadow instances, object instances, and shadow-object associations, thereby enabling us to efficiently explore the internal relationship between shadows and objects. Importantly, we design the bidirectional relation learning module to explore the shadow-object association pairs, in which we learn an offset vector from the center of each shadow instance to the center of its associated object instance, as well as the other way around; see Figure 1 (c) for the learned locations of paired shadow and object instances. To facilitate the learning process, we adopt class vectors to indicate the learning directions (shadow → object or object → shadow) and use the segmentation loss and offset loss to optimize the network. These new techniques help the network to learn a better pairing between shadow and object instances, thus improving the overall performance of instance shadow detection; see Figures 1 (b) & (d) for the visual comparison results. Below, we summarize the major contributions of this work.

- First, we design a single-stage instance shadow detection network with only fully convolutional operations to predict shadow instances, object instances, and shadow-object associations in an end-to-end manner.
- Second, we formulate the bidirectional relation learning module in a deep network to learn the relation between shadow instances and object instances.
- Third, we compare our method with the previous state-of-the-art method on the benchmark dataset for instance shadow detection. Results show that our method outperforms the state-of-the-art with an improvement of over 29% in accuracy.

2. Related Work

Shadow detection. Generic shadow detection aims to produce a binary mask to mark the shadow regions in the input image. Early methods explored spectral properties [42, 45] or built illumination models [37] to detect shadows. Later, learning-based approaches were introduced to detect shadows by exploring various hand-crafted features, *e.g*., T-junction [23], texture [65, 49, 12, 51], color [23, 49, 12, 51], and edge [23, 65, 20]. However, methods based on hand-crafted features have limited representation capabilities and may fail to detect shadows in complex environments.

In recent years, deep-learning-based methods show remarkable performance on shadow detection. Khan *et al*. [21] presented the first work that detects shadows by automatically learning features through a convolutional neural network (CNN). Shen *et al*. [43] and Hou & Vicente *et al*. [15, 52] formulated a structured learning framework and a stacked-CNN, respectively, for shadow detection. Nguyen *et al*. [36] used an adjustable parameter in a conditional generative adversarial network to balance the weights of shadow and non-shadow regions. Wang *et al*. [53] sequentially detected and removed shadows by leveraging two conditional generative adversarial networks. Hu *et al*. [16, 19] formulated an attention mechanism in a spatial recurrent network to learn the direction-aware spatial context for shadow detection. Le *et al*. [26] used generated adversarial training samples to train a shadow detection network, in which the training samples were generated by a shadow attenuation network. Zhu *et al*. [66] designed a bidirectional feature pyramid network with recurrent attention residual modules to detect shadows. Zheng *et al*. [62] formulated a distraction-aware shadow detection network by revisiting the predicted false negatives and false positives. Ding *et al*. [8] detected and removed shadows in a recurrent manner through an attentive recurrent generative adversarial network. Later, Hu *et al*. [18] built a new dataset to support shadow detection in a complex world and designed a fast shadow detection network. Chen *et al*. [4] presented a semi-supervised shadow detection algorithm by exploring the knowledge from unlabeled data through a multi-task mean teacher framework.

Apart from generic shadow detection, various recent works explored deep learning to remove shadows in natural images [22, 41, 17, 8, 24, 5, 60, 25] and in document images [31], to generate shadows in augmented reality [32], and to manipulate portrait shadows [61]. Very recently, Wang and Hu *et al*. [54] proposed a new shadow detection task called instance shadow detection. This is the most related work to ours. It designed a two-stage framework and adopted a post-processing to predict the paired shadow and object instances. In contrast, in this work, we formulate the bidirectional relation learning module in a fully convolutional network to directly learn the relation between shadow instances and object instances in an end-to-end manner.

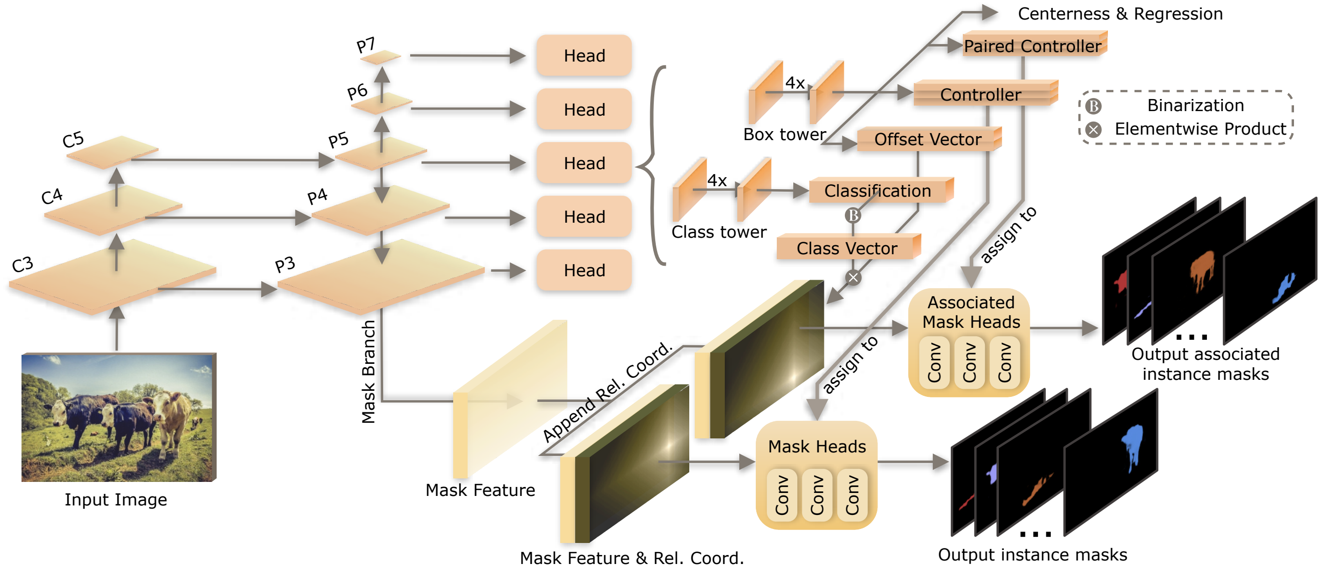


Figure 2: The schematic illustration of our single-stage instance shadow detection network (SSIS). The mask feature and the outputs of the box tower and class tower are used to formulate the bidirectional relation learning module; see Figure 3. Note that each head has its own box head and class head, and the filter parameters among these heads are sharing.

Instance segmentation. One category of methods for instance segmentation predicts region proposals in the input images and then generates an instance mask for each proposal, *e.g.*, MNC [6], DeepMask [39], InstanceFCN [6], and SharpMask [40], FCIS [28], BAIS [13], MaskLab [2], Mask R-CNN [14], PANet [33], MegDet [38], and HTC [1]. Among these works, Mask R-CNN simultaneously predicted the category label, bounding box, and segmentation mask for each region proposal and achieved great success for instance segmentation. The other category directly predicts the instance masks and the corresponding categories in the whole images, *e.g.*, TensorMask [3], SSAP [9], SOLO [55], EmbedMask [58], SOLOv2 [56], CenterMask [27], and CondInst [47]. Among them, CondInst developed a dynamic instance-aware network that learns to generate different network parameters for different instances and achieved comparable performance with Mask R-CNN. CondInst serves as the basic network, on which we further formulate our bidirectional relation learning module to learn the relation between shadow and object instances in a fully convolutional manner.

Visual relation detection. This task aims to find the objects and their relationships from the images. Newell and Deng [35] identified the objects and their relations in a CNN as a graph. Zhang et al. [59] embedded objects and relations into two vector spaces. Though the relation between shadow and object in an image can be regarded as a type of visual relation, works on visual relation detection mainly focus on detecting relations that are single direction, while our method simultaneously learns a bidirectional relation between shadows and objects in a single network.

3. Methodology

3.1. Overall Network Architecture

Figure 2 shows the overview architecture of our single-stage instance shadow detection network (SSIS) that employs the bidirectional relation learning module. Given the input image, we adopt a convolutional neural network to extract the feature maps in different resolutions, and build a feature pyramid network [29] with multiple feature levels (from $P3$ to $P7$). Then, we adopt multiple heads at different levels and add four convolutional layers in a class tower to predict the classification scores and another four convolutional layers in a box tower to generate other predictions. In summary, we obtain a set of predictions for each head:

(i) **classification** scores, which are used to indicate the categories of shadow, object, and background;

(ii) **offset vector**, which are the image-space vectors from the centers of the shadow instances to the centers of the corresponding object instance, and vice versa;

(iii) **controller** and **paired controller**, which learn two sets of filter parameters used in the mask head to predict the masks for shadow/object instances. Note that each instance has its individual filter parameters to predict mask; see [47] for detail. In our framework, if the controller generates the filter parameters for a shadow instance, the paired controller will generate the filter parameters for the corresponding object instance, and vice versa; and

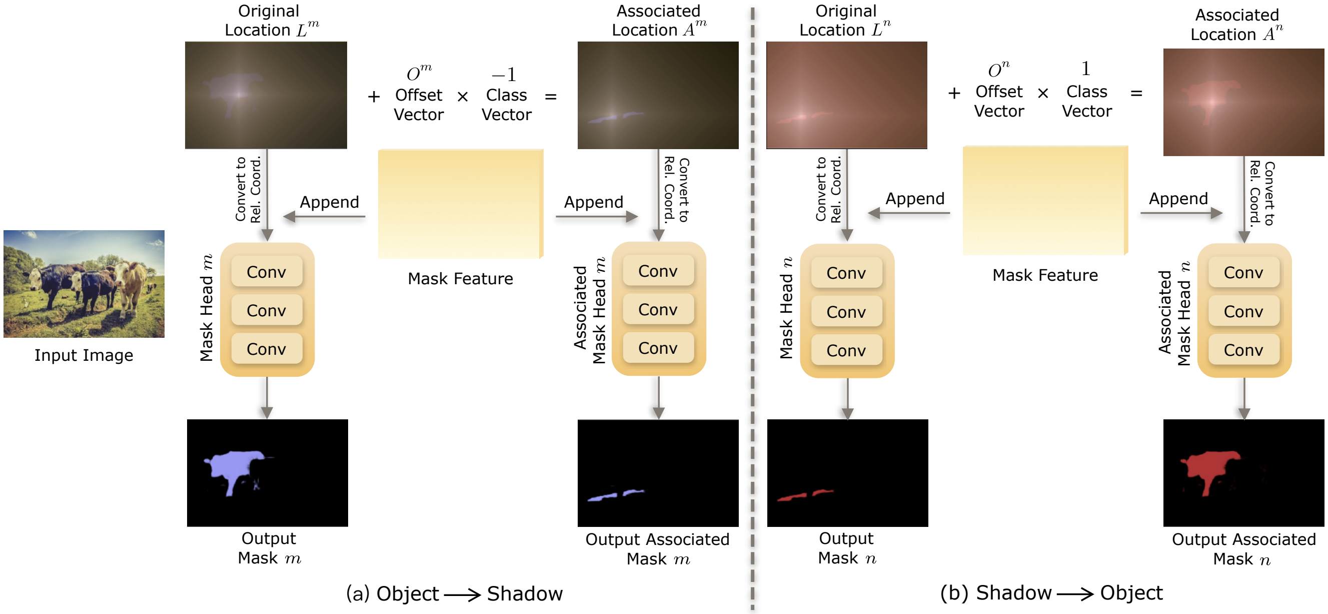


Figure 3: The schematic illustration of the bidirectional relation learning module in our SSIS network. The left part (Object → Shadow) shows how to find the associated shadow instance from the location of the paired object instance while the right part (Shadow → Object) shows how to find the associated object instance from the location of the paired shadow instance.

(iv) **regression** and **centerness**, where regression predicts the bounding box for each shadow and object instance while centerness is used to regularize the prediction by reducing the number of low-quality predicted bounding boxes that are far from the center of a target shadow/object; see [48] for detail.

Next, we formulate a mask branch, which takes the feature map at $P3$ as input and generates the mask feature. We copy and append the mask feature with two relative coordinate (Rel. Coord.) maps, where one relative coordinate map indicates the centers of the object/shadow instances, while another relative coordinate is obtained by first multiplying the offset vectors with a class vector then added the results with the coordinate to represent the centers of the corresponding shadow/object instances. Note that the class vector is generated from the classification score, where -1 indicates the direction from object to shadow, 1 indicates the direction from shadow to object, and the relative coordinate map is computed from the predicted locations of shadow/object instances. Finally, we use the learned filter parameters from the controller and paired controller to perform convolutional operations on the concatenated feature mask and relative coordinate, and predict the masks for the shadow/object instances and the paired object/shadow instances.

In the following subsections, we will elaborate on how to learn the relation between shadow instances and object instances in Section 3.2 and introduce the training and testing strategies of our approach in Section 3.3.

3.2. Bidirectional Relation Learning

Architecture. Figure 3 shows the detailed structure of the proposed bidirectional relation learning module used in our SSIS network. Figure 3 (a) illustrates how to learn the paired shadow instance from the object instance while Figure 3 (b) illustrates this strategy in the opposite direction. As shown in the top left corner, after obtaining the original location L^m of the m-th object instance, we append the location with the mask feature and adopt the m-th mask head to predict the segmentation mask of this instance. Note that the filter parameters in the mask head are produced from the controller and the filter parameters vary in different mask heads; see "Controller" in Figure 2.

Then, we compute the associated location A^m to indicate the center of the paired shadow instance by using the learned offset vector O^m and class vector -1:

$$A^m = L^m + O^m \times -1 \, , \tag{1}$$

where the offset vector is learned from box tower and it represents the distance between the center of the object instance and the center of the paired shadow instance; the class vector is generated from the classification scores and we adopt -1 to represent the direction from object to shadow and use 1 to represent the direction from shadow to object. Next, we concatenate the associated location A^m and mask feature, and use the m-th associated mask head to generate the mask for the shadow instance, and the filter parameters of the associated mask head are learned from the paired controller automatically, as shown in Figure 2.

Similarly, taking the original location L^n of the shadow instance as the input, we compute the associated location A^n of the paired object instance by:

$$A^n = L^n + O^n \times 1 , \tag{2}$$

where O^n denotes the n-th offset vector and 1 represents the learning direction is from shadow to object. Also, we adopt the mask head and the associated mask head to generate the segmentation masks for the paired shadow and object instances, as shown in the right part of Figure 3.

Note that the location maps (L^m, A^m, L^n, A^n) shown in Figure 3 are the visualization results of the learned locations, demonstrating that our network can successfully learn the locations for shadow and object pairs.

Loss function. We define the overall loss function $\mathcal{L}_{\text{all}}$ of our SISS network as:

$$\begin{aligned} \mathcal{L}_{\text{all}} = \mathcal{L}_{\text{cls}} + \mathcal{L}_{\text{center}} + \mathcal{L}_{\text{box}} + \varphi \mathcal{L}_{\text{offset}} \\ + \lambda \mathcal{L}_{\text{mask}} + \lambda \mathcal{L}_{\text{mask}}^{\text{associated}} , \end{aligned} \tag{3}$$

where classification loss $\mathcal{L}_{\text{cls}}$, centerness loss $\mathcal{L}_{\text{center}}$, box regression loss $\mathcal{L}_{\text{box}}$ are same as the losses used in [48]. We adopted the dice loss [34] to compute the losses of the output instance masks $\mathcal{L}_{\text{mask}}$ and output associated instance mask $\mathcal{L}_{\text{mask}}^{\text{associated}}$; see Figure 2 for the predictions. The offset loss $\mathcal{L}_{\text{offset}}$ has the format of the smooth $\mathcal{L}_1$ loss [10] and it is used to optimize the offset vectors:

$$\mathcal{L}_{\text{offset}}(u, v) = \sum_{i \in \{x,y\}} \begin{cases} 0.5 \left(u_i - v_i\right)^2 , & \text{if } |u_i - v_i| < 1; \\ |u_i - v_i| - 0.5 , & \text{otherwise} , \end{cases} \tag{4}$$

where u_i is the product of the predicted offset vector and class vector:

$$u_i = O_i \times C_i , \tag{5}$$

and v_i denotes the ground truth offset vector:

$$v_i = \tilde{L}_i - L_i , \tag{6}$$

where $\tilde{L}_i$ and L_i are the ground truth location of shadow/object instance and the predicted location of the paired object/shadow instance, respectively. The hyper-parameters λ and φ are empirically set as one and 0.1 to balance the weights of different losses. Note that except for the offset loss $\mathcal{L}_{\text{offset}}$, the mask loss of the associated instance mask $\mathcal{L}_{\text{mask}}^{\text{associated}}$ also propagates the gradient to offset vectors, which helps to optimize the network during the training process.

3.3. Training and Testing Strategies

Training parameters. We trained our SSIS network by following the training strategies of CondInst [47] and AdelaiDet [46]. First, we adopted the weights of ResNeXt-101-BiFPN [57, 44] trained on ImageNet [7] to initialize the parameters of the backbone network, set the mini-batch size as two, and optimized our network on two NVIDIA TITAN Xp GPUs (one image per GPU). Second, we set the base learning rate as $1e-3$, adopted a warm-up [11] strategy to linearly increase the learning rate from $1e-4$ to $1e-3$ in the first $1,000$ iterations, and dropped the learning rate to $1e-4$ at $40,000$ iterations, and stopped the learning after $45,000$ iterations. Third, we re-scaled the input images, such that the longer side was less than $1,333$ and the shorter side was less than 640 without changing the image aspect ratio. Lastly, we randomly and horizontally flipped the images for data augmentation.

Inference. In testing, the mask heads in our SSIS network produce the masks for shadow and object instances while the associated mask heads generate the masks for the paired object and shadow instances based on the learned offset vectors; see Figure 3. Since we design a bidirectional relation learning module in our network, for each pair of shadow and object instances, we obtain two sets of predicted masks: (i) if the main branch (the left branches in Figure 3 (a)&(b)) produces the mask of its shadow instance, the associated branch (the right branches in Figure 3 (a)&(b)) will generate the mask of its object instance; (ii) if the main branch produces the mask of its object instance, the associated branch will generate the mask of its shadow instance. However, the accuracy of mask predictions in the main branch is usually better than the predictions in the associated branch, since the associated branch needs to learn both tasks of mask prediction and shadow-object relation, making the training process difficult. Hence, we only adopt the predictions of the associated branch to find the paired relation of shadow and object instances, and take the masks predicted from the main branch as the results. Finally, we adopt mask non-maximum suppression (NMS) to refine the results.

Discussion. Our SSIS has strong ability to find shadow and object locations, but it is infeasible to find instances in some extreme scenarios, in which we cannot find another set of masks, e.g., very small shadow regions. In our implementation, we ignore instances that contain only one set of masks. In practice, this situation is very rare.

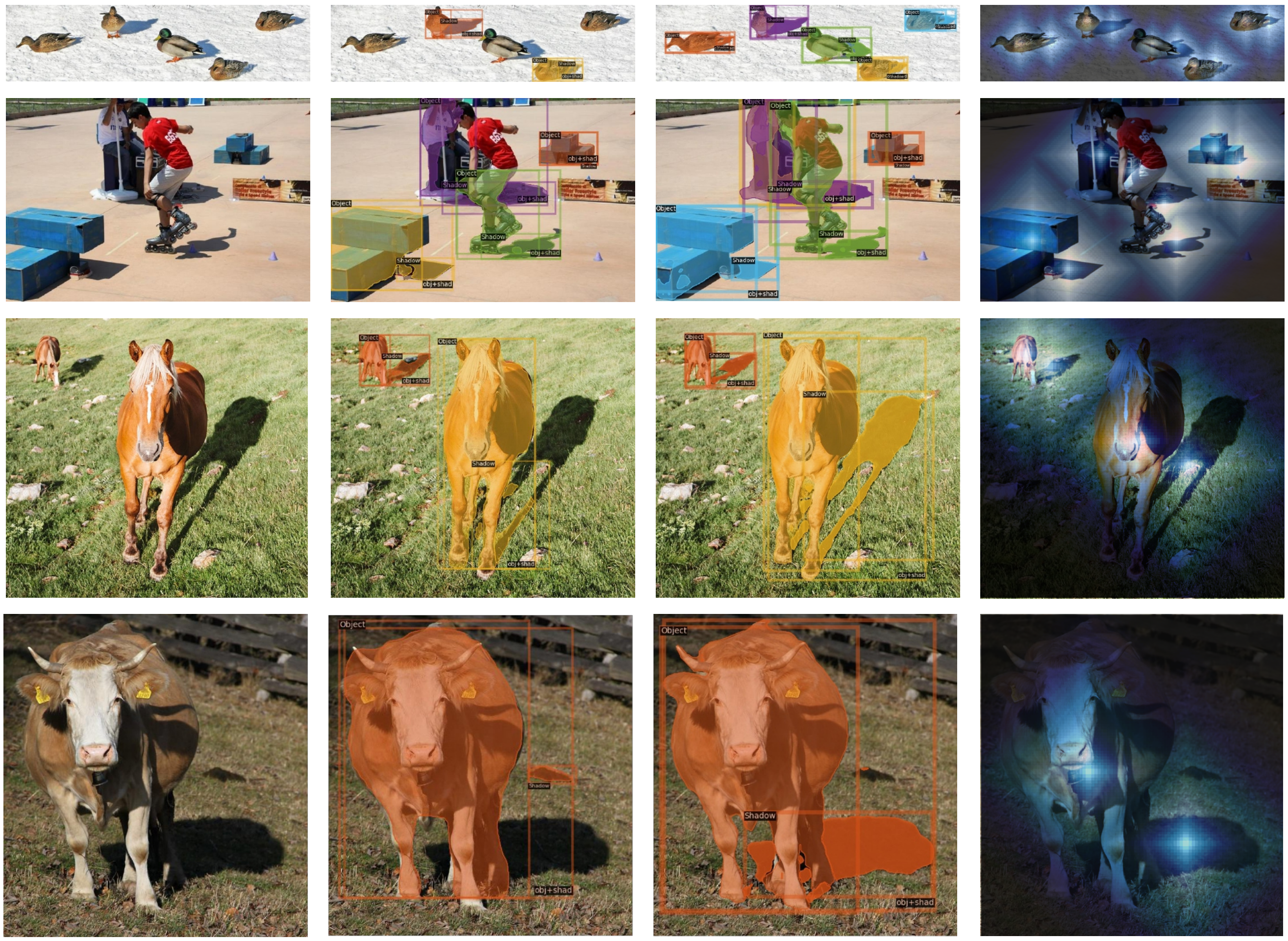
4. Experimental Results

4.1. Dataset and Evaluation Metrics

Benchmark dataset. SOBA [54], named after Shadow OBject Association dataset, is a dataset used for instance shadow detection, which contains $1,000$ images collected from the ADE20K [63, 64], SBU [15, 50, 52], ISTD [53], Microsoft COCO [30], and Internet. In this dataset, there are $3,623$ shadow-object pairs with the labeled masks of shadow instances, object instances, and shadow-object associations. The training set of SOBA includes 840 images with $2,999$ shadow-object pairs and the testing set includes

Table 1: Comparison with the previous state-of-the-art method (LISA) for instance shadow detection.

Network	$SOAP_{segm}$	$SOAP_{bbox}$	Association AP_{segm}	Association AP_{bbox}	Instance AP_{segm}	Instance AP_{bbox}
LISA [54]	21.2	21.7	40.8	49.0	37.0	38.1
SSIS (ours)	**27.4** (29.25%)	**25.5** (17.51%)	**50.5** (23.77%)	**56.2** (14.69%)	**40.3** (8.92%)	**39.6** (3.94%)



(a) input images (b) LISA [54] (c) SSIS (ours) (d) paired locations learned in SSIS

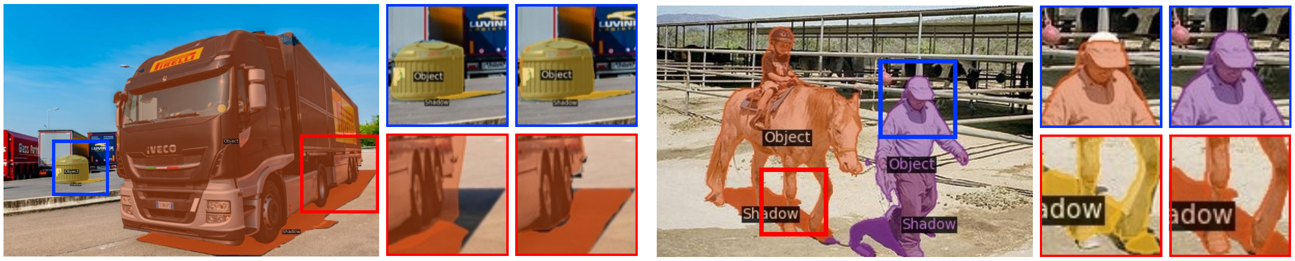
Figure 4: Visual comparison of instance shadow detection results produced by different methods and our learned locations for pairing shadow and object instances.

160 images with 624 shadow-object pairs. We adopt this training set to train our method and evaluate the trained model on the testing set.

Evaluation metrics. SOAP [54], named after Shadow-Object Average Precision, is a metric to evaluate the performance of instance shadow detection. It computes the average precision (AP) with the intersection over union (IoU) and considers a sample as true positive when the IoU between the predicted and ground-truth shadow instances, object instances, and shadow-object associations are all no less than a threshold τ. By setting τ as 0.5 or 0.75, and the average over multiple τ [0.5:0.05:0.95], we can report SOAP$_{50}$, SOAP$_{75}$, and SOAP. Except for SOAP, we further report the average precision over the thresholds [0.5:0.05:0.95] for shadow/object instances, and shadow-object associations, respectively. Finally, we report the evaluation metrics in terms of both bounding boxes and masks.

4.2. Comparison with the State-of-the-art Method

We compare our method with the previous state-of-the-art method, *i.e.*, LISA [54], named after Light-guided Instance Shadow-object Association framework. It adopts the light direction as the guidance in a two-stage object detector to predict the shadow/object instances and shadow-object associations, and leverages a post-processing strat-



SSIS (ours) LISA [54] SSIS (ours) SSIS (ours) LISA [54] SSIS (ours)

Figure 5: More visual comparison results on instance shadow detection, where our method generates higher quality masks on the details of shadow and object instances.

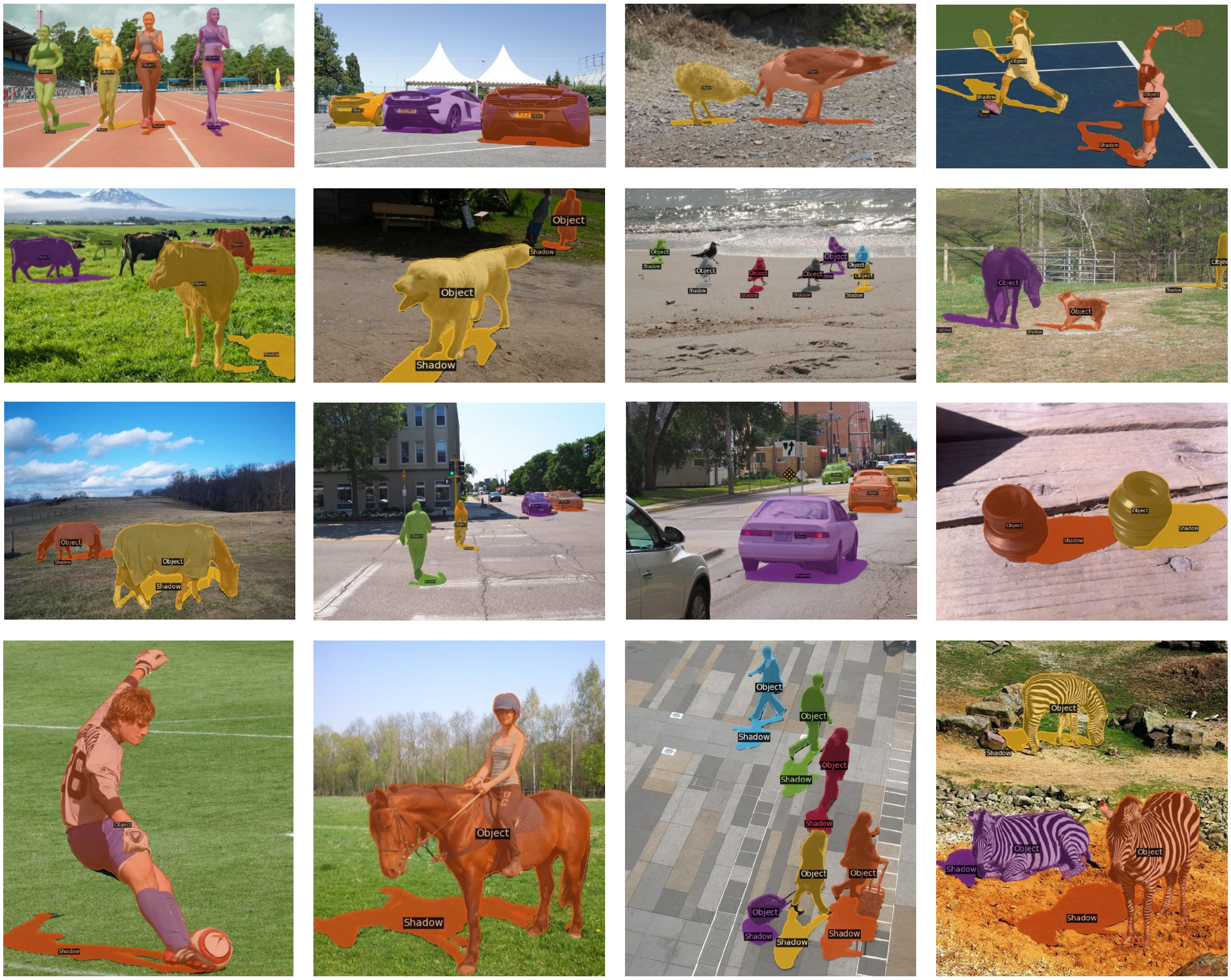


Figure 6: More instance shadow detection results produced by our SSIS over a wide variety of objects and shadows.

egy to pair up the predicted instances and associations. For a fair comparison, we downloaded the results of LISA from the website, which are provided by the authors.

Table 1 reports the comparison results, where we can see that our method clearly outperforms LISA for all the evaluation metrics. Our method achieves a large improvement compared with the previous state-of-the-art method, where the improvements on $SOAP_{segm}$ and $SOAP_{bbox}$ are 29.25% and 17.51%, respectively, showing the supervisory of our method. Moreover, the improvements on Association APs are obvious, demonstrating that our SSIS network can successfully discover the relation between shadow instances

Table 2: Component analysis.

Network	$SOAP_{segm}$	$SOAP_{bbox}$	Association AP_{segm}	Association AP_{bbox}	Instance AP_{segm}	Instance AP_{bbox}
basic	25.5	24.4	47.4	53.7	38.4	38.7
+ offset	25.4	25.1	48.9	55.5	39.3	39.5
+ class	**27.4**	**25.5**	**50.5**	**56.2**	**40.3**	**39.6**

and object instances.

We further provide visual comparison results in Figure 4, where (a) shows the input images, (b) and (c) show the results produced by LISA and our SSIS, and (d) shows the paired locations learned by our method to indicate the paired shadow and object instances. From the results, we can see that (i) our method can discover more shadow-object association pairs, as shown in the first row; (ii) our method can produce more accurate masks for shadow and object instances, as shown in the second and third rows; (iii) our method can successfully pair up the object and shadow instances, but LISA may fail, as shown in the last row. (iv) our method can learn the locations of shadow-object pairs through our directional relation learning module, as shown in (d). Figure 5 illustrates more visual comparison results on instance shadow detection, where we can see that comparing with LISA, our method generates higher quality masks on the details of shadow and object instances. Please see Figure 6 for more instance shadow detection results produced by our SSIS on various types of objects and shadows. The source code, trained model, and detection results are publicly available at https://github.com/stevewongv/SSIS.

4.3. Evaluation on the Network Design

Component analysis. We perform an experiment to evaluate each component in our network design. Here, we consider two baseline networks. "basic" is a network built by removing the offset vectors and class vectors from our SSIS network and adopting only the segmentation loss to optimize the network. "+ offset" learns the offset vectors but ignores the class vectors that use to indicate learning directions. "+ class" is our full pipeline, which further considers the class vectors. Table 2 reports the comparison results, showing that using offset vectors to learn the locations of paired shadow and object instances gives obvious improvements to the association APs. Moreover, considering the class vectors to indicate learning directions can improve the performance in terms of all the metrics.

Bidirectional learning strategy analysis. To evaluate the effectiveness of the bidirectional learning strategy, we perform the experiments to learn the shadow-object pairs in one direction. As shown in Table 3, for "object → shadow", we use the architecture in Figure 3 (a) to predict the masks of object instances from the mask heads in the main branch, and to predict the masks of shadow instances from the associated heads. "shadow → object" leverages the architecture in Figure 3 (b) for the mask prediction. "main + associated" means that we use the masks predicted from the main branch and the corresponding associated branch without using the strategy in Section 3.3-Inference. From the results, we can see that learning the relations of shadows and objects from two directions with our inference strategy achieves the best performance for instance shadow detection.

Table 3: Evaluation on the bidirectional learning strategy.

Network	$SOAP_{segm}$	$SOAP_{bbox}$
object → shadow	20.8	19.3
shadow → object	23.9	21.4
main + associated	21.1	22.3
SSIS	**27.4**	**25.5**

5. Conclusion

This paper presents a new single-stage fully-convolutional network with a bidirectional relation learning module for instance shadow detection. Our key idea is to directly learn the relation between shadow instances and object instances by optimizing the network in an end-to-end manner. To achieve this, we formulate the bidirectional relation learning module, by which we learn offset vectors to indicate the relative locations from shadows to objects and from objects to shadows. Moreover, we design a class vector to indicate the learning directions, and present the offset loss and segmentation loss to jointly optimize the network. In the end, we evaluate our method on the benchmark SOBA dataset, compare it with the best existing method for instance shadow detection, and show the superiority of our method, both qualitatively and quantitatively. In the future, we plan to improve the performance of our network by exploring the knowledge from the existing data prepared for other vision tasks, *e.g.*, shadow detection and instance segmentation, from the synthetic data generated by computer graphic techniques, and from the unlabeled data downloaded from the Internet.

Acknowledgments. This work was supported by the CUHK Direct Grant for Research 2018/19 and the Hong Kong Research Grants Council GRF Project (CUHK 14201620). We thank Google Colab for providing GPUs and Tianyu Wang thanks the inimitable care and support from Zhouxiang Xiang and her family.

References

[1] Kai Chen, Jiangmiao Pang, Jiaqi Wang, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jianping Shi, Wanli Ouyang, et al. Hybrid task cascade for instance segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 4974–4983, 2019. 3

[2] Liang-Chieh Chen, Alexander Hermans, George Papandreou, Florian Schroff, Peng Wang, and Hartwig Adam. MaskLab: Instance segmentation by refining object detection with semantic and direction features. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 4013–4022, 2018. 3

[3] Xinlei Chen, Ross Girshick, Kaiming He, and Piotr Dollár. TensorMask: A foundation for dense object segmentation. *arXiv preprint arXiv:1903.12174*, 2019. 3

[4] Zhihao Chen, Lei Zhu, Liang Wan, Song Wang, Wei Feng, and Pheng-Ann Heng. A multi-task mean teacher for semi-supervised shadow detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 5611–5620, 2020. 2

[5] Xiaodong Cun, Chi-Man Pun, and Cheng Shi. Towards ghost-free shadow removal via dual hierarchical aggregation network and shadow matting GAN. In *AAAI Conference on Artificial Intelligence*, pages 10680–10687, 2020. 2

[6] Jifeng Dai, Kaiming He, and Jian Sun. Instance-aware semantic segmentation via multi-task network cascades. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3150–3158, 2016. 3

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Fei-Fei Li. ImageNet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. 5

[8] Bin Ding, Chengjiang Long, Ling Zhang, and Chunxia Xiao. ARGAN: Attentive recurrent generative adversarial network for shadow detection and removal. In *IEEE International Conference on Computer Vision*, pages 10213–10222, 2019. 2

[9] Naiyu Gao, Yanhu Shan, Yupei Wang, Xin Zhao, Yinan Yu, Ming Yang, and Kaiqi Huang. SSAP: Single-shot instance segmentation with affinity pyramid. In *IEEE International Conference on Computer Vision*, pages 642–651, 2019. 3

[10] Ross Girshick. Fast R-CNN. In *IEEE International Conference on Computer Vision*, pages 1440–1448, 2015. 5

[11] Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch SGD: Training ImageNet in 1 hour. *arXiv preprint arXiv:1706.02677*, 2017. 5

[12] Ruiqi Guo, Qieyun Dai, and Derek Hoiem. Single-image shadow detection and removal using paired regions. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2033–2040, 2011. 2

[13] Zeeshan Hayder, Xuming He, and Mathieu Salzmann. Boundary-aware instance segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 5696–5704, 2017. 3

[14] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask R-CNN. In *IEEE International Conference on Computer Vision*, pages 2961–2969, 2017. 3

[15] Le Hou, Tomás F. Yago Vicente, Minh Hoai, and Dimitris Samaras. Large scale shadow annotation and detection using lazy annotation and stacked CNNs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019. to appear. 2, 5

[16] Xiaowei Hu, Chi-Wing Fu, Lei Zhu, Jing Qin, and Pheng-Ann Heng. Direction-aware spatial context features for shadow detection and removal. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(11):2795–2808, 2020. 2

[17] Xiaowei Hu, Yitong Jiang, Chi-Wing Fu, and Pheng-Ann Heng. Mask-ShadowGAN: Learning to remove shadows from unpaired data. In *IEEE International Conference on Computer Vision*, pages 2472–2481, 2019. 2

[18] Xiaowei Hu, Tianyu Wang, Chi-Wing Fu, Yitong Jiang, Qiong Wang, and Pheng-Ann Heng. Revisiting shadow detection: A new benchmark dataset for complex world. *IEEE Transactions on Image Processing*, 30:1925–1934, 2021. 2

[19] Xiaowei Hu, Lei Zhu, Chi-Wing Fu, Jing Qin, and Pheng-Ann Heng. Direction-aware spatial context features for shadow detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 7454–7462, 2018. 2

[20] Xiang Huang, Gang Hua, Jack Tumblin, and Lance Williams. What characterizes a shadow boundary under the sun and sky? In *IEEE International Conference on Computer Vision*, pages 898–905, 2011. 2

[21] Salman Hameed Khan, Mohammed Bennamoun, Ferdous Sohel, and Roberto Togneri. Automatic feature learning for robust shadow detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1939–1946, 2014. 2

[22] Salman Hameed Khan, Mohammed Bennamoun, Ferdous Sohel, and Roberto Togneri. Automatic shadow detection and removal from a single image. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(3):431–446, 2016. 2

[23] Jean-François Lalonde, Alexei A. Efros, and Srinivasa G. Narasimhan. Detecting ground shadows in outdoor consumer photographs. In *European Conference on Computer Vision*, pages 322–335, 2010. 2

[24] Hieu Le and Dimitris Samaras. Shadow removal via shadow image decomposition. In *IEEE International Conference on Computer Vision*, pages 8578–8587, 2019. 2

[25] Hieu Le and Dimitris Samaras. From shadow segmentation to shadow removal. In *European Conference on Computer Vision*, 2020. to appear. 2

[26] Hieu Le, Tomás F. Yago Vicente, Vu Nguyen, Minh Hoai, and Dimitris Samaras. A+D Net: Training a shadow detector with adversarial shadow attenuation. In *European Conference on Computer Vision*, pages 662–678, 2018. 2

[27] Youngwan Lee and Jongyoul Park. CenterMask: Real-time anchor-free instance segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 13906–13915, 2020. 3

[28] Yi Li, Haozhi Qi, Jifeng Dai, Xiangyang Ji, and Yichen Wei. Fully convolutional instance-aware semantic segmentation.

In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2359–2367, 2017. 3

[29] Tsung-Yi Lin, Piotr Dollar, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2117–2125, 2017. 3

[30] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common objects in context. In *European Conference on Computer Vision*, pages 740–755, 2014. 5

[31] Yun-Hsuan Lin, Wen-Chin Chen, and Yung-Yu Chuang. BEDSR-Net: A deep shadow removal network from a single document image. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 12905–12914, 2020. 2

[32] Daquan Liu, Chengjiang Long, Hongpan Zhang, Hanning Yu, Xinzhi Dong, and Chunxia Xiao. ARShadowGAN: Shadow generative adversarial network for augmented reality in single light scenes. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 8139–8148, 2020. 2

[33] Shu Liu, Lu Qi, Haifang Qin, Jianping Shi, and Jiaya Jia. Path aggregation network for instance segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 8759–8768, 2018. 3

[34] Fausto Milletari, Nassir Navab, and Seyed-Ahmad Ahmadi. V-Net: Fully convolutional neural networks for volumetric medical image segmentation. In *International Conference on 3D Vision (3DV)*, pages 565–571, 2016. 5

[35] Alejandro Newell and Jia Deng. Pixels to graphs by associative embedding. In *Conference on Neural Information Processing Systems*, pages 2172–2181, 2017. 3

[36] Vu Nguyen, Tomás F. Yago Vicente, Maozheng Zhao, Minh Hoai, and Dimitris Samaras. Shadow detection with conditional generative adversarial networks. In *IEEE International Conference on Computer Vision*, pages 4510–4518, 2017. 2

[37] Alexandros Panagopoulos, Chaohui Wang, Dimitris Samaras, and Nikos Paragios. Illumination estimation and cast shadow detection through a higher-order graphical model. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 673–680, 2011. 2

[38] Chao Peng, Tete Xiao, Zeming Li, Yuning Jiang, Xiangyu Zhang, Kai Jia, Gang Yu, and Jian Sun. MegDet: A large mini-batch object detector. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 6181–6189, 2018. 3

[39] Pedro O. Pinheiro, Ronan Collobert, and Piotr Dollár. Learning to segment object candidates. In *Conference on Neural Information Processing Systems*, pages 1990–1998, 2015. 3

[40] Pedro O Pinheiro, Tsung-Yi Lin, Ronan Collobert, and Piotr Dollár. Learning to refine object segments. In *European Conference on Computer Vision*, pages 75–91, 2016. 3

[41] Liangqiong Qu, Jiandong Tian, Shengfeng He, Yandong Tang, and Rynson W.H. Lau. DeshadowNet: A multi-context embedding deep network for shadow removal. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 4067–4075, 2017. 2

[42] Elena Salvador, Andrea Cavallaro, and Touradj Ebrahimi. Cast shadow segmentation using invariant color features. *Computer Vision and Image Understanding*, 95(2):238–259, 2004. 2

[43] Li Shen, Teck Wee Chua, and Karianto Leman. Shadow optimization from structured deep edge detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2067–2074, 2015. 2

[44] Mingxing Tan, Ruoming Pang, and Quoc V. Le. EfficientDet: Scalable and efficient object detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 10781–10790, 2020. 5

[45] Jiandong Tian, Xiaojun Qi, Liangqiong Qu, and Yandong Tang. New spectrum ratio properties and features for shadow detection. *Pattern Recognition*, 51:85–96, 2016. 2

[46] Zhi Tian, Hao Chen, Xinlong Wang, Yuliang Liu, and Chunhua Shen. AdelaiDet: A toolbox for instance-level recognition tasks. https://git.io/adelaidet, 2019. 5

[47] Zhi Tian, Chunhua Shen, and Hao Chen. Conditional convolutions for instance segmentation. In *European Conference on Computer Vision*, 2020. to appear. 3, 5

[48] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. FCOS: A simple and strong anchor-free object detector. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020. to appear. 4, 5

[49] Tomás F. Yago Vicente, Minh Hoai, and Dimitris Samaras. Leave-one-out kernel optimization for shadow detection. In *IEEE International Conference on Computer Vision*, pages 3388–3396, 2015. 2

[50] Tomás F. Yago Vicente, Minh Hoai, and Dimitris Samaras. Noisy label recovery for shadow detection in unfamiliar domains. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3783–3792, 2016. 5

[51] Tomás F. Yago Vicente, Minh Hoai, and Dimitris Samaras. Leave-one-out kernel optimization for shadow detection and removal. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(3):682–695, 2018. 2

[52] Tomás F. Yago Vicente, Le Hou, Chen-Ping Yu, Minh Hoai, and Dimitris Samaras. Large-scale training of shadow detectors with noisily-annotated shadow examples. In *European Conference on Computer Vision*, pages 816–832, 2016. 2, 5

[53] Jifeng Wang, Xiang Li, and Jian Yang. Stacked conditional generative adversarial networks for jointly learning shadow detection and shadow removal. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1788–1797, 2018. 2, 5

[54] Tianyu Wang*, Xiaowei Hu*, Qiong Wang, Pheng-Ann Heng, and Chi-Wing Fu. Instance shadow detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1880–1889, 2020. * Joint first authors. 1, 2, 5, 6, 7

[55] Xinlong Wang, Tao Kong, Chunhua Shen, Yuning Jiang, and Lei Li. SOLO: Segmenting objects by locations. *arXiv preprint arXiv:1912.04488*, 2019. 3

[56] Xinlong Wang, Rufeng Zhang, Tao Kong, Lei Li, and Chunhua Shen. SOLOv2: Dynamic, faster and stronger. *arXiv preprint arXiv:2003.10152*, 2020. 3

[57] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1492–1500, 2017. 5

[58] Hui Ying, Zhaojin Huang, Shu Liu, Tianjia Shao, and Kun Zhou. EmbedMask: Embedding coupling for one-stage instance segmentation. *arXiv preprint arXiv:1912.01954*, 2019. 3

[59] Ji Zhang, Yannis Kalantidis, Marcus Rohrbach, Manohar Paluri, Ahmed Elgammal, and Mohamed Elhoseiny. Large-scale visual relationship understanding. In *AAAI Conference on Artificial Intelligence*, volume 33, pages 9185–9194, 2019. 3

[60] Ling Zhang, Chengjiang Long, Xiaolong Zhang, and Chunxia Xiao. RIS-GAN: explore residual and illumination with generative adversarial networks for shadow removal. In *AAAI Conference on Artificial Intelligence*, pages 12829–12836, 2020. 2

[61] Xuaner Cecilia Zhang, Jonathan T. Barron, Yun-Ta Tsai, Rohit Pandey, Xiuming Zhang, Ren Ng, and David E. Jacobs. Portrait shadow manipulation. *ACM Transactions on Graphics (SIGGRAPH)*, 39(4):78, 2020. 2

[62] Quanlong Zheng, Xiaotian Qiao, Ying Cao, and Rynson W.H. Lau. Distraction-aware shadow detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 5167–5176, 2019. 2

[63] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ADE20K dataset. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 633–641, 2017. 5

[64] Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ADE20K dataset. *International Journal of Computer Vision*, 127(3):302–321, 2019. 5

[65] Jiejie Zhu, Kegan G.G. Samuel, Syed Z. Masood, and Marshall F. Tappen. Learning to recognize shadows in monochromatic natural images. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 223–230, 2010. 2

[66] Lei Zhu, Zijun Deng, Xiaowei Hu, Chi-Wing Fu, Xuemiao Xu, Jing Qin, and Pheng-Ann Heng. Bidirectional feature pyramid network with recurrent attention residual modules for shadow detection. In *European Conference on Computer Vision*, pages 121–136, 2018. 2