This ECCV 2018 paper, provided here by the Computer Vision Foundation, is the author-created version.
The content of this paper is identical to the content of the officially published ECCV 2018
LNCS version of the paper as available on SpringerLink: https://link.springer.com/conference/eccv

Bidirectional Feature Pyramid Network with Recurrent Attention Residual Modules for Shadow Detection

Lei Zhu[1,2,5,⋆], Zijun Deng[3,⋆], Xiaowei Hu[1], Chi-Wing Fu[1,5,⋆⋆], Xuemiao Xu[4], Jing Qin[2], and Pheng-Ann Heng[1,5]

[1] The Chinese University of Hong Kong, Hong Kong, China
[2] The Hong Kong Polytechnic University, Hong Kong, China
[3] South China University of Technology, Guangzhou, China
[4] Guangdong Provincial Key Lab of Computational Intelligence and Cyberspace Information (South China University of Technology), Guangzhou, China
[5] Shenzhen Key Laboratory of Virtual Reality and Human Interaction Technology, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, China

Abstract. This paper presents a network to detect shadows by exploring and combining global context in deep layers and local context in shallow layers of a deep convolutional neural network (CNN). There are two technical contributions in our network design. First, we formulate the *recurrent attention residual* (RAR) module to combine the contexts in two adjacent CNN layers and learn an attention map to select a residual and then refine the context features. Second, we develop a bidirectional feature pyramid network (BFPN) to aggregate shadow contexts spanned across different CNN layers by deploying two series of RAR modules in the network to iteratively combine and refine context features: one series to refine context features from deep to shallow layers, and another series from shallow to deep layers. Hence, we can better suppress false detections and enhance shadow details at the same time. We evaluate our network on two common shadow detection benchmark datasets: SBU and UCF. Experimental results show that our network outperforms the best existing method with 34.88% reduction on SBU and 34.57% reduction on UCF for the balance error rate.

1 Introduction

Shadows are regions that receive less illumination than the surroundings, due to lights occluded by associated objects in the scene. To detect shadows in images, early works develop physical models with heuristic priors [1, 2] or take a machine learning approach based on hand-crafted features. However, image priors and hand-crafted features are not effective for extracting high-level semantics.

More recently, methods based on the convolutional neural network (CNN) [3–7] show distinct performance on various shadow detection benchmarks, e.g., [4,

⋆ Joint first authors
⋆⋆ Corresponding author (cwfu@cse.cuhk.edu.hk)

8]. A key factor for the successes is that CNN is able to learn the global spatial contexts in shadow images, as demonstrated in very recent works [5–7].

To further explore the spatial contexts and improve the shadow detection performance, it requires an understanding of the *global contexts* about the objects and illumination conditions in the scene, the *local contexts* about the details in the shadow shapes, as well as an *integration of various contexts* extracted in different scales. This drives us to explore shadow contexts over different CNN layers, where the shallow layers help reveal local contexts and the deep layers help reveal the global contexts due to a large receptive field.

In this work, we design a *bidirectional feature pyramid network* (BFPN), which extends over the feature pyramid network architecture [9]. Particularly, we aim to leverage the spatial contexts across deep and shallow layers, as well as iteratively integrate the contexts for maximized shadow detection performance. In detail, we have the following technical contributions in this work:

- First, we develop the *recurrent attention residual module*, or RAR module for short, to combine and process spatial contexts in two adjacent CNN layers. Inside the module, an attention map is learnt and predicted by the network to select a residual and to refine the context features.
- Second, we design our *bidirectional feature pyramid network* (BFPN) by taking the RAR modules as building blocks. Inside the BFPN, we first apply the convolutional neural network (CNN) to generate a set of feature maps (i.e., spatial contexts) in different resolutions, and then use two series of RAR modules to iteratively integrate spatial contexts over the CNN layers: one series of RAR modules from deep to shallow layers and another series from shallow to deep layers. Lastly, the prediction results from the two directions are further integrated by means of an attention mechanism.

To demonstrate the performance of our network, we evaluate it on two common benchmarks, i.e., SBU [4] and UCF [8], and compare its performance against several state-of-the-art methods designed for shadow detection, shadow removal, saliency detection and semantic segmentation. Results show that our model clearly outperforms the best existing method with over 34.88% reduction on SBU and 34.57% reduction on UCF in terms of the balance error rate. The code and models of our method are available at https://github.com/zijundeng/BDRAR.

2 Related Work

Shadows in natural images have been employed as hints in various computer vision problems for extracting the scene geometry [10, 11], light direction [12], and camera location and parameters [13]. On the other hand, shadows are also beneficial to a variety of high-level image understanding tasks, e.g., image segmentation [14], object detection [15], and object tracking [16].

In the literature, a number of single-image shadow detection methods have been proposed. Early works [1, 2, 17] focused on illumination models and color

information to detect shadows from inputs, but just worked well for wide dynamic range images [5, 18]. Data-driving statistical learning is another popular strategy for shadow detection by learning shadow properties from images with annotated ground truths. These methods usually began by first designing some hand-crafted features [8, 18–21] and then employing some classifiers [8, 18–21] for shadow detection. While showing the performance improvement on the shadow detection, they often failed in complex cases, due to the limited discriminative capability of the hand-crafted features.

Compared with traditional methods based on hand-crafted features, deep convolutional neural network (CNN) based methods have refreshed many computer vision tasks [6, 7, 9, 22, 23], including shadow detection. For instance, Khan et al. [3] was the first one to use deep learning to automatically learn features for shadow detection with a significant improvement. They trained one CNN to detect shadow regions and another CNN to detect shadow boundaries, and then fed the prediction results to a conditional random field (CRF) for classifying image pixels as shadows/non-shadows. Later, a stacked CNN [4] was presented to detect shadows by considering the global prediction of an image and the shadow predictions of image patches. They first trained a fully convolutional network to obtain an image-level shadow prior, which was combined with local image patches to train a patch-based CNN for the final shadow map prediction.

Recently, a fast deep shadow detection network [24] was introduced by obtaining a shadow prior map produced from hand-crafted features and then applying a patch level CNN to compute the improved shadow probability map of the input image. And a generative adversarial network based shadow detector, called scGAN [5], was developed by formulating a conditional generator on input RGB images and learning to predict the corresponding shadow maps. When detecting shadows for a given image, they combined the predicted shadow masks for a large quantity of multi-scale crops for the final shadow mask prediction. The most recent work by Hu et al. [6, 7] presented a deep network with direction-aware spatial context modules to analyze the global semantics.

The deep models in state-of-the-art works [5–7] mainly emphasized the importance of inferring global contexts for shadow detection. Compared to these methods, we suggest to develop a network by fully leveraging the global and local contexts in different layers of the CNN to detect shadows. Results show that our method clearly outperforms [5–7] in terms of the BER values on two widely-used benchmark datasets.

3 Methodology

Fig. 1 presents the workflow of the overall shadow detection network that employs two series of RAR modules (see Fig. 2 (d)) to fully exploit the global contexts and the local contexts at two adjacent layers of the convolutional neural network. Our network takes a single image as input and outputs the shadow detection result in an end-to-end manner. First, it leverages a convolutional neural network (CNN) to extract the feature maps with different resolutions. The

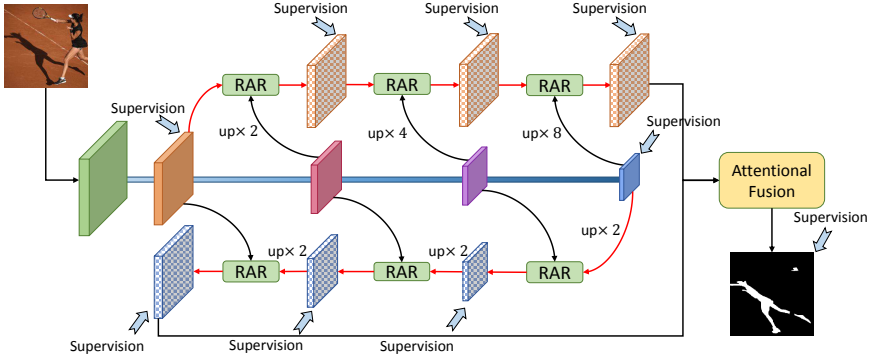


Fig. 1: The schematic illustration of the overall shadow detection network. Best viewed in color.

feature maps in shallow layers discover the fine shadow detail information in the local regions while the feature maps in deep layers capture shadow semantic information of the whole image [25]. Then, we develop RAR modules to progressively refine features at each layer of the CNN by taking two adjacent feature maps as inputs to learn an attention map and to select a residual for the refinement of context features. We embed multiple RAR modules into a bidirectional feature pyramid network (BFPN), which uses two directional pathways to harvest the context information at different layers: one pathway is from shallow layers to deep layers, while another pathway is in the opposite direction. Lastly, we predict the score maps from the features at the last layers of two directional pathways, and then fuse those two score maps in an attentional manner to generate the final shadow detection result.

In the following subsections, we firstly elaborate how the RAR module refine the feature maps at each layer of the CNN, then present the details on how we embed our RAR modules into the shadow detection network (called bidirectional feature pyramid network (BFPN) with RAR modules, *BDRAR* for short), and finally introduce the training and testing strategies of our network.

3.1 Recurrent Attention Residual Module

One of the main issues in our method is to refine the context features at each layer for shadow detection by combining the context features at two adjacent layers of the CNN. A common way is to use an element-wise addition (see Fig. 2 (a)) like the original FPN [9] to merge these two adjacent features. It up-samples the low-resolution feature maps and then adds it with the high-resolution feature maps. However, the element-wise addition on the two input context features simply merges the features at different layers, suffering from a limited ability to suppress non-shadow details in the high-resolution feature maps and introducing the non-shadow regions into the results. To alleviate this problem, we introduce

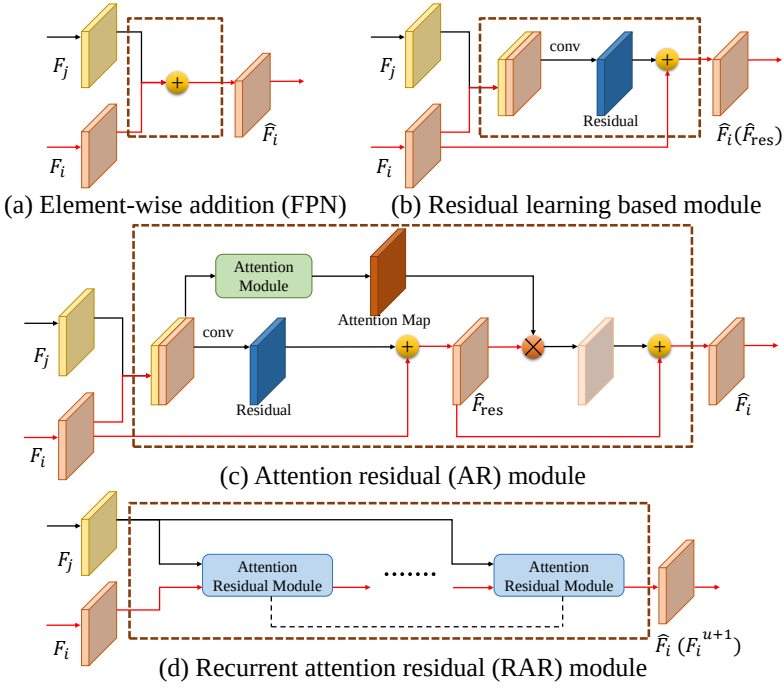
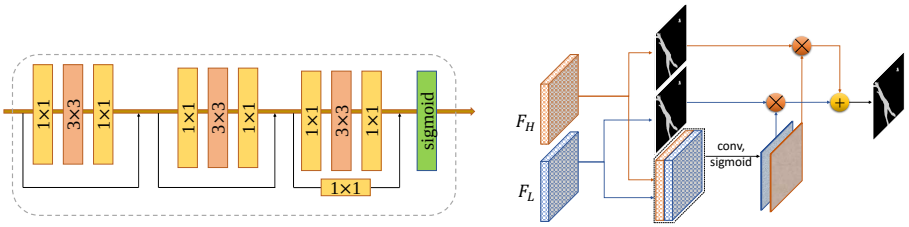


Fig. 2: The schematic illustration of different modules to merge features (F_i and F_j) at two layers for the feature refinement (output refined feature: $\widehat{F}_i$).

the residual learning technique [26, 27] to improve the feature refinement by learning the residual of input features. As shown in Fig. 2 (b), it begins by taking the concatenation of two input feature maps as the inputs and learning to produce the residual maps to refine original features by the element-wise addition. Learning the residual counterpart (see Fig. 2 (b)) instead of adding the feature maps directly (see Fig. 2 (a)) makes the feature refinement task easier, since it only needs to learn the complementary information from the features at different layers and can preserve the original features.

To further improve the performance of feature refinement, we develop a recurrent attention residual (RAR) module (see Fig. 2 (d)), which recurrently applies an attention residual (AR) module (see Fig. 2 (c)) to compute the refined context features. Let $\widehat{F}_{res}$ denote the refined output features produced by using the residual learning based module of Fig. 2 (b). Our AR module improves the feature enhancement performance by recurrently learning an attention map to select the useful information of $\widehat{F}_{res}$ as the residual, which is added by the original $\widehat{F}_{res}$ as the output refined features. Specifically, the AR module starts by concatenating the input two adjacent context features, and then utilizes an attention module (see Fig. 3 (a)) to produce a weight (or an attention) map from the concatenated features. The attention map works as a feature selector



(a) Attention module in RAR (b) Attentional fusion

Fig. 3: (a) The schematic illustration of the attention module in RAR; (b) The details of attentional fusion for the final shadow detection map; see Sec. 3.2.

to enhance good features and suppress noise in $\widehat{F}_{res}$. Then, the output refined feature of AR module is obtained by multiplying the learned attention map with the $\widehat{F}_{res}$, and then adding it with $\widehat{F}_{res}$ using an element-wise addition, as shown in Fig. 3 (c). Hence, our RAR module computes the refined context features by recurrently employing the AR modules, where the output refined features at the previous recurrent step are used as the input of subsequent AR modules, and the parameters of different AR modules are shared to reduce the risk of overfitting.

Mathematically, our RAR computes the refinement features (denoted as F_i^{u+1}) at the layer i as:

$$F_i^{u+1} = \Big(1 + A\big(Cat(F_i^u, F_j)\big)\Big) * \big[\mathbf{\Phi}\big(Cat(F_i^u, F_j)\big) + F_j\big] \ , \tag{1}$$

where u=1,2,..., U; U is the number of recurrent steps; F_i^u is the refined features after u recurrent steps, $F_i^1 = F_i$, which is the context features at layer i of the CNN; F_j is the context features at layer j of the CNN; Cat denotes the concatenation operation on F_i and F_j; $A(Cat(F_i, F_j))$ is the learned weight map using the attention mechanism (see the paragraph below for the details); and $\mathbf{\Phi}$ represents the residual function.

Attention module in the RAR. Motivated by the attention mechanism used for image classification [23], we develop an attention module (see Fig. 3 (a)) to learn a weight map from the concatenated features ($Cat(F_i^u, F_j)$ of Eq. (1)). It starts with three residual blocks, where each block has a 1*1 convolution layer, a 3*3 dilated convolution layer, and a 1*1 convolutional layer. After that, we compute the weight (attention) map $\Big(A\big(Cat(F_i^u, F_j)\big)\Big)$ by using a sigmoid function on the feature maps (denoted as H) learned from three residual blocks:

$$a(p, q, c) = \frac{1}{1 + exp\big(-H(p, q, c)\big)} \ , \tag{2}$$

where $a(p, q, c)$ is the weight at the spatial position (p, q) of the c-th channel of the learned weight map $\Big(A\big(Cat(F_i^u, F_j)\big)\Big)$, while $H(p, q, c)$ is the feature value at the spatial position (p, q) of the c-th channel of H.

3.2 Our Network

Note that the original FPN [9] iteratively merges features in a top-down pathway until reaching the last layer with the largest resolution. We argue that such single top-down pathway is not enough to capture the shadow context information spanned in different layers of the CNN. To alleviate this problem, we design a bidirection mechanism to integrate context information of different layers: one (top-down) pathway is to integrate features from low-resolution layers to high-resolution layers, while another (bottom-up) pathway is from high-resolution layers to low-resolution layers, and we use our RAR module (see Sec. 3.1) to refine features at each layer by merging two adjacent features. After that, we, following [28], use an attention mechanism (see Fig. 3 (b)) to generate the final shadow detection map by fusing the shadow predictions from the refined features (denoted as F_H) at the last layer in the top-down direction and the features (denoted as F_L) at the last layer in the bottom-up direction. As shown in Fig. 3 (b), we first generate two shadow detection maps from the refined features (F_H and F_L) by using a 1×1 convolutional layer. Then, we perform two convolution layers (3×3 and 1×1) on the concatenation of F_H and F_L, and use a sigmoid function to generate attention maps, which are multiplied with the shadow detection maps to produce the final shadow detection result.

The designed bidirection feature pyramid network (BFPN) can effectively use the complementary information of features in two directional pathways for shadow detection. Please refer to the Ablation study in Sec. 4.4 for the comparisons between the original FPN and our BFRN on two shadow detection benchmark datasets.

3.3 Training and Testing Strategies

We implement our network using PyTorch, and adopt ResNeXt101 [29] as the basic convolutional neural network for feature extraction.

Loss function. As shown in Fig. 1, our network utilizes the deep supervision mechanism [30] to impose supervision signals to the features at each layer of two bidirectional pathways to promote useful information propagation to the shadow regions. During the training process, binary cross-entropy loss is used for each output of the network, and the total loss is the summation of the losses of all the output score maps.

Training parameters. To accelerate the training procedure and reduce the overfitting risk, we initialize the parameters of the basic convolutional neural network by ResNeXt [29], which has been well-trained for the image classification task on the ImageNet. Other parameters are initialized by random noise. Stochastic gradient descent (SGD) equipped with momentum of 0.9 and weight decay of 0.0005 is used to optimize the whole network for 3000 iterations. We adjust the learning rate by the poly strategy [31] with the basic learning rate of 0.005 and the power of 0.9. We train our network on the SBU training set, which contains

Table 1: Comparing our method (BDRAR) with state-of-the-arts for shadow detection (DSC [6, 7], scGAN [5], stacked-CNN [4], patched-CNN [24] and Unary-Pairwise [19]), for shadow removal (DeshadowNet [33]), for saliency detection (SRM [34] and Amulet [35]), and for semantic segmentation (PSPNet [36]).

	SBU [4]	UCF [8]
method	BER	BER
BDRAR (ours)	**3.64**	**5.30**
DSC [6, 7]	5.59	8.10
scGAN [5]	9.10	11.50
stacked-CNN [4]	11.00	13.00
patched-CNN [24]	11.56	-
Unary-Pairwise [19]	25.03	-
DeshadowNet [33]	6.96	8.92
SRM [34]	7.25	9.81
Amulet [35]	15.13	15.17
PSPNet [36]	8.57	11.75

4089 images. Moreover, we augment the training set by random horizontal flipping. We resize all the images to the same resolution (416×416). Our network is trained on a single GTX 1080Ti GPU with a mini-batch size of eight, and the whole training process only takes about 40 minutes.

Inference. During testing, we first resize the input images to the same resolution as we used in the training stage. Then, we take the output of the attentional fusion module (see Fig. 3 (b)) as the final output of the whole network for shadow detection. Finally, we use the fully connected conditional random field (CRF) [32] to further enhance the detection results by optimizing the spatial coherence of each pixel on the output of our network.

4 Experimental Results

4.1 Datasets and Evaluation Metrics

Benchmark datasets. We evaluate the effectiveness of the proposed network on two widely-used shadow benchmark datasets: SBU [4] and UCF [8]. Each image in both two benchmark datasets has its corresponding annotated binary shadow mark. The SBU dataset is the largest publicly available annotated shadow dataset with 4089 training images and 638 testing images, while the UCF dataset consists of 145 training images and 76 testing images. In our experiment, we train our shadow detection network using SBU training set, and evaluate our method and competitors on the testing sets of the SBU and UCF. Our network takes 0.056 s to process an image of 416×416 resolution.

Evaluation metrics We employ a commonly-used metric, which is balance

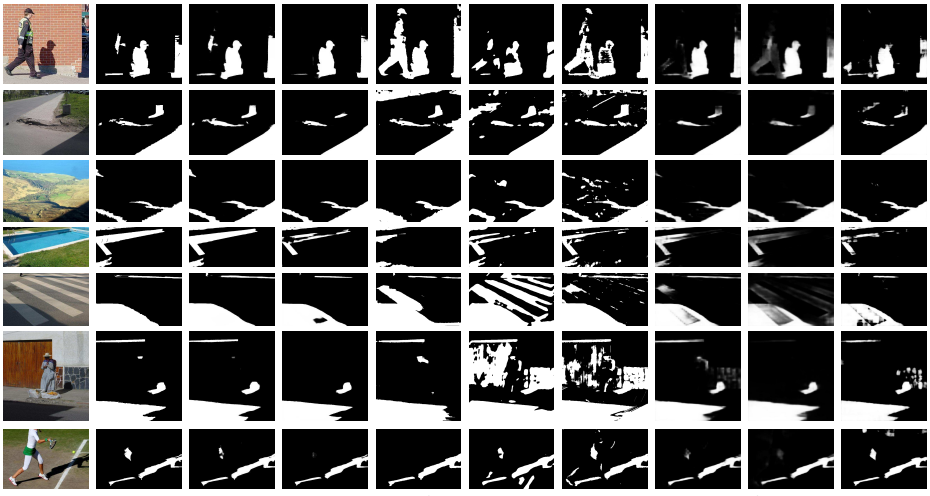


Fig. 4: Visual comparison of shadow maps produced by our method and others (4th-10th columns) against ground truths shown in 2nd column. Note that "stCNN" and "paCNN" stand for "stacked-CNN" and "patched-CNN", respectively.

error rate (BER), to quantitatively evaluate the shadow detection performance; please refer to this work [6] for the definition of the BER. Note that better performance is indicated by a lower BER value.

4.2 Comparison with the State-of-the-art Shadow Detectors

We compare our method with five recent shadow detectors: DSC [6, 7], scGAN [5], stacked-CNN [4], patched-CNN [24] and Unary-Pairwise [19]. To make a fair comparison, we obtain other shadow detectors' results either directly from the authors or by using the public implementations provided by the authors with recommended parameter settings.

Table 1 reports the quantitative results of different methods. From the results, we can find that the deep learning based methods [4, 6, 7, 24] usually have better shadow detection results than hand-crafted detectors [19], since they can learn more powerful features for shadow detection from the annotated training set. DSC [6, 7] achieves a superior performance than other existing deep learning models [4, 5, 24] by analyzing the directional contexts to understand the global image semantics to infer shadows. Compared to DSC, our method has 34.88% reduction on SBU and 34.57% reduction on UCF in terms of BER, demonstrating that our method (BDRAR) outperforms the others on both benchmark datasets. Although our shadow detection network is trained on the SBU training set [4], it still has a superior performance over the others on the UCF dataset, which demonstrates the generalization capability of our network.

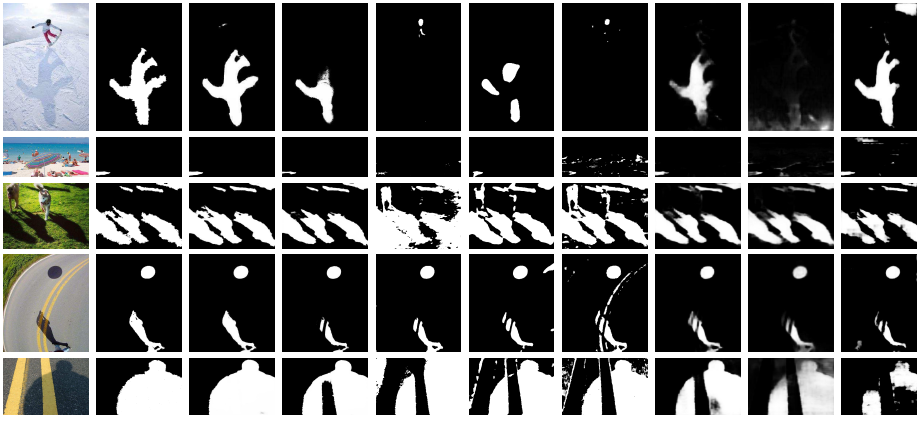


Fig. 5: Visual comparison of shadow maps produced by our method and others (4th-10th columns) against ground truths shown in 2nd column. Note that "stCNN" and "paCNN" stand for "stacked-CNN" and "patched-CNN", respectively.

In Figs. 4 and 5, we provide visual comparison results on different input images. From the results, we can find that our method (3rd column of Figs. 4 and 5) can effectively locate shadows under various backgrounds and avoid false positives, and thus has the best performance among all the shadow detectors. Moreover, for high-contrast objects in a large shadow region, our method still recognizes them as shadows; see the last two rows of Fig. 5.

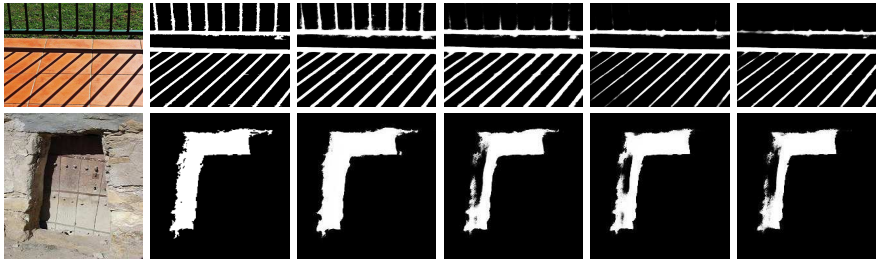
4.3 Comparison with Methods of Shadow Removal, Saliency Detection and Semantic Segmentation

Note that deep networks designed for shadow removal, saliency detection and semantic image segmentation can be re-trained for shadow detection by using annotated shadow datasets. To further evaluate the effectiveness of our method, another experiment is conducted by comparing our method with a recent shadow removal model, i.e., DeshadowNet [33], two recent deep saliency detection models, i.e., SRM [34] and Amulet [35], and a recent semantic segmentation model, i.e., PSPNet [36].

Since we cannot obtain the original code of DeshadowNet [33], we carefully follow the published paper of DeshadowNet to implement it with our best efforts and train the model for shadow detection on the SBU training set. For the other three methods, we obtain the code of these methods from their project web-pages, and re-train their models on the SBU training set. For a fair comparison, we try our best to tune their training parameters and select the best shadow detection results. The last four rows in Table 1 report the BER values of these methods. Even though they have better BER values than some existing

Table 2: Ablation analysis. We train all the networks using the SBU training set and test them using the SBU testing set [4], and UCF testing set [8].

	SBU [4]	UCF [8]
network	BER	BER
FPN	5.78	7.70
BD	5.37	7.11
RAR	4.33	6.09
BDR	4.23	6.71
BDAR	3.74	6.19
BDRAR (ours)	**3.64**	**5.30**
BDRAR_w/o_sw	3.89	5.66



(a) inputs (b) GT (c) ours (d) RAR (e) BD (f) FPN

Fig. 6: Comparing the shadow maps produced by our method (c) and by the other three models (d)-(f) against the ground truths (denoted as "GT") in (b).

shadow detectors, our method still demonstrates a superior shadow detection performance over them on both benchmark datasets. On the other hand, the last three columns in Figs. 4 and 5 present the predicted shadow maps, showing that our network can consistently produce better shadow detection maps than the other methods.

4.4 Ablation Analysis

We perform experiments to evaluate the bidirectional feature integration in the FPN and the effectiveness of the RAR module design. The basic model is the original "FPN [9]," which only uses the top-down direction to integrate features and removes all the RAR modules shown in Fig. 1. The second model (denoted as "BD") is similar to the "FPN," but it uses our bidirectional pathway to merge features at different layers of the CNN. The third model (denoted as "RAR") is the "FPN" with the RAR modules only. The fourth model (denoted as "BDR") replaces all the RAR modules of our network with residual learning based modules (see Fig. 2 (b)), while the fifth model (denoted as "BDAR") replaces all our RAR modules with the attention residual learning modules (see Fig. 2 (c)), which means that this model is constructed by removing the recurrent mechanism from our RAR modules. The last model (denoted as "BDRAR_w/o_sw")

Table 3: RAR module with different recurrent steps.

	SBU [4]		UCF [8]	
number of recurrent steps	BER	improvement	BER	improvement
1 (BDRA)	3.74	-	6.19	-
2	3.64	2.67%	5.30	14.38%
3	3.89	-6.87%	5.66	-6.79%

has a similar structure with our BDRAR, but it uses independent weights at each recurrent step in our RAR modules.

Table 2 summaries the compared BER values on both benchmark datasets. From the results, we can see that both "replacing the single top-down pathway of the FPN with the bidirectional pathways" and "adopting the RAR modules" lead to an obvious improvement. Compared to results of our network with the residual learning based module (see Fig. 2 (b)) and attention residual module (see Fig. 2 (c)), our RAR modules (see Fig. 2 (d)) have better performance on shadow detection, since it can recurrently learn a set of attention weights to select good residual features to refine the integrated features, as shown in Table 2. Moreover, we provide visual analysis to evaluate how RAR and bidirectional feature integration contribute by conducting an experiment by comparing our method with three models: "FPN", "BD", and "RAR." Fig. 6 shows the comparisons on two input images, showing that RAR and BD can detect more shadow regions, as shown in Fig. 6(d-f). More importantly, our method with both RAR and bidirectional integration produces the best performance, and our predicted shadow maps are more similar to the ground truths (GT). Lastly, our method also outperforms "BDRAR_w/o_sw", showing that sharing weights in the RAR modules can reduce the learning parameters of the network, and thus leads to better shadow detection results.

Note that our RAR module (see Fig. 2 (d)) recurrently employs the AR module (see Fig. 2 (c)) to refine features at each layer by merging two adjacent features. Hence, a basic question of configuring our network is how many recurrent steps we use in our RAR modules. We adopt the network with the RAR modules as the baseline (BDAR), which has only one recurrent step (see Table 2); We conduct an experiment for comparisons by modifying our network with different rounds of recurrent steps (the round of AR modules; see Fig. 2 (c) in our RAR), and Table 3 reports the results. As shown in Table 3, we can find that having two recurrent steps in the RAR module achieves the best performance on shadow detection. Compared to only one AR module, the network with two rounds of AR models can enhance the quality of the refined features at each layer by further integrating the adjacent features. However, when there are three rounds of AR modules in our RAR, it largely increases the complexity of our network, thus making the network training more difficult.

Fig. 7: More results produced from our network.

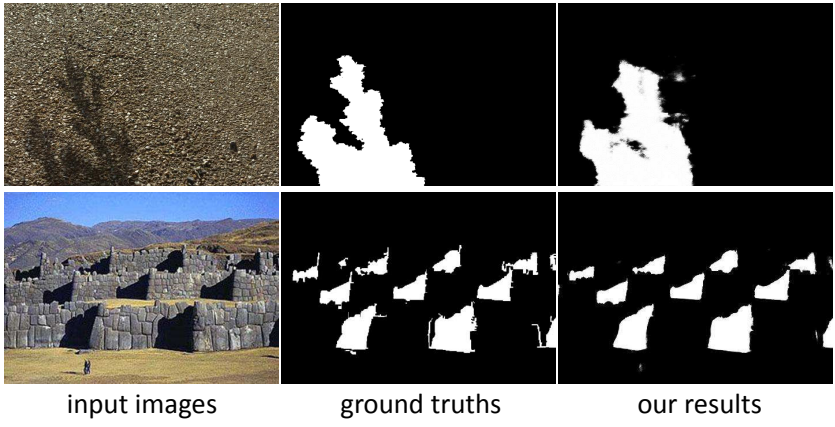


Fig. 8: Failure cases of our network.

4.5 More Shadow Detection Results

In Fig. 7, we show more shadow detection results: (a) low-contrast shadow boundary; (b) unconnected shadows with a black background; (c) multiple human objects; and (d) tiny and irregular shadows. From the results, we can see that our method can still detect these shadows fairly well. Note that our method also has its limitations, and tends to fail in some extreme cases, such as the soft shadows (see Fig. 8 (top)), and shadows with tiny details (see Fig. 8 (bottom)).

4.6 Saliency Detection

Our deep model has the potential to handle other vision tasks. Here, we take the saliency detection as an example. To evaluate the saliency detection performance of our deep model, we first re-trained our model on "MSRA10k," which is a widely-used dataset for saliency object detection, and then tested the trained model on four widely-used benchmark datasets, including ECSSD, HKU-IS, PASCAL-S, and DUT-OMRON; please refer to [37, 38] for the details

Table 4: Comparison with the state-of-the-art methods on saliency detection.

Method	ECSSD		HKU-IS		PASCAL-S		DUT-OMRON	
	F_β	MAE	F_β	MAE	F_β	MAE	F_β	MAE
NLDF [39]	0.905	0.063	0.902	0.048	0.831	0.099	0.753	0.080
UCF [40]	0.910	0.078	0.886	0.073	0.821	0.120	0.735	0.131
DSS [37]	0.916	0.053	0.911	0.040	0.829	0.102	0.771	0.066
Amulet [35]	0.913	0.059	0.887	0.053	0.828	0.095	0.737	0.083
SRM [34]	0.917	0.056	0.906	0.046	0.844	**0.087**	0.769	0.069
RADF [38]	0.924	0.049	0.914	0.039	0.832	0.102	0.789	0.060
BDRAR (ours)	**0.935**	**0.045**	**0.916**	**0.038**	**0.846**	0.109	**0.808**	**0.058**

of these datasets. Moreover, we used two common metrics (F_β and MAE; see [37] for their definitions) for the quantitative comparisons among different saliency detectors. Table 4 shows the quantitative comparisons between our model and several state-of-the-art saliency detectors. From the table, we can see that our model produces the best performance on almost all the four benchmarks in terms of F_β and MAE, showing that our model predicts more accurate saliency maps.

5 Conclusion

This paper presents a novel network for single-image shadow detection. Two new techniques, recurrent attention residual (RAR) module and bidirectional feature pyramid network (BFPN), are presented to fully explore the global and local context information encoded in different layers of the convolutional neural network (CNN). The RAR module presents a novel feature refinement strategy for the context features at adjacent layers by learning the attention weights to select a residual in a recurrent manner, while the BFPN aggregates the shadow context features at different layers in two directions, and it can enhance the shadow boundaries as well as suppress the non-shadow regions. In the end, our network achieves the state-of-the-art performance on two benchmark datasets and outperforms other methods by a significant margin.

Acknowledgments

The work is supported by the National Basic Program of China, 973 Program (Project no. 2015CB351706), the Research Grants Council of the Hong Kong Special Administrative Region (Project no. CUHK 14225616), Shenzhen Science and Technology Program (No. JCYJ20160429190300857 and JCYJ20170413162617606), the CUHK strategic recruitment fund, the NSFC (Grant No. 61272293, 61300137, 61472145, 61233012) and NSFG (Grant No. S2013010014973), RGC Fund (Grant No. CUHK14200915), Science and Technology Planning Major Project of Guangdong Province (Grant No. 2015A070711001), and Open Project Program of Guangdong Key Lab of Popular High Performance Computers and Shenzhen Key Lab of Service Computing and Applications (Grant No.SZU-GDPHPCL2015). Xiaowei Hu is funded by the Hong Kong Ph.D. Fellowship.

References

1. Finlayson, G.D., Hordley, S.D., Lu, C., Drew, M.S.: On the removal of shadows from images. IEEE Transactions on Pattern Analysis and Machine Intelligence **28**(1) (2006) 59–68
2. Finlayson, G.D., Drew, M.S., Lu, C.: Entropy minimization for shadow removal. International Journal of Computer Vision **85**(1) (2009) 35–57
3. Khan, S.H., Bennamoun, M., Sohel, F., Togneri, R.: Automatic feature learning for robust shadow detection. In: CVPR. (2014) 1939–1946
4. Vicente, T.F.Y., Hou, L., Yu, C.P., Hoai, M., Samaras, D.: Large-scale training of shadow detectors with noisily-annotated shadow examples. In: ECCV. (2016) 816–832
5. Nguyen, V., Vicente, T.F.Y., Zhao, M., Hoai, M., Samaras, D.: Shadow detection with conditional generative adversarial networks. In: ICCV. (2017) 4510–4518
6. Hu, X., Zhu, L., Fu, C.W., Qin, J., Heng, P.A.: Direction-aware spatial context features for shadow detection. In: CVPR. (2018) 7454–7462
7. Hu, X., Fu, C.W., Zhu, L., Qin, J., Heng, P.A.: Direction-aware spatial context features for shadow detection and removal. arXiv preprint arXiv:1805.04635 (2018)
8. Zhu, J., Samuel, K.G., Masood, S.Z., Tappen, M.F.: Learning to recognize shadows in monochromatic natural images. In: CVPR. (2010) 223–230
9. Lin, T.Y., Dollár, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: CVPR. (2017) 2117–2125
10. Okabe, T., Sato, I., Sato, Y.: Attached shadow coding: Estimating surface normals from shadows under unknown reflectance and lighting conditions. In: ICCV. (2009) 1693–1700
11. Karsch, K., Hedau, V., Forsyth, D., Hoiem, D.: Rendering synthetic objects into legacy photographs. ACM Trans. on Graphics (SIGGRAPH Asia) **30**(6) (2011) 157:1–157:12
12. Lalonde, J.F., Efros, A.A., Narasimhan, S.G.: Estimating natural illumination from a single outdoor image. In: ICCV. (2009) 183–190
13. Junejo, I.N., Foroosh, H.: Estimating geo-temporal location of stationary cameras using shadow trajectories. In: ECCV. (2008) 318–331
14. Ecins, A., Fermuller, C., Aloimonos, Y.: Shadow free segmentation in still images using local density measure. In: ICCP. (2014) 1–8
15. Cucchiara, R., Grana, C., Piccardi, M., Prati, A.: Detecting moving objects, ghosts, and shadows in video streams. IEEE Transactions on Pattern Analysis and Machine Intelligence **25**(10) (2003) 1337–1342
16. Nadimi, S., Bhanu, B.: Physical models for moving shadow and object detection in video. IEEE Transactions on Pattern Analysis and Machine Intelligence **26**(8) (2004) 1079–1087
17. Tian, J., Qi, X., Qu, L., Tang, Y.: New spectrum ratio properties and features for shadow detection. Pattern Recognition **51** (2016) 85–96
18. Lalonde, J.F., Efros, A.A., Narasimhan, S.G.: Detecting ground shadows in outdoor consumer photographs. In: ECCV. (2010) 322–335
19. Guo, R., Dai, Q., Hoiem, D.: Single-image shadow detection and removal using paired regions. In: CVPR. (2011) 2033–2040
20. Huang, X., Hua, G., Tumblin, J., Williams, L.: What characterizes a shadow boundary under the sun and sky? In: ICCV. (2011) 898–905
21. Vicente, Y., Tomas, F., Hoai, M., Samaras, D.: Leave-one-out kernel optimization for shadow detection. In: ICCV. (2015) 3388–3396

22. Tai, Y., Yang, J., Liu, X.: Image super-resolution via deep recursive residual network. In: CVPR. (2017) 3147–3155
23. Wang, F., Jiang, M., Qian, C., Yang, S., Li, C., Zhang, H., Wang, X., Tang, X.: Residual attention network for image classification. In: CVPR. (2017) 3156–3164
24. Hosseinzadeh, S., Shakeri, M., Zhang, H.: Fast shadow detection from a single image using a patched convolutional neural network. arXiv preprint arXiv:1709.09283 (2017)
25. Bell, S., Zitnick, C.L., Bala, K., Girshick, R.: Inside-outside net: Detecting objects in context with skip pooling and recurrent neural networks. In: CVPR. (2016) 2874–2883
26. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR. (2016) 770–778
27. Deng, Z., Hu, X., Zhu, L., Xu, X., Qin, J., Han, G., Heng, P.A.: R^3Net: Recurrent residual refinement network for saliency detection. In: IJCAI. (2018) 684–690
28. Li, G., Xie, Y., Lin, L., Yu, Y.: Instance-level salient object segmentation. In: CVPR. (2017) 247–256
29. Xie, S., Girshick, R., Dollár, P., Tu, Z., He, K.: Aggregated residual transformations for deep neural networks. In: CVPR. (2017) 5987–5995
30. Xie, S., Tu, Z.: Holistically-nested edge detection. In: ICCV. (2015) 1395–1403
31. Liu, W., Rabinovich, A., Berg, A.C.: Parsenet: Looking wider to see better. arXiv preprint arXiv:1506.04579 (2015)
32. Krähenbühl, P., Koltun, V.: Efficient inference in fully connected CRFs with Gaussian edge potentials. In: NIPS. (2011) 109–117
33. Qu, L., Tian, J., He, S., Tang, Y., Lau, R.W.: DeshadowNet: A multi-context embedding deep network for shadow removal. In: CVPR. (2017) 4067–4075
34. Wang, T., Borji, A., Zhang, L., Zhang, P., Lu, H.: A stagewise refinement model for detecting salient objects in images. In: ICCV. (2017) 4019–4028
35. Zhang, P., Wang, D., Lu, H., Wang, H., Ruan, X.: Amulet: Aggregating multi-level convolutional features for salient object detection. In: ICCV. (2017) 202–211
36. Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J.: Pyramid scene parsing network. In: CVPR. (2017) 2881–2890
37. Hou, Q., Cheng, M.M., Hu, X.W., Borji, A., Tu, Z., Torr, P.: Deeply supervised salient object detection with short connections. In: CVPR. (2017) 3203–3212
38. Hu, X., Zhu, L., Qin, J., Fu, C.W., Heng, P.A.: Recurrently aggregating deep features for salient object detection. In: AAAI. (2018) 6943–6950
39. Luo, Z., Mishra, A., Achkar, A., Eichel, J., Li, S., Jodoin, P.M.: Non-local deep features for salient object detection. In: CVPR. (2017) 6609–6617
40. Zhang, P., Wang, D., Lu, H., Wang, H., Yin, B.: Learning uncertain convolutional features for accurate saliency detection. In: ICCV. (2017) 212–221