Supplementary material for Fine-Context Shadow Detection using Shadow Removal

Jeya Maria Jose Valanarasu, and Vishal M. Patel
Johns Hopkins University
{jvalana1, vpatel36}@jhu.edu

In this supplementary material, we give more explanations for our proposed methods. We show emprically why our fine context detector block helps in learning more fine details about the shadow region. We then describe more about the fusion block in FCSD-Net and illustrate it. We give more details and reasons behind our design of FCSD-Net and R2D. We also illustrate more results for comparison.

1. Fine Context Feature Learning

A generic ConvNet has a encoder-decoder architecture which is an undercomplete type of architecture spatially that learns more high level features when the network is designed more deep. Overcomplete representations [3] were initially introduced in signal processing as an alternate method for signal representation. Overcomplete bases or dictionaries were proposed where the number of basis functions are more than the number of samples of input signal. Overcomplete bases have a better flexibility at capturing the structure of the data and so is more robust. In [8], overcomplete auto-encoders were observed to be better feature extractors for denoising. In an overcomplete auto-encoder, the number of neurons in the hidden layer is more than the that of the initial layers. So typically, the dimensionality of the representation in the deeper layers is more than that of the input layer. In the deep learning era, the concept of overcomplete representations has been under-explored [6]. In an overcomplete alternate convolutional network the input image is taken to a higher dimension spatially. So, the max-pooling layers in a typical ConvNet can be replaced with upsampling layers to prevent the receptive field size to increase in the deeper layers of the network.

Consider a configuration of two conv layers in succession where I be the input image, F_1 and F_2 be the feature maps extracted from the conv layers 1 and 2, respectively. Let the initial receptive field of the conv filter be $k \times k$ on the image. Now, if there is a max-pooling layer present in between the conv layers like in generic ConvNets, the receptive field would become larger in the successive layers. The receptive field size change due to max-pooling layer is dependent on two variables- pooling coefficient and stride of the pooling filter. Considering a default configuration (like in most cases) where both pooling coefficient and stride is 2, the receptive field of conv layer 2 (to which F_1 is forwarded) on the input image would be $2 \times k \times 2 \times k$. Similarly, the receptive field of conv layer 3 (to which F_2 is forwarded) would be $4 \times k \times 4 \times k$. This increase in receptive field can be generalized for the i^{th} layer in an undercomplete network as follows:

$$RF(w.r.t\ I) = 2^{2*(i-1)} \times k \times k \tag{1}$$

In an overcomplete ConvNet, we propose using an upsampling layer instead of the max-pooling layer. As the upsampling layer actually works opposite to that of max-pooling layer, the receptive field of conv layer 2 on the input image now would be $\frac{1}{2} \times k \times \frac{1}{2} \times k$. Similarly, the receptive field of conv layer 3 now would be $\frac{1}{4} \times k \times \frac{1}{4} \times k$. This increase in receptive field can be generalized for the i^{th} layer in the overcomplete ConvNet as follows:

$$RF(w.r.t\ I) = \left(\frac{1}{2}\right)^{2*(i-1)} \times k \times k. \tag{2}$$

This helps in an overcomplete network to learn more low-level information like edges and other finer details better. So in our work, the Fine Context Block has this alternate ConvNet architecture to learn fine details of the shadow region.

2. FCSD-Net

In this section, we give more explanation about the fusion block and give a justification for the design of FCD architecture.

2.1. Fusion Block

The fusion block in FCSD-Net is used to fuse the features from the FCD and CCD blocks. Fusion block is illustrated in 1. The input to the fusion blocks are features $F1$ to

$F6$, $DS1$ to $DS6$. In the fusion block, first we interpolate all the feature maps to the same size as of the input image so that the dimensions are consistent during fusion. After this we concatenate all the feature maps and pass them through a 1×1 conv layer. This output is forwarded to a sigmoid activation to get the binary shadow map as output.

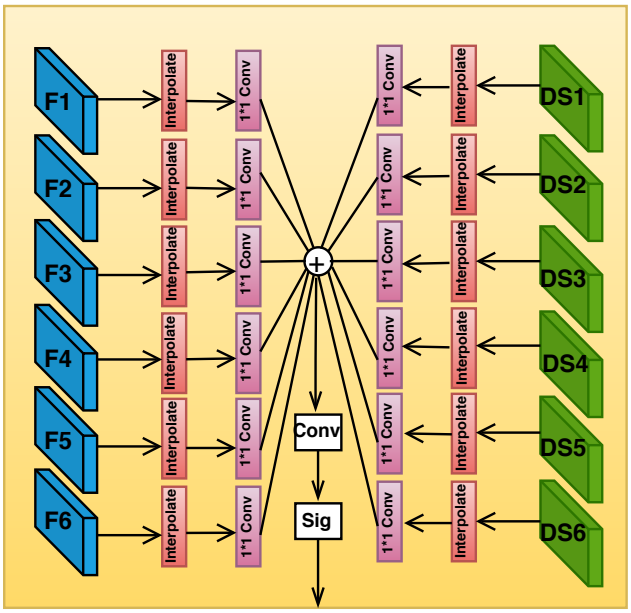

Figure 1. The details of the fusion block found in FCSD-Net. Input feature maps $F1-F6$ and $DS1-DS6$ are interpolated and passed through a $1*1$ conv layer and then fused. It is then passed through a conv layer followed by a sigmoid activation to get the prediction.

2.2. FCD architecture Design Justification

In the Fine Context Detector (FCD) block, we use four convolutional blocks where we upsample the input features at each block. Each conv block has a conv layer followed by an upsampling layer and ReLU activation. The upsampling layer used here is bilinear interpolation. We use four conv blocks such that the final resolution of the feature map at FCD block is 400×400. This limit is based on trade-off between performance and model complexity. We noticed that the change in performance was not too affected after we reached a resolution of 400×400. This observation could be explained as with increase in resolution the complexity of the network training gets hindered. We conducted experiments on ISTD dataset to show how the number of conv blocks in FCD block affected the performance. These observations can be found in Table 1.

3. R2D

In this section, we give justification for using U-Net as our restoration network. We also discuss on why we chose $R1$ and $R2$ from layers 2 and 5 respectively.

No. of conv blocks	1	2	3	4	5
BER	2.10	1.85	1.77	**1.71**	**1.71**

Table 1. Change in performance with the number of conv blocks in the FCD block. The performance saturates after 4 number of conv blocks.

3.1. U-Net as R() - Justification

In R2D, we proposed using an U-Net [5] architecture for our restoration network. We use a 5 layer deep U-Net which has 5 conv blocks in its encoder and decoder. The conv block in encoder has a conv layer followed by max-pooling, batch normalization and ReLU activation. Each conv block in decoder has a conv layer followed by an up-sampling layer, batch normalization and ReLU activation. The number of filters in encoder are as follows: 32,64,128,256, and 512. The number of filters in the decoder are in reverse order of the above.

The main reason we choose U-Net as our restoration network because of its low complexity when compared to other networks specifically designed for shadow removal. Notable works like [1, 2, 4, 7] provide methods and networks specifically designed for shadow removal. However, using those methods as our restoration network $R()$ increases the complexity of the whole framework as well as complicates the optimization process. As the main objective of this work is to perform shadow detection, we just use U-Net as $R()$ instead of the state of the art restoration networks as it is itself capable of extracting the shadow features from the input image as seen in Figure 7 of the main paper.

3.2. Choosing R1 and R2

In our proposed framework, we chose $R1$ and $R2$ from the second and last layer of encoder in U-Net respectively. We do not feed forward all the features from $R()$ to $D()$ as it increases the complexity of network. The motivation is to feed forward a good combination of both local and global features of the shadow region to the detection network $D()$. We observe that features at layer 1 and 2 extract local features while layers 3,4, and 5 extract global features. So we choose $R1$ and $R2$ from layers 2 and 5 to get the abstract local and global features from network $R()$.

4. Performance Analysis and Comparison with other methods

Although we use less extra images than MTMT-Net (we used an extra of 1300 shadow free images while MTMT-Net used 3472 real world images), we achieve better results. Also, we note that adopting the semi-supervised strategy of MTMT-Net for FCSD-Net should further enhance the performance. We note that the ISTD dataset consists of a very

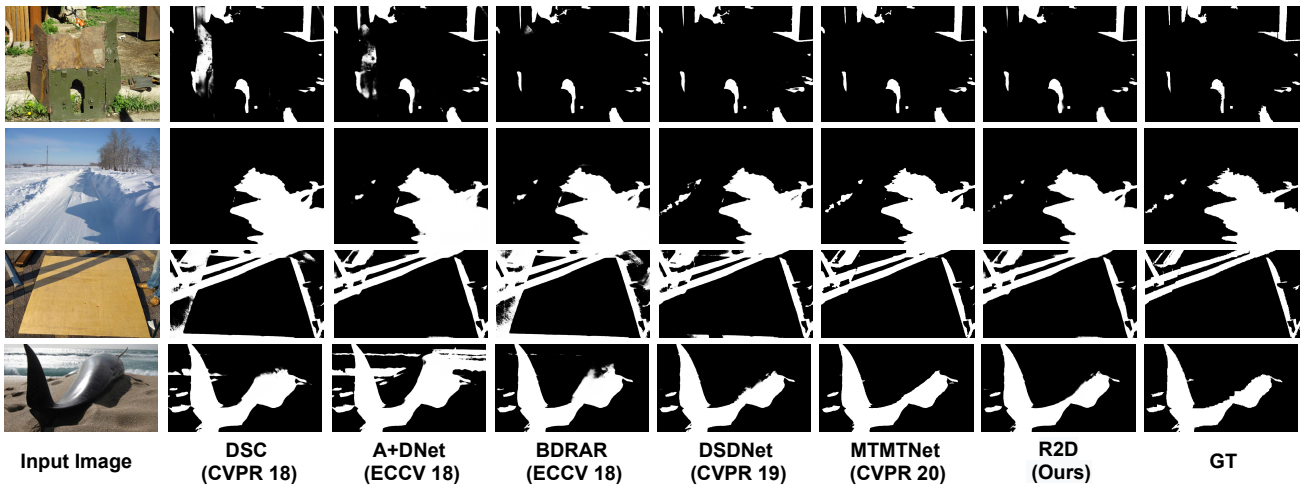

Figure 2. Comparison of predictions of our proposed methods with leading shadow detection methods. The first and last columns correspond to the input and ground-truth, respectively. Other columns correspond to the predictions obtained using different methods.

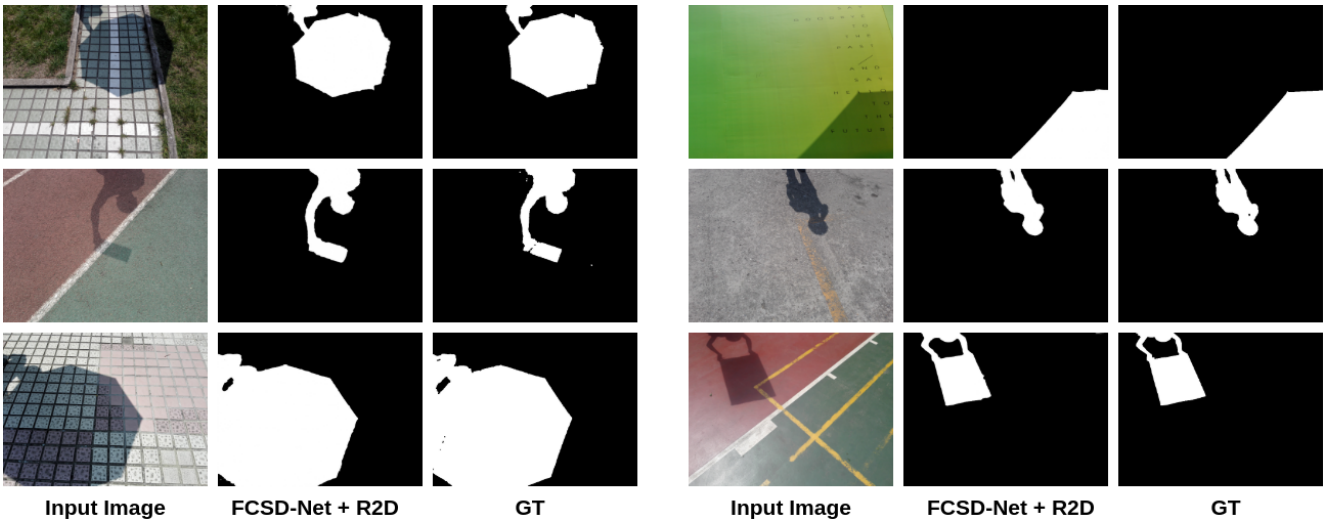

Figure 3. Our predictions for different backgrounds found in ISTD dataset.

limited number of shadow shapes (less than 10) and background. This makes the shadow detection task less challenging for any network trained on this dataset. So, it can be seen that the improvements in the UCF and SBU datasets are more when compared to the ISTD dataset as there are still more features from varied shadow objects for R2D to learn and leverage.

5. Additional Results

We illustrate more results for comparison in Fig 2. From all the images, it can be observed that our method detects all the fine details better than the other methods. We also visualize more predictions from ISTD dataset in Fig 3 where our method gives a prediction as good as the ground truth.

6. Conclusion

R2D is a promising step towards achieving best detection performances as a pre-trained R() can be directly plugged into any D(), to improve its performance. Also, FSCD-Net solves most confounding cases by focusing on local context which we believe are important contributions to the shadow detection literature.

References

[1] Xiaodong Cun, Chi-Man Pun, and Cheng Shi. Towards ghost-free shadow removal via dual hierarchical aggregation network and shadow matting gan. In *AAAI*, pages 10680–10687, 2020.

[2] Hieu Le and Dimitris Samaras. From shadow segmentation to shadow removal. *arXiv preprint arXiv:2008.00267*, 2020.

[3] Michael S Lewicki and Terrence J Sejnowski. Learning overcomplete representations. *Neural computation*, 12(2):337–365, 2000.

[4] Liangqiong Qu, Jiandong Tian, Shengfeng He, Yandong Tang, and Rynson WH Lau. Deshadownet: A multi-context embedding deep network for shadow removal. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4067–4075, 2017.

[5] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[6] Jeya Maria Jose Valanarasu, Vishwanath A Sindagi, Ilker Hacihaliloglu, and Vishal M Patel. Kiu-net: Overcomplete convolutional architectures for biomedical image and volumetric segmentation. *arXiv preprint arXiv:2010.01663*, 2020.

[7] Florin-Alexandru Vasluianu, Andres Romero, Luc Van Gool, and Radu Timofte. Self-supervised shadow removal. *arXiv preprint arXiv:2010.11619*, 2020.

[8] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning*, pages 1096–1103, 2008.