V-DESIRR: Very Fast Deep Embedded Single Image Reflection Removal

B H Pawan Prasad[1] Green Rosh K S[1] Lokesh R B[1] Kaushik Mitra[2]
Sanjoy Chowdhury[1]
[1]Samsung R&D Institute Bangalore, India [2]IIT Madras, Chennai, India
pawan.prasad,greenrosh.ks,lokesh.rb,s5.chowdhury@samsung.com, kmitra@ee.iitm.ac.in

Abstract

Real world images often gets corrupted due to unwanted reflections and their removal is highly desirable. A major share of such images originate from smart phone cameras capable of very high resolution captures. Most of the existing methods either focus on restoration quality by compromising on processing speed and memory requirements or, focus on removing reflections at very low resolutions, there by limiting their practical deploy-ability. We propose a light weight deep learning model for reflection removal using a novel scale space architecture. Our method processes the corrupted image in two stages, a Low Scale Sub-network (LSSNet) to process the lowest scale and a Progressive Inference (PI) stage to process all the higher scales. In order to reduce the computational complexity, the sub-networks in PI stage are designed to be much shallower than LSSNet. Moreover, we employ weight sharing between various scales within the PI stage to limit the model size. This also allows our method to generalize to very high resolutions without explicit retraining. Our method is superior both qualitatively and quantitatively compared to the state of the art methods and at the same time $20\times$ *faster with* $50\times$ *less number of parameters compared to the most recent state-of-the-art algorithm RAGNet. We implemented our method on an android smart phone, where a high resolution 12 MP image is restored in under 5 seconds.*

1. Introduction

Image capture in the vicinity of a reflective surface such as glass windows is very challenging due to the formation of undesirable reflection artifacts. These artifacts not only affect the perceptual quality of the image, but also impact high-level tasks such as image recognition and object detection. Hence removal of reflection is very desirable and is an active area of research in image processing and computer vision.

Several methods have been proposed in the past to address the problem of reflection, The earliest methods try to solve the problem by imposing additional handcrafted constraints on the problem definition such as natural scene statistics [10], sparsity priors [11], gradient smoothness [32] and ghosting cues [22]. However it has been observed that solutions based on handcrafted priors do not adapt very well to the complex reflection patterns often observed in real life.

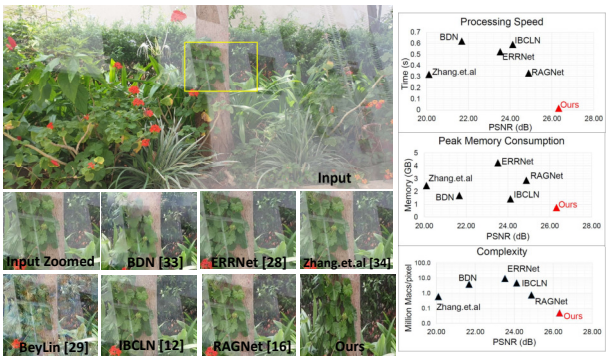


Figure 1: High Resolution Performance and Complexity Analysis. (a) Comparison of our method against current state of the art for a high resolution 12MP input image. (b) Complexity evaluation on NVIDIA GTX 1080Ti for an image tile of size 540x400.

In order to model real-life reflection patterns, several data-driven approaches using deep neural networks have been proposed in the recent past [12] [16] [28] [34] . Even though these methods achieve state-of-the-art results and is able to model strong and complex reflections efficiently, they suffer from two major drawbacks:

a) ***High computation requirements:*** Reflection removal needs a large receptive field to efficiently gather the semantic information required for recovering the transmission layer. The conventional methods try to increase the receptive field by stacking up a large number of convolution filters which tremendously increases the computation and memory requirements. For example, a recent method proposed by RAGNet [16] consists of 131 million parameters, consumes a peak memory of 2.9 GB and takes 20

seconds to process a 12 MP image on a NVIDIA 1080Ti GPU. This makes it impossible to deploy such methods on a smart phone device with limited computation power and memory making their real-life applicability extremely limited.

b) ***Inability to remove reflections from high resolution images:*** Several contemporary smart phone vendors provide multiple cameras with resolutions varying from 8 megapixels to 108 megapixels. Hence a deploy-able solution should be able to remove reflections from a wide range of image resolutions. However, most of the state-of-the-art methods uses fixed network architectures thus making its receptive field static. Moreover most of these methods are trained on low resolution data sets which are less than a megapixel in size. Since the semantic content of an image in a fixed receptive field varies over resolutions, these methods are not 'scale invariant' and hence cannot remove reflections on high resolution images. Hence these methods need to be retrained for each resolution to efficiently remove reflections which is very cumbersome.

In this paper, we propose a novel method to address the aforementioned challenges while maintaining output image quality. Inspired by recent success of scale-space approaches in image deblurring [19] [25], we propose a scale-space reflection removal approach for increasing the receptive field with minimal increase in computational overhead. The corrupted input image is transformed into its scale space representation, and is processed by identical, weight-shared deep CNNs at each scale except the lowest scale. To make our method more efficient, we use a deeper network at the lowest scale (Low Scale Network - LSSNet) and a much shallower network for higher scales (High Scale Network - HSSNet). The output at each scale is up sampled using Convolutional Guided Filters (CGF) [30] and appended to the input of the immediate higher level to aid the reflection removal process at each scale. As observed by [25], the same problem is solved at each scale, and hence weights can be shared between all the HSSNets and CGF blocks in the scale space resulting in a very low memory footprint. This also enables us to dynamically increase the effective receptive field during inference easily by increasing the number of scales. Hence the proposed method can remove reflections from high resolution images without any explicit retraining making it easily adaptable to smart phones.

A sample output from the proposed method is shown in Fig. 1. The input image given to the network is of very high resolution (12 MP) and the state of the art methods train on images of much lower resolution with a fixed receptive field, hence fails to remove reflections from the image. Whereas our method is able to generalize well to high resolution images even without training on such images. Also in Fig. 1, we show three plots to demonstrate how the performance with respect to execution time, memory and network complexity compares to the state-of-the-art methods. We show in the later sections that our method is able to achieve better performance than the state of the art methods while being much more computationally efficient. It runs $20\times$ faster with a reduction in peak memory usage of $3\times$ compared to RAGNet [16] on NVIDIA 1080Ti GPU. Further, our method uses only 2.6 million learnable parameters which is $50\times$ less than RAGNet [16]. We also implemented our method on a mobile android device with 8 GB RAM and Qualcomm Snapdragon 888 processor. We observed that the proposed method can process a 12 megapixel input data in under 5 seconds and hence can easily be adapted to smart phone devices to process even high resolution images under a reasonable time.

The contributions of our work are as follows:
(1) We propose a fast scale space approach for reflection removal which can easily be deployed on resource limited devices such as smart phones.
(2) To make our method computationally efficient, we use a deeper network only at the lowest scale, while the higher scales are processed using much shallower networks. This makes our method $20\times$ faster than the most recent state-of-the-art method RAGNet.
(3) The proposed algorithm is scalable to handle high resolution input images (tested up to 64 MP) without the need for explicit retraining.
(4) The proposed algorithm can process a 12 MP image in under 5 seconds on a smart phone with a Qualcomm Snapdragon 888 chip set and 8 GB RAM. To the best of our knowledge, this is the fastest deep learning based method for reflection removal with state-of-the-art results.
(5) We build a high resolution dataset captured using latest smartphones with real world reflections that can enable future evaluations. The dataset will be available at https://www.github.com/ee19d005/vdesirr.

2. Related Work

Existing methods on reflection removal fall into the following three categories [26] based on the type of inputs used to generate a reflection free output namely (a) Single image [11] [3] [15] [22] [5] [27] [34] [16] [12] [28] (b) Multi image[6] [7] [14] [1] [17] [31] [24] (c) Multi modal reflection removal [20] [2] [13]. We only provide further details of the single image based methods in this section.

Several methods explored in the past use traditional optimization based approaches or more recently deep learning based approaches. Traditional methods rely on priors such as gradient sparsity prior depending upon edges [11], corners [3], layer smoothness priors that use gradient information of reflection and transmission layers to perform edge classification [32], different probability distributions to model transmission and reflection layers [15]. A Gaus-

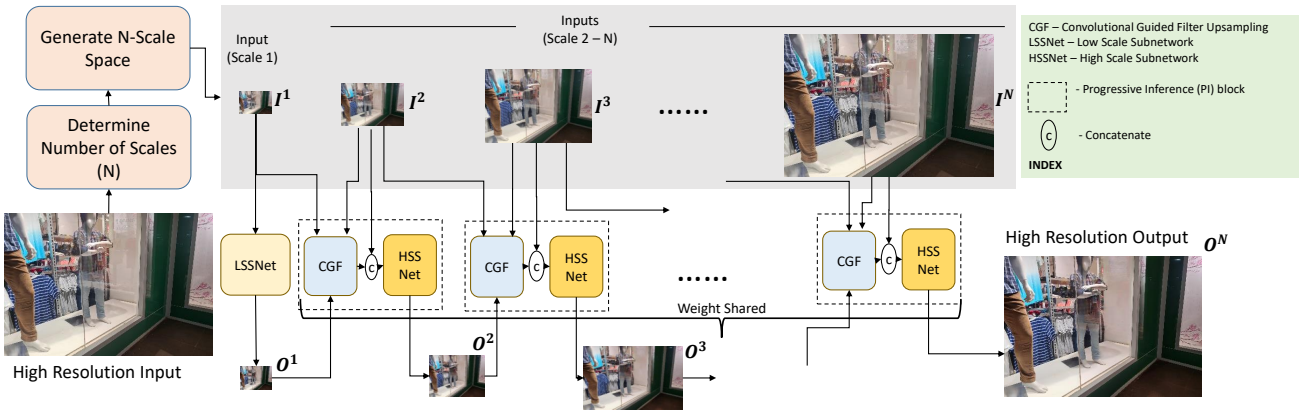


Figure 2: Overview of the proposed very fast deep embedded single image reflection removal for high resolution images.

sian mixture model (GMM) patch based prior with an image formation model comprising of reflection and its spatial shifts and a GMM to model the distributions was used in [22]. Single image deep learning methods have seen good amount of progress in the recent past. An end-to-end single image deep learning architecture was proposed in [5]. Usage of perceptual loss was introduced later in [34] and a multi scale guided concurrent neural network was proposed in [27]. A non-linear blending model was used to model realistic reflection in [29]. A bi-directional approach where an estimated reflection layer is used to refine the transmission layer was proposed in [33]. An alignment invariant loss function was introduced in [28] that would reduce the challenges involved in data acquisition by relaxing the constraint that the ground truth reflection free image and the input degraded image needs to be perfectly aligned for training the network. More recently, a cascaded two stage architecture was proposed that uses reflection aware guidance to further improve single image reflection removal [16].

3. Proposed Scale-Space Architecture

Following sub sections describe the different components of the proposed method.

3.1. Pipeline

An overview of the proposed method is provided in Fig. 2. Given a corrupted input image (I) of resolution $H \times W$, the proposed method determines the number of scales N as

$$N = \max(1, ceil(1 + \log_2(\frac{min(H, W)}{k}))) \qquad (1)$$

where k has to be greater than the receptive field of all the sub-networks used in the pipeline, and is chosen as 300 for the proposed method.

Next, a N-scale space representation of the input image is constructed using a Gaussian pyramid. The final reflection free output image is generated from the scale space in two stages: a) Low scale sub-network (LSSNet) and b) Progressive Inference (PI) of the higher scales using Convolutional Guided Filter up-sampling (CGF) and High Scale Sub-network (HSSNet). Each of these components are detailed next in this section.

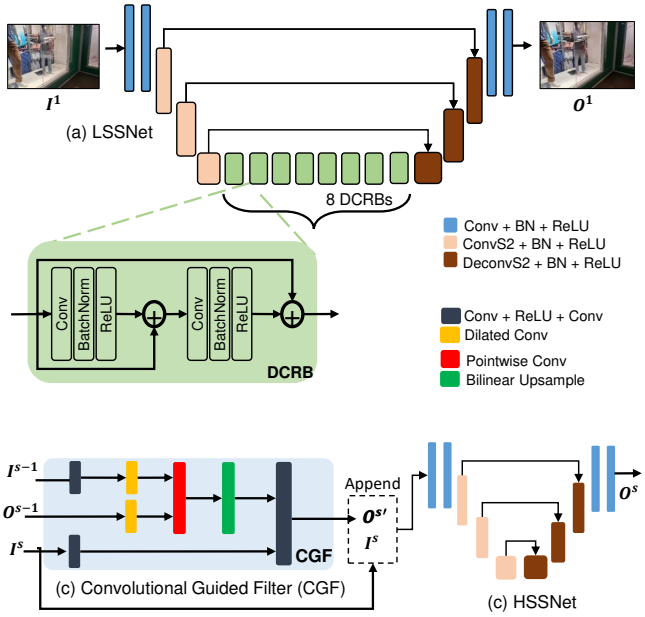


Figure 3: Network Architecture of the two proposed sub networks Low Scale Space Network (LSSNet) and High Scale Space Network (HSSNet) along with the Convolution Guided Filter (CGF).

3.1.1 Low Scale Sub Network - LSSNet

The input image at the lowest scale is passed to LSSNet to generate an estimate of the output image (O^1) at scale 1. The network architecture of LSSNet is provided

in Fig. 3, the design is similar to a U-net [21] with additional enhancements. The encoder blocks consist of 2 convolutional blocks of stride 1 followed by 3 convolutional blocks of stride 2. Each convolutional block follows Conv-Batchnorm-ReLU pattern. The encoder is followed by 8 Double Convolutional Residual Blocks (DCRB). The DCRBs are introduced in the lowest level to minimize the computational complexity. Each DCRB consists of two convolutional layers with skip connections. This is followed by a decoder block to up-sample the features back to the original resolution. The decoder block consists of 3 deconvolutional layers of stride 2 followed by 2 convolutional layers of stride 1. Skip connections are provided between encoder and decoder blocks to pass information between feature maps of corresponding spatial dimensions. All the skip connections are implemented using element-wise addition that reduces the computational complexity.

3.1.2 Progressive Inference

We introduce an iterative Progressive Inference (PI) scheme for estimating O^s for scales $\{2, .., N\}$ once O^1 is estimated. The output image for any scale s can iteratively be estimated as

$$O^s = \mathbf{PI}(I^s, I^{s-1}, O^{s-1}) \tag{2}$$

The $\mathbf{PI}$ function is implemented using two cascaded blocks: Convolutional Guided Filter (CGF) for up-sampling O^{s-1} and High Scale Sub-network (HSSNet) for removing reflections from each scale.

Convolutional Guided Filter: We use CGF block to up-sample the O^{s-1} using higher resolution I^s as a guide. The CGF block, originally introduced by [30], is a fast end-to-end trainable version of the classical guided filter [8]. CGF blocks have been successfully used to improve the computational efficiency of solutions in domains such as dehazing, image matting and style transfer. The flow diagram of CGF block is shown in Fig. 2. It accepts 3 images: low resolution input I^{s-1}, low resolution output O^{s-1} and high resolution input I^s to generate the high resolution output $O^{s'}$. CGF block generates much sharper output images as opposed to methods with deconvolutional filters. Moreover, the CGF block is lightweight and adds minimal computation overhead to the solution.

High Scale Sub-network: The high resolution output $O^{s'}$ generated by the CGF block needs to be further refined to generate the output image O^s for scale s. We use HSSNet for this purpose. HSSNet also follows an encoder-decoder architecture similar to LSSNet. However, since HSSNet operates on higher scales, DCRB blocks are not used in order to reduce computational complexity.

It should be noted that the weights for CGF and HSSNet are shared across all the scales. The weight sharing enables reusing the PI block iteratively over multiple scales, benefit of which is two-fold. First, weight sharing drastically reduces the number of parameters required for a N-scale space pyramid especially when N is large, hence reducing the memory footprint of the solution. Second, since the PI blocks can be reused over scales, the proposed solution can realistically remove reflections from a wide range of input resolutions without the need for retraining, by simply varying N. Moreover, the proposed solution can increase the receptive field by a factor of 2^N while the computation time increases only by $\frac{4}{3}.(1 - \frac{1}{4}^N)$ where N is the number of scales. This enables efficient reflection removal from very high resolution images while keeping the computation and memory constraints and hence can be deployed on an embedded device with ease.

The proposed scale space approach has both computational as well as performance benefits over a conventional pyramid proposed in [28] that extract features at different scales. Firstly, the method in [28] has a fixed number of MACs/pixel (9.48M) for any given input resolution. However in our method, the MACs/pixel is a function of number of scale levels N. The complexity does not scale exponentially with increase in input resolution. Further computational advantage can be obtained by simplifying the PI stage. Secondly, [28] extract pyramidal features using images of size 224x224. We found that the reflection removal performance of [28] is satisfactory up to 1MP after which it deteriorates significantly. Our method (LSSNet) even though has been trained using similar resolution (256x256), the scale space level at which LSSNet operates is dynamically chosen and hence ensures that the network has a full view of input image.

3.2. Loss Function

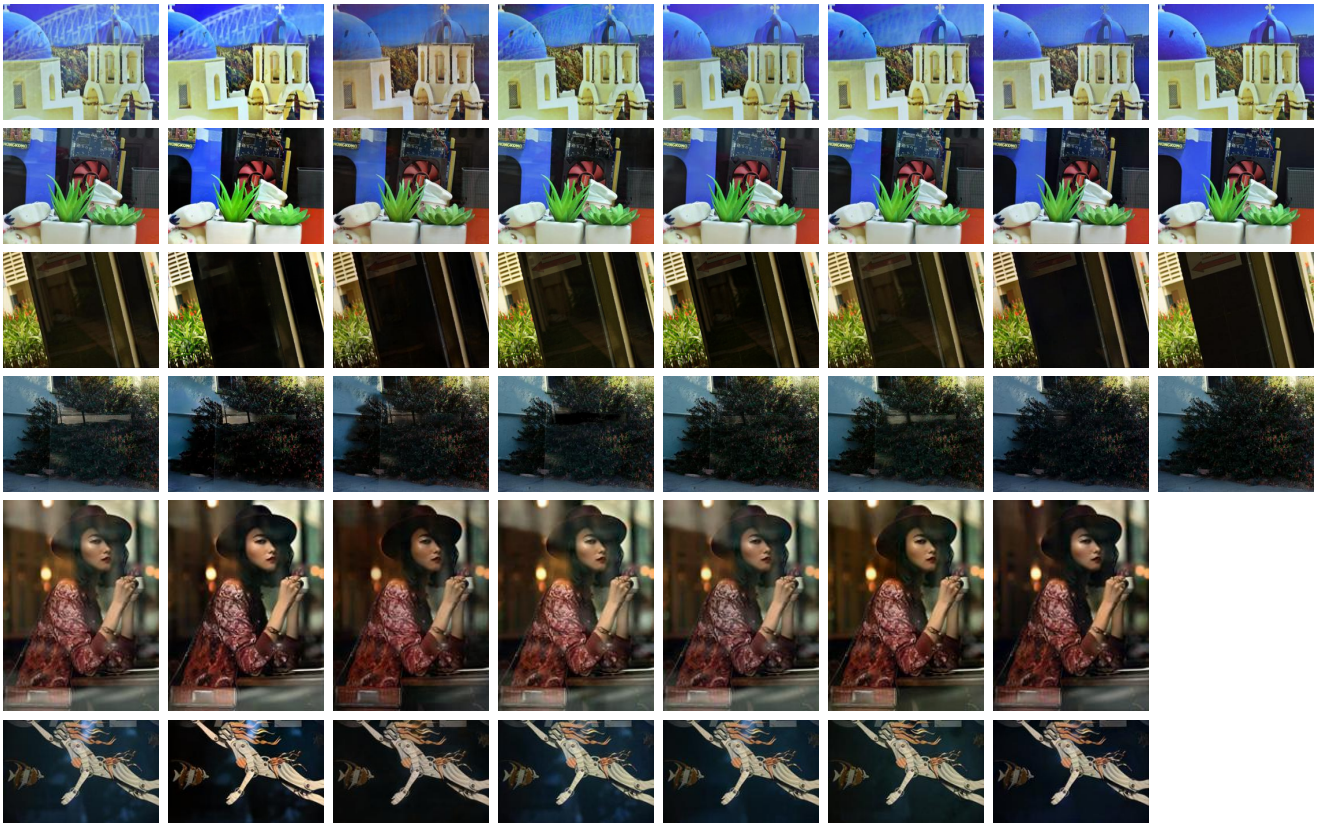
Both the sub networks LSSNet and HSSNet are trained using a combination of different loss functions that comprise of pixel and feature losses. The pixel wise intensity difference is penalized using a combination of three component losses as given in Equation 3.

$$\begin{aligned}\mathcal{L}_p = \alpha||\hat{O} - O||_2^2 + \beta||\hat{O} - O||_1 \\ + \gamma(||\nabla_x\hat{O} - \nabla_x O||_1 + ||\nabla_y\hat{O} - \nabla_y O||_1).\end{aligned} \tag{3}$$

where, ∇_x and ∇_y are the gradient operators along x and y directions respectively and $\hat{O}$ and O are respectively the estimated transmission output and ground truth. We also use contextual loss [18] given below.

$$\mathcal{L}_c = -log(\mathbf{CX}(\phi^l(\hat{O}), \phi^l(O))) \tag{4}$$

where, $\phi^l(.)$ and $\phi^l(.)$ are the feature maps extracted from layer l of the perceptual network, which in our case is VGG19 network [23]. The function $\mathbf{CX}$ defines the contextual similarity as described in [18]. The contextual loss helped in minimizing color artifacts while training with



(a) Input (b) BDN [33] (c)Zhang.et.al[34] (d) IBCLN [12] (e) ERRNet[28] (f) RAGNet[16] (g) Ours (h) Ground Truth

Figure 4: Qualitative Comparisons on Public Datasets: The columns (b) to (f) are the results from the latest state of the art methods. The column (g) shows the result from the proposed method and column (h) contains the corresponding ground truth images. The last two rows show comparisons on the Berkeley real45 dataset [34] where a reference ground truth is not available. Our method is either superior or at par with the state of the art methods with significantly lower complexity - see Table 3.

aligned data and also provided the stability during training. The total loss is a combination of both the pixel loss and contextual loss.

$$\mathcal{L} = \mathcal{L}_p + \delta\mathcal{L}_c \tag{5}$$

In our experiments, we empirically set $\alpha = 0.2$, $\beta = 0.2$, $\gamma = 0.4$ and $\delta = 0.8$.

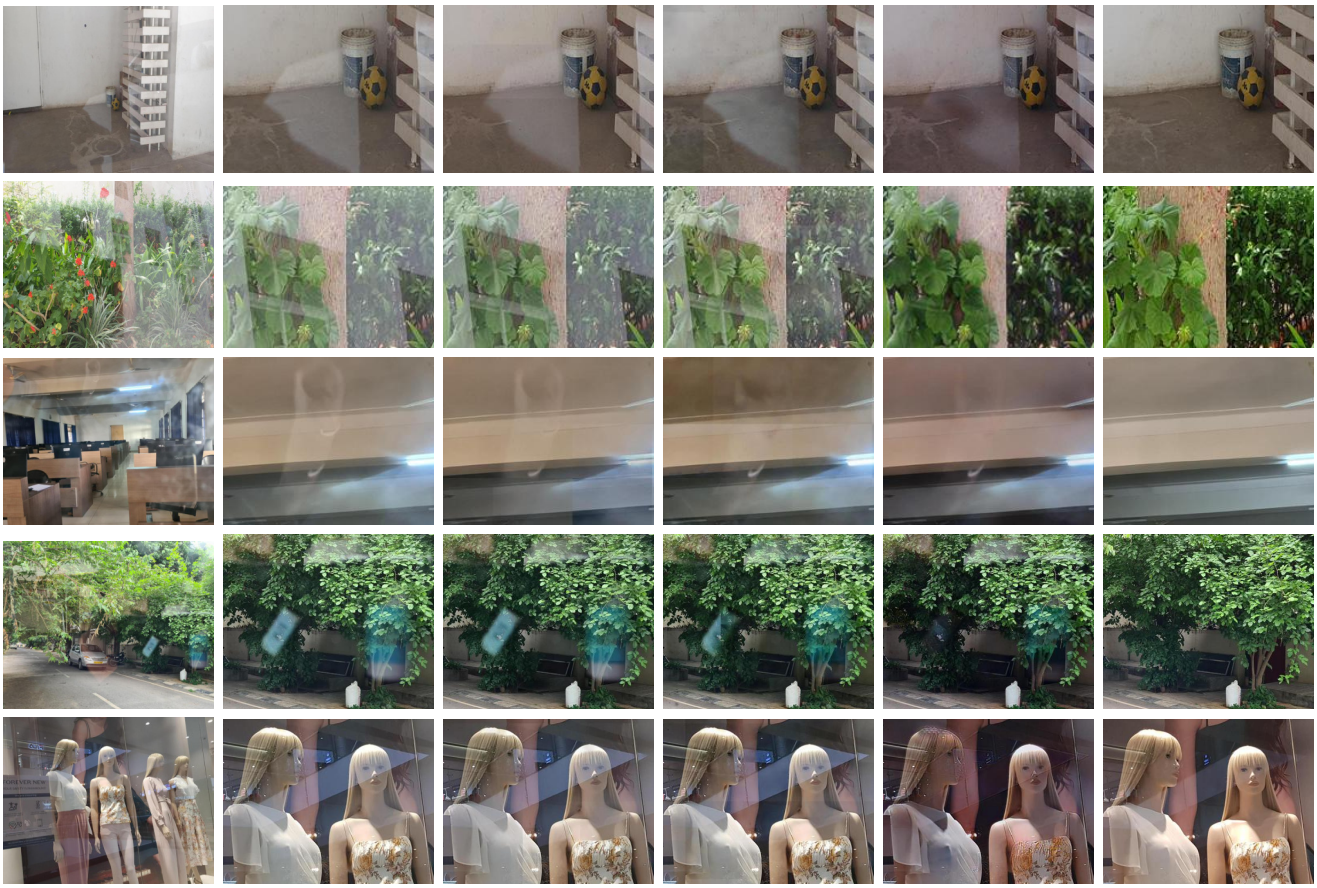
4. Experimental Results

4.1. Datasets

We generate 7400 images with synthetic reflections from the PASCAL-VOC dataset [4] using the method proposed in [28]. This dataset is used for pretraining both the subnetworks. We also use the Berkeley real dataset [34] consisting of 110 real image pairs captured using a a Canon 600D camera and a portable glass to introduce reflections. We use 90 images from this set for training, while 20 images are used for evaluation and choice of these images are similar to the strategy followed in [28]. We further use the $SIRR^2$ benchmark dataset [26] that consists of 460 image pairs split across 3 categories namely solid objects, post card and wild scenes. This dataset is exclusively used for testing purpose. The solid object and post card datasets consists of images taken in indoor controlled environment while wild scenes consists of real life scenes in unconstrained scenarios. We also capture a set of high resolution images (12MP and 64MP) using a smart-phone camera for the purpose of evaluation of our method against latest state of the art methods. Among these, several of them are captured using a portable glass to introduce reflection while the remaining sets consists of reflections in the wild such as glass walls in malls, museums, coffee shops, etc. A polarizer is used to obtain a reference ground truth image without reflection similar to [9].

4.2. Training Details

The proposed method is implemented in Pytorch running on a PC with Intel Xeon E5-2620v3 with 128GB RAM and an NVIDIA GTX 1080Ti GPU with 12GB memory for training purpose. Firstly, LSSNet is initially trained on the



(a) Input (12MP) Image. (b) Input Zoomed (c) ERRNet [28] (d) RAGNet[16] (e) Ours (g) Ground Truth.

Figure 5: Qualitative comparisons on high resolution 12MP images: The columns (a), (b) contains the input showing the entire high resolution field of view and its zoomed version. The columns (c), (d) show the results of the best performing state of the art methods [28] and [16], respectively. The column (e) shows the our results. The first 4 rows contain sets captured with and without glass, and the ground truth for the last row is captured using a polarizer, see Section 4.1. The ground truth captured using the polarizer is only used for visual reference.

PASCAL VOC synthetic dataset for about 200 epochs with a batch size of 8 using Adam optimizer. The initial learning rate was set to 0.0001. Next the output from LSSNet is generated to train HSSNet and CGF module. Both the sub networks were initialized with Xavier weights for all convolutional layers. For training, randomly cropped patches of size 256x256 with random horizontal and vertical flipping were utilized. Finally, the LSSNet is fine tuned on the Berkeley real dataset [34] for 1000 epochs to achieve model convergence. The image IDs we used for training are same as what was used in [28] [34]. During training, we resize the images depending upon the scale level N (Eq. 1) to which the training image belong to.

4.3. Qualitative Evaluation

We first provide image comparison study of the our method against state of the art methods in Fig. 4 on publicly available datasets. The methods proposed in BDN [33], Zhang [34], IBCLN [12], ERRNet [28] and RAGNet [16] are used for this comparison. In Fig. 4 the publicly available datasets by SIR Solid Object, Wild Scene, Postcard and Berkeley Real 20 and Real 45 datasets are used. It should be be noted that for Real 45 dataset, the ground truths are not provided. Our method chooses the number of levels required for inference using Eq. 1.

Next, we provide qualitative evaluation of our method on high resolution images. Figure 5 shows a comparison against the latest state of the art methods on our high resolution 12MP images. Despite the fact that the the existing state of the art methods work very well on smaller image resolutions, they fail to remove reflections on higher resolutions and their performance significantly degrades with increase in image resolution. Our method with the help of dynamic choice of the levels in the scale space is able to

Sl No	Method	Solid Object		Post Card		Wild Scene		Real20		Average	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
1	CEILNet	23.37	0.875	20.09	0.786	18.87	0.805	21.54	0.692	21.41	0.822
2	Zhang.et.al	22.68	0.879	16.81	0.797	21.52	0.832	22.55	0.788	20.10	0.836
3	BDN	22.73	0.853	20.71	0.857	22.34	0.821	18.81	0.737	21.68	0.846
4	ERRNet	24.85	0.894	21.99	0.874	24.16	0.847	23.19	**0.817**	23.51	0.877
5	IBCLN	24.88	0.893	23.39	0.875	24.71	**0.886**	22.04	0.772	24.12	0.88
6	RAGNet	26.03	0.903	23.66	0.879	25.52	0.88	21.26	0.766	24.79	0.885
7	Ours	**26.78**	**0.906**	**26.26**	**0.906**	**26.41**	0.885	**25.06**	0.816	**26.45**	**0.899**

Table 1: Quantitative Comparison on Public Datasets: Our method achieves an overall improvement of $1.5dB$ over the state of the art RAGNet [16] at significantly lower computational complexity, see Table 3. The best results are shown in boldface and the second best is underlined.

(a) Input (b) $N = 1$ (c) $N = 2$ (d) $N = 3$ (e) $N = 4$ (f) $N = 5$

Figure 6: Scale space evaluation for a ultra high resolution image of 64MP. The columns (b) - (f) show the results of varying number of levels in the scale space. The model inference with $N = 1$ comprises of LSSNet only, $N = 2$ comprises of LSSNet followed by HSSNet, $N = 3$ has LSSNet followed by two HSSNets and so on. The CGF block is used for all our experiments.

handle different image resolutions without compromising on the speed and memory requirements.

4.4. Quantitative Evaluation

We provide a quantitative comparison study of the proposed method against the state-of-the-art methods in Table 1. The study is conducted on the same methods and datasets chosen for qualitative studies. We use PSNR and SSIM as the objective metrics for the study. From Table 1, it is evident that the proposed method is able to generate output images with best PSNR scores on all the datasets considered. Moreover the proposed method is able to provide an overall improvement of $1.5dB$ on PSNR averaged over all the four datasets. The proposed scale space approach provides a full view of the input image by virtue of its large receptive field, there by enabling us to achieve superior performance on lower resolution images present in the public test sets.

4.5. Ablation Studies

In this section, we provide ablation studies for our method on the real datasets [26], [34] and [5] that are the publicly available. Firstly, network component analysis is presented which describes the advantages and disadvantages of different design strategies. Secondly, scale space analysis that evaluates the progressive improvement that is achieved with different choices of the number of levels in the scale space (N).

Sl No	Network	PSNR	SSIM
1	LSSNet Only	24.789	0.878
2	LSSNet + HSSNet	25.069	0.897
3	LSSNet + LSSNet	25.412	0.895
4	LSSNet + HSSNet + CGF	26.323	0.898
5	LSSNet + LSSNet + CGF	**26.45**	**0.899**

Table 2: Ablation study of our network architecture: The first row experiment uses only LSSNet by explicitly setting $N = 1$. The second row uses LSSNet for the lowest scale and HSSNet for higher scales for any choice of N.

4.5.1 Network Component Analysis

We evaluate the proposed network architecture by evaluating 5 different design strategies as shown in Table 2. First, we evaluate the performance when only LSSNet is used. In the subsequent experiments, we introduce different choices of networks for higher scales. And finally we evaluate the impact of CGF blocks. Our method when used with LSSNet for both the lower and higher levels of the scale space yields the best results, however using HSSNet shows only a marginal degradation in PSNR but provides an improvement of $2.5\times$ in terms of processing speed and memory requirements. It is also evident from Table 2 that the presence of CGF blocks helps in further improvement of the overall performance of the model.

Sl No	Method	No of Parameters (Millions)	MAC Operations	Processing Time (seconds)	Peak Memory (GB)	Model Size (MB)
1	Zhang.et.al	77.6	601K	0.41	2.46	381
2	BDN	75.2	3.91M	0.62	1.69	299
3	IBCLN	21.6	4.81M	0.42	1.42	83
4	ERRNet	18.9	9.46M	0.524	4.23	331
5	RAGNet	130.9	758K	0.331	2.87	560
6	Ours	**2.6**	**49K**	**0.0149**	**0.74**	**32**
7	Ours (Smart-phone)	2.6	49K	0.097	0.27	9

Table 3: Model Complexity Analysis: Comparison of different practical considerations of the proposed method against the latest state of the art. The first 6 rows show the evaluation on NVIDIA GTX 1080Ti GPU and the last row on an embedded smart-phone device. The best results are shown in boldface and the second best is underlined.

4.5.2 Scale Space Analysis

We evaluate the proposed method on very high resolution input images (64MP) for different choices of N. A choice of $N = 1$ suffers from significant degradation in reflection removal quality as shown in Figure 6. With the increase in N, the overall quality progressively improves and eventually saturates at $N = 5$. An intelligent choice of N as described in Equation 1 for a given input image resolution yields optimal results.

4.6. Complexity Evaluation of Proposed Method

A detailed comparison of different practical considerations against state of the art methods is shown in the Table 3. We evaluated several aspects such as processing time, peak memory consumption, number of multiply and accumulate (MAC) operations and also the no of the learnable parameters. It is quite evident from the Table 3 that our method is at least $20\times$ faster in terms of processing time which has been achieved using a model with least number of learnable parameters and MAC operations. The proposed method being light weight is also suitable for deploying on low memory devices to achieve real time performance.

4.7. Embedded Device Deployment

In order to deploy the proposed method on a low power device such as a smart-phone, we first quantize the models of both the sub networks using the publicly available Qualcomm Neural Processing SDK for AI. The quantized models are then deployed on a smart-phone with a supporting system on chip. In our experiments, we used the latest available SDK version from Qualomm and deployed on a android smart-phone. The quantized model shows a marginal quality degradation that is visually very similar to the non-quantized model output as shown in Figure 7. We used Qualcomm's enhanced quantizer that uses a proprietary algorithm to determine the optimal range and is especially useful for models with long tail distribution of the data being quantized. The quantized model is able to achieve a remarkable performance of 97 msecs for a tile of size 540x400 for $N = 2$ levels. A high resolution image (Ex: 12MP) takes roughly about 5 seconds of processing time on a smart-phone device.

Figure 7: Qualitative Evaluation of quantized model outputs generated on a smart phone. The enhanced quantizer shows image quality comparable to non-quantized output.

5. Conclusions

In this paper, we propose a novel, light-weight scale space architecture for single image reflection removal. To reduce the number of computations, we use a deeper architecture only at the lowest scale while the higher scales are processed using shallower networks. We use convolutional guided filters to upsample lower scale outputs to provide as guide to higher scales. We also share the weights between the sub networks used in the higher levels which helps reduce the memory. The scale space architecture along with shared weights enables us to increase the effective receptive field during inference and hence our method generalizes well to high resolution images. We have shown that our method can remove reflections on very high resolution images (even 64 MP) even though the network was trained on much smaller resolutions. Moreover, our method out performs the state of the art methods both qualitatively and quantitatively and also runs $20\times$ faster with $50\times$ less parameters than the most recent state of the art algorithm RAGNet [16]. We also implemented a quantized version of our solution on an android smart-phone powered by Qualcomm snapdragon 888 chipset with 8 GB on-board RAM where a high resolution 12MP is restored in under 5 seconds.

References

[1] Jean-Baptiste Alayrac, Joao Carreira, and Andrew Zisserman. The visual centrifuge: Model-free layered video representations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2457–2466, 2019. 2

[2] Paramanand Chandramouli, Mehdi Noroozi, and Paolo Favaro. Convnet-based depth estimation, reflection separation and deblurring of plenoptic images. In *Asian Conference on Computer Vision*, pages 129–144. Springer, 2016. 2

[3] Yun-Chung Chung, Shyang-Lih Chang, Jung-Ming Wang, and Sei-Wang Chen. Interference reflection separation from a single image. In *2009 Workshop on Applications of Computer Vision (WACV)*, pages 1–6. IEEE, 2009. 2

[4] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010. 5

[5] Qingnan Fan, Jiaolong Yang, Gang Hua, Baoquan Chen, and David Wipf. A generic deep architecture for single image reflection removal and image smoothing. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3238–3247, 2017. 2, 3, 7

[6] Kun Gai, Zhenwei Shi, and Changshui Zhang. Blind separation of superimposed moving images using image statistics. *IEEE transactions on pattern analysis and machine intelligence*, 34(1):19–32, 2011. 2

[7] Xiaojie Guo, Xiaochun Cao, and Yi Ma. Robust separation of reflection from multiple images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2187–2194, 2014. 2

[8] Kaiming He, Jian Sun, and Xiaoou Tang. Guided image filtering. *IEEE transactions on pattern analysis and machine intelligence*, 35(6):1397–1409, 2012. 4

[9] Chenyang Lei, Xuhua Huang, Mengdi Zhang, Qiong Yan, Wenxiu Sun, and Qifeng Chen. Polarized reflection removal with perfect alignment in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1750–1758, 2020. 5

[10] Anat Levin, Assaf Zomet, and Yair Weiss. Learning to perceive transparency from the statistics of natural scenes. In *Advances in Neural Information Processing Systems*, pages 1271–1278, 2003. 1

[11] Anat Levin, Assaf Zomet, and Yair Weiss. Separating reflections from a single image using local features. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, volume 1, pages I–I. IEEE, 2004. 1, 2

[12] Chao Li, Yixiao Yang, Kun He, Stephen Lin, and John E Hopcroft. Single image reflection removal through cascaded refinement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3565–3574, 2020. 1, 2, 5, 6

[13] Tingtian Li, Daniel PK Lun, Yuk-Hee Chan, et al. Robust reflection removal based on light field imaging. *IEEE Transactions on Image Processing*, 28(4):1798–1812, 2018. 2

[14] Yu Li and Michael S Brown. Exploiting reflection change for automatic reflection removal. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2432–2439, 2013. 2

[15] Yu Li and Michael S Brown. Single image layer separation using relative smoothness. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2752–2759, 2014. 2

[16] Yu Li, Ming Liu, Yaling Yi, Qince Li, Dongwei Ren, and Wangmeng Zuo. Two-stage single image reflection removal with reflection-aware guidance. *arXiv preprint arXiv:2012.00945*, 2020. 1, 2, 3, 5, 6, 7, 8

[17] Yu-Lun Liu, Wei-Sheng Lai, Ming-Hsuan Yang, Yung-Yu Chuang, and Jia-Bin Huang. Learning to see through obstructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14215–14224, 2020. 2

[18] Roey Mechrez, Itamar Talmi, and Lihi Zelnik-Manor. The contextual loss for image transformation with non-aligned data. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 768–783, 2018. 4

[19] Seungjun Nah, Tae Hyun Kim, and Kyoung Mu Lee. Deep multi-scale convolutional neural network for dynamic scene deblurring. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3883–3891, 2017. 2

[20] Abhijith Punnappurath and Michael S Brown. Reflection removal using a dual-pixel sensor. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1556–1565, 2019. 2

[21] 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. 4

[22] YiChang Shih, Dilip Krishnan, Fredo Durand, and William T Freeman. Reflection removal using ghosting cues. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3193–3201, 2015. 1, 2, 3

[23] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 4

[24] Chao Sun, Shuaicheng Liu, Taotao Yang, Bing Zeng, Zhengning Wang, and Guanghui Liu. Automatic reflection removal using gradient intensity and motion cues. In *Proceedings of the 24th ACM international conference on Multimedia*, pages 466–470, 2016. 2

[25] Xin Tao, Hongyun Gao, Xiaoyong Shen, Jue Wang, and Jiaya Jia. Scale-recurrent network for deep image deblurring. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8174–8182, 2018. 2

[26] Renjie Wan, Boxin Shi, Ling-Yu Duan, Ah-Hwee Tan, and Alex C Kot. Benchmarking single-image reflection removal algorithms. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3922–3930, 2017. 2, 5, 7

[27] Renjie Wan, Boxin Shi, Ling-Yu Duan, Ah-Hwee Tan, and Alex C Kot. Crrn: Multi-scale guided concurrent reflection

removal network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4777–4785, 2018. 2, 3

[28] Kaixuan Wei, Jiaolong Yang, Ying Fu, David Wipf, and Hua Huang. Single image reflection removal exploiting misaligned training data and network enhancements. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8178–8187, 2019. 1, 2, 3, 4, 5, 6

[29] Qiang Wen, Yinjie Tan, Jing Qin, Wenxi Liu, Guoqiang Han, and Shengfeng He. Single image reflection removal beyond linearity. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3771–3779, 2019. 3

[30] Huikai Wu, Shuai Zheng, Junge Zhang, and Kaiqi Huang. Fast end-to-end trainable guided filter. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1838–1847, 2018. 2, 4

[31] Tianfan Xue, Michael Rubinstein, Ce Liu, and William T Freeman. A computational approach for obstruction-free photography. *ACM Transactions on Graphics (TOG)*, 34(4):1–11, 2015. 2

[32] Qing Yan, Yi Xu, and Xiaokang Yang. Separation of weak reflection from a single superimposed image using gradient profile sharpness. In *2013 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 937–940. IEEE, 2013. 1, 2

[33] Jie Yang, Dong Gong, Lingqiao Liu, and Qinfeng Shi. Seeing deeply and bidirectionally: A deep learning approach for single image reflection removal. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 654–669, 2018. 3, 5, 6

[34] Xuaner Zhang, Ren Ng, and Qifeng Chen. Single image reflection separation with perceptual losses. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4786–4794, 2018. 1, 2, 3, 5, 6, 7