

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/366616425>

A Dual CNN Architecture for Single Image Raindrop and Rain Streak Removal

Conference Paper · December 2022

DOI: 10.1109/ICITR57877.2022.9993091

CITATIONS

0

READS

13

2 authors:



Anparasy Sivaanpu
University of Vavuniya

6 PUBLICATIONS 3 CITATIONS

[SEE PROFILE](#)



Kokul Thanikasalam
University of Jaffna

26 PUBLICATIONS 60 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



Underwater Image Enhancement [View project](#)



A Comprehensive Study on Deep Image Classification with Small Datasets [View project](#)

A Dual CNN Architecture for Single Image Raindrop and Rain Streak Removal

Anparasy Sivaanpu
Department of Physical Science
University of Vavuniya
Sri Lanka
s.anparasy@vau.ac.lk

Kokul Thanikasalam
Department of Computer Science
University of Jaffna
Sri Lanka
kokul@univ.jfn.ac.lk

Abstract—Visual quality of rainy images are considerably poor due to the raindrops in camera lens and the rain streaks in the background scenes. Although the raindrops and rain streaks are appeared together in real-world rainy images, most of the previous approaches are proposed to remove either of them. In this paper, we have proposed a novel CNN model architecture to remove raindrops and rain streaks together. The proposed CNN model architecture has two branches and it consumes two formats of a rainy image via an encoder-decoder network and a dense CNN network. At the end of the architecture, outputs of both branches are combined to produce a high-visibility rain-free image with natural colours. In addition, internal and external skip connections are introduced in the blocks of these branches to improve the performance further. The proposed model is trained and then tested on RainDrop, Rain100H, Rain100L, and Rain12 benchmarks and showed excellent performance than the state-of-the-art approaches.

Keywords—Raindrop removal, rain streak removal, De-raining, Convolutional Neural Network

I. INTRODUCTION

Bad weather conditions are significantly degrading the visibility of outdoor scenes and rain is the primary cause. Poor visibility of outdoor images in rainy sessions is the major challenge for many computer vision based automated systems such as autonomous driving[1], vehicle tracking[2], traffic monitoring[2], and visual surveillance[3]. Raindrops and rain streaks significantly reduce the quality of images and videos by dropping or blocking on camera lens, and creating a fogging appearance, respectively. Automated rain removal systems are proposed using computer vision based algorithms and they are objective to improve the visual quality of single rainy images by enhancing and restoring the colour and details.

Raindrops and rain streaks are the major causes for the poor quality in outdoor images. Raindrops blocks the camera lens and hence irregular geometric structures are appeared on images and then the image is captures with blurred scenes. In addition, they reduced the contrast of the image and fade the natural colours. On the other hand, rain streaks create a foggy appearance in the image by blurring the background and hence the visibility is reduced. Moreover, in nearby scenes, individual rain streaks are significantly visible and causes visibility degradation in rainy images. Generally, raindrops and rain streaks are simultaneously appearing in an image and they cause change of an object’s shape and occlusion in a scene, respectively. Differences of raindrops and rain streaks in rainy images are illustrated in Fig.1. Due to the differences of their physical appearance, most of the researchers individually investigated the raindrops and rain streaks in images and then proposed the separate models for automated raindrop removal and rain streak removal.

In recent years, a considerable numbers of automated raindrop removal approaches[4-12] are proposed. These

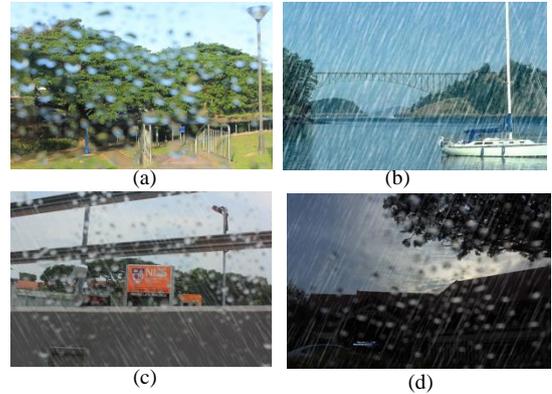


Fig. 1: Poor visibility of rainy images. (a) Blurred scene by raindrops, (b) Rain streaks are occluded in a scene, (c) & (d) quality degradation by both raindrops and rain streaks in single images.

approaches are objective to eliminate the raindrops and then produce a high-quality image with clear background. Some of the early approaches[11] proposed their models based on the physical imaging characterises of raindrops such as light reflection and camera focus. However, most of the recent raindrop removal approaches did not consider the physical imaging factors and rely only on deep learning based algorithms. In these approaches, Generative Adversarial Networks (GANs)[4, 6, 10, 12] and Convolutional Neural Networks (CNNs)[5, 7-9, 13] are used to produce a clear raindrop-free image in an end-to-end manner. Most of the researchers use a publicly available raindrop benchmark dataset[4] to evaluate their performances.

In the past decade, several rain streak removal approaches[14-25] are proposed. In these approaches, a rainy image (R) with rain streaks are considered and modelled as follows:

$$R = S + B \quad (1)$$

where S and B are the rain streak layer and B is clean background layer, in that rainy image. Based on that, researchers considered rain streak removal as separating layer B from layer S for a given input R . Although early approaches depending on image processing based algorithms to the separate streak layer from the background layer, most of the recent approaches[14] are rely on deep learning based solutions such as CNNs[14, 16, 18-21, 23-25] and GANs[17]. There are three well-known benchmark datasets[26, 27] are available to evaluate the performance of rain streak models.

In real-world situations, raindrops and rain streaks are regularly appeared together as shown in Fig.1 (c) and (d). Due to this fact, a few numbers of approaches[26-31] are tried to provide a joint solution for both problems. In most situations, these approaches proposed a single model to remove the raindrops and rain streaks. Since there are no benchmark datasets available with raindrops and rain streaks together in

images, these approaches evaluated their performances in raindrop and rain streak benchmarks in an individual manner.

To handle the real-world situations, a robust rain removal framework should remove the raindrops and rain streaks from single images in real-time. Also, they should restore the details and natural colours. To achieve these objectives, we have proposed a novel CNN model architecture to remove the raindrops and rain streaks together. The proposed architecture has two CNN branches to costumes two different images and then produces a clear rain-free image. A raw rainy input image and an its colour enhanced image are fed to the proposed CNN model architecture to obtain a rain-free image with missing details and natural colours. The performance of the proposed CNN model architecture is individually evaluated on publicly available raindrop removal and rain streaks removal datasets. Based on the experimental results, our proposed CNN model architecture qualitatively and quantitatively outperforms the existing approaches on these benchmarks and real-world images.

II. RELATED WORK

The most recent and related works are reviewed in this section since several experimental reviews[32, 33] are available on single image raindrop removal and rain streak removal.

A. Filter and Prior Based Approaches

Most of the early raindrop removal and rain streak removal approaches are proposed based on either filter based or prior based techniques. In filter based approaches, an image filter is used to separate the rain layer from the background layer. Xu et al., [34] used the Guided filter to remove the rain and snow from a single image. Kim et al., [35] used non local means filter in rain removal based on the aspect ratio and rotation angle of the elliptical kernel in a rainy image. Rain removal performance of filter based approach are considerably poor since they have applied the same filter in all local image regions without considering the density variations of rain layer.

In prior based approaches, physical and imaging characterises of rain in an image are analysed and then that prior information is used to model the rainy images. Thereafter, that rainy image prior model is used to decompose the image into rain and background layers based on the physical properties of the rain. Tan et al., [26] proposed a model based on Gaussian mixture and it can able to separate the rain layer in local regions by considering the orientation and size of rain streaks. Luo et al., [15] proposed a non-linear decomposing technique by creating a discriminative dictionary in local image patches. Although prior based approaches are performed better than the filter based approaches, their outputs are not in satisfactory level since they are depending on hand-crafted features and image priors.

B. Deep Learning Based Rain Removal Methods

In recent years, deep learning based techniques and architectures are became popular to remove the raindrops and rain streaks in a single rainy image. These approaches are able to remove the rain layer from background layer from a raw rainy image in an end-to-end manner. Several novel CNN model architectures and GAN architectures are proposed to achieve the high performance.

In the last five years, several deep learning based rain drop removal approaches are proposed. Qian et al., [4] proposed an attentive generative adversarial network to learn the raindrop regions and structures through an adversarial learning process. A novel aggregation CNN model architecture is proposed by Lin et al., [5]. They have converted a RGB rainy image to YUV and then fed it to the CNN model to remove the raindrops. Nauyen and Lee [6] used an enhanced version of GAN in raindrop removal by utilizing several prior information. The same authors [12] proposed another approach based on the multi task GAN and also used the segmentation to improve the raindrop removal process. Similar to that approach, Zhang et al., [10] used a conditional GAN architecture to generate the rain-free image. Hao et al., [7] generated synthetic raindrop images to handle the data deficiency and then used a CNN model to generate raindrop-free image. Porav et al.,[8] restored the missing details in a raindrop image by using a denoising generator. In [9], a faster rain removal approach is proposed using a pixel-wise filtering technique on a CNN model. Zhang and Patel [13] estimated the density of an rainy image using an residual CNN architecture and then used that information in rain removal. Most of these approaches are used GAN as their major deep learning architecture. Since GANs are having two subnetworks with different loss functions, these raindrop removal approaches are facing non-convergence problem and hence their performances are limited at a level.

Another set of researchers proposed various deep learning based models for rain streak removal. Fu et al.,[14] included several image filtering techniques in a CNN model to train the architecture. A novel spatial attentive CNN model is used in [16]. Wei et al., [17] trained a transfer learning based GAN model, called as DerainCycleGAN, in unsupervised manner. Similar to this approach, a semi-supervised approach is proposed in [20] using synthetic and non-synthetic images. Another semi-supervised approach is proposed in [24] with similar CNN model and a distillation network. In [18], rain streaks removal is considered as a multiple streak layer removal problem, based on their size and orientation, and then a recurrent neural network is used in removal process. Fan et al.,[19] used a cascade CNN model architecture in rain streak removal. Wang et al., [23] modelled the physical characterises of rain streaks and then combined it with a CNN model in an end-to-end manner. Yasarla and Patel [25] proposed an approach by measuring the location information of rain streaks and then trained a CNN model.

C. Identified Research Gap in Existing Approaches

Deep learning based raindrop and rain streaks removal approaches showed excellent rain removal performances in benchmark datasets than the filter and prior based approaches because of their hierarchical learning capability. However, most of these approaches are suitable for either raindrop removal or rain streak removal and not suitable to remove both. In addition, since a massive dataset is not available so far, data deficiency is the major issue to train deeper models and hence they are struggling to optimize their models with fewer number of samples. Moreover, most of the CNN based models produces rain-free images with unnatural colours since they lost the spatial information in deeper convolutional layers. Based on these issues, most of the approaches are not able to output a high quality rain-free image beyond some extent.

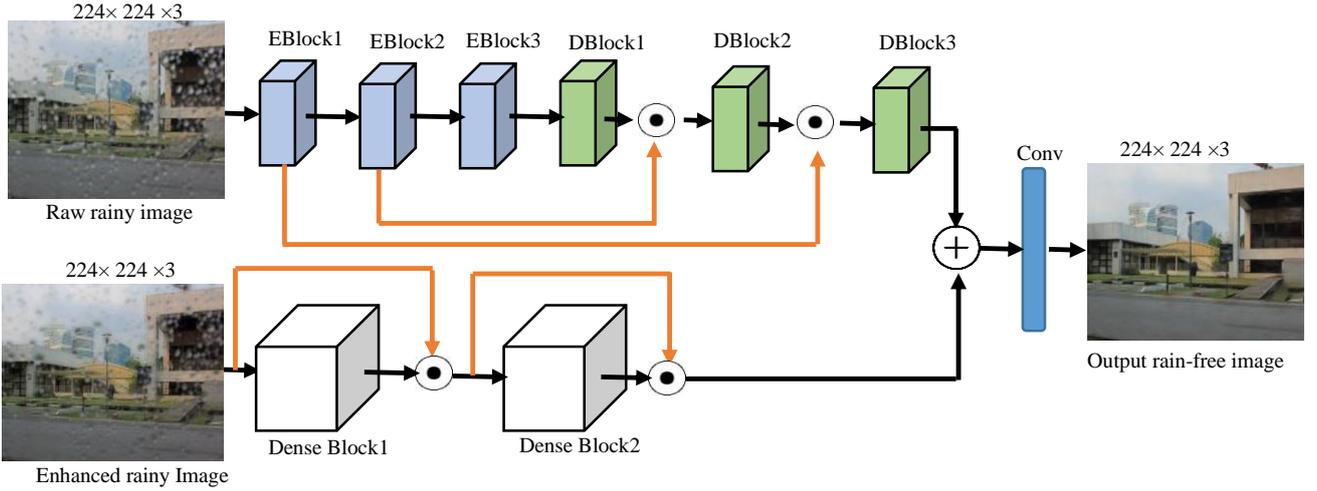


Fig.2: Block diagram of the proposed CNN model architecture. Raw rainy image is fed by encoder and decoder (denoted as EBlock and DBlock, respectively) network branch and guided filter based enhanced image is fed via Dense CNN branch. Skip connections (denoted as orange lines) are introduced in both branches to optimize the model with fewer samples. Finally, outputs of both branches are fused and then a convolutional layer is used to produce the final output image. \oplus and \odot symbols are denoting element-wise addition and concatenation operations, respectively.

III. METHODOLOGY

We have proposed a dual CNN model architecture for single image raindrop and rain streak removal. The proposed architecture consumes two forms of a same input image through two CNN branches and then produces a rain-free image with realistic colours and better visibility. In the first branch, a raw rainy image is fed and then processed by an encoder-decoder CNN architecture. Simultaneously, a weighted median filter and then a guided filter are used to remove the low frequency noise components in the same input image and then the obtained output is fed to the second CNN branch. A dense CNN architecture is used in the second branch. The first and second CNN branches are objective to restore the details and enhance the realistic colours, respectively. In both branches, skip connections are introduced to keep the spatial information of rainy image and to optimize the model with a smaller number of training samples. Finally, outputs of both branches are fused and then a convolutional layer is used to produce the final output image with better visibility and more natural colours. The block diagram of the proposed CNN model is shown in Fig.2. Each step of the proposed method is explained in the following sections.

A. Image enhancement with Weighted Median and Guided Filters

We have used the guided filter to remove the low frequency components of a rainy image by considering the raindrops and rain streaks pixels as noises. In this process, a weighted median guided filter is used to enhance the visibility and quality of a rainy images without damaging the geometrical details. As the first step of enhancement, a weighted median filter is applied with the window size of 3×3 . Based on the maximum and minimum pixels values of the window, noisy pixel is identified and then its pixel value is replaced by the weighted median value of that window. Suppose several noisy pixels are identified, then the size of window is increased to 5×5 and then the median pixel value is obtained. Although, the median filtering process removes the tiny noises in a rainy image, some texture information and edge details are also removed.

In the second part of the image enhancement, we have used the guided filter [36] to recover the missing texture and edge details in the median filtered image. The raw rainy image and median filtered image are fed to the guided filter as the input and guided images, respectively. The guided filter produces the output image with sharp edges by filtering the input image based on the structure of the guidance image. The guided filtered image is fed to the second branch of the proposed CNN architecture since it has more sharpen edges and natural colours than the raw image.

B. Proposed CNN Model Architecture

Proposed architecture has two CNN branches and it consumes raw rainy image and corresponding enhanced image by an encoder-decoder branch and a dense CNN network branch, respectively. As illustrated in Fig.2, both input images are resized to $224 \times 224 \times 3$ and then fed to the CNN model architecture to produce an output with the same size.

We have used the encoder-decoder CNN branch to generate the pixel values of the output rain-free image. In this branch, the encoder network is objective to capture the discriminative features of the rainy image and then the

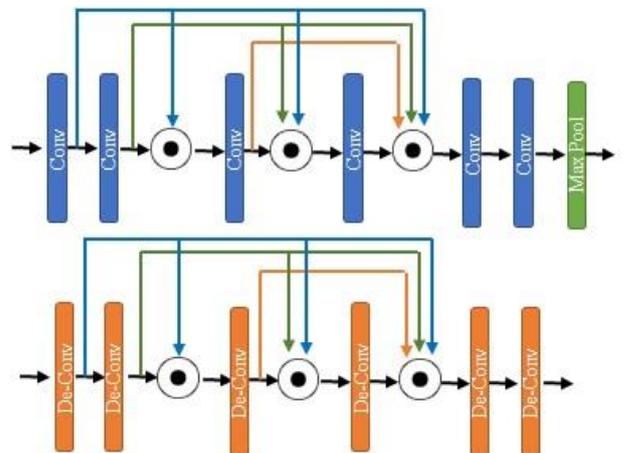


Fig.3: Top: Diagram of an Encoder block, Bottom: Block diagram of a decoder block. Convolutional and De-convolutional (transposed convolutional) layers are denoted as Conv and De-conv, respectively.

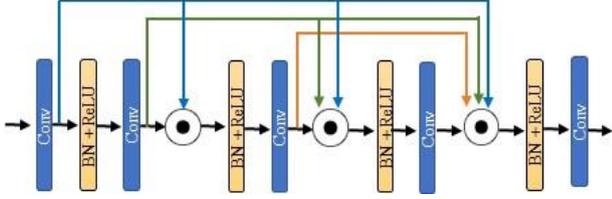


Fig. 4: Diagram of a Dense CNN block. BN and ReLU are denoting batch normalization layer and Rectified Linear Unit activation function.

decoder network is objective to project the learnt features in pixel space. The proposed encoder-decoder network branch has three encoder blocks and three decoder blocks. The block diagram of an encoder and decoder blocks are shown in Fig.3. We have used five convolutional layers in encoder blocks and same number of de-convolutional (transposed convolutional) layers in the decoder block. In both blocks, a 3×3 kernel is used to capture the features in each layer with the stride size of 1. We kept 32 channels in each convolutional and de-convolutional layer of both blocks. We have utilized the ReLU function to activate the features in these layers. At the end of each encoder blocks, a max Pooling layers is used to eliminate the minor details and noises of the features. The total number of encoder-decoder blocks are set experimentally based on the validation results.

We have used the dense CNN blocks in the second branch of the proposed model architecture. This branch is objective to produce the rain-free image with natural and realistic colours. Similar to the encoder blocks, five convolutional layers are included in each dense block. To enable faster and independent learning, batch normalization layer is placed in between the convolutional layers. A 3×3 kernel is used in these blocks with the filter size of 32. In this branch, total number of dense blocks are identified based on the validation results.

In both branches of the proposed CNN model architecture, external skip connections are introduced in between encoder-decoder blocks and in between dense blocks as shown in Fig.2. Moreover, internal skip connections are introduced in between individual layers of encoder-decoder blocks and dense blocks as shown in Fig.3 and Fig.4, respectively. In these blocks, the n^{th} layer consumes all previous layers' feature maps and concatenate them along the channel dimension. Then an activation function is used to obtain the output feature map of that layer. Because of this internal and external skip connections, proposed CNN model is able to optimize with fewer number of samples and able to preserve the spatial information in deeper layers.

In the deeper part of the proposed CNN model architecture, outputs of both branches are fused by an element-wise addition operation. Then a single convolutional layer is used to learn the combined features and then to produce final output image.

C. Training

As the first step of training, all the hyper parameters are finetuned such as batch size, filter size, number of blocks, number of layers in a block, and learning rate. Then, we have trained the proposed model for a fixed number of iterations and the best performing model was identified based on the validation accuracy. In each training iteration, the loss between obtained output rain-free image (I^{out}) and corresponding reference image (I^{ref}) is calculated as follows:



Fig. 5: Rainy image and corresponding reference image pairs of benchmark datasets. (first row - first column): RainDrop benchmark, (first row - second column): Rain100L benchmark, (second row - first column): Rain12 benchmark, and (second row second column): Rain100H benchmark.

$$\text{Loss} = \frac{1}{224 \times 224} \sum_{i=1}^{224} \sum_{j=1}^{224} [I_{(i,j)}^{out} - I_{(i,j)}^{ref}]^2 \quad (2)$$

IV. RESULTS AND DISCUSSION

A. Implementation Details

The Google Colab cloud platform is used to develop the code and Keras-TensorFlow library is utilized. The Adam optimizer is used in the training and the model is trained for 100 iterations with the batch size of 32. The code is uploaded at <https://github.com/RPRO5/Derain>

B. Details of the Benchmark Datasets

Similar to other researchers, the proposed model is trained and tested on publicly available benchmark datasets such as Rain100H[27], Rain100L[27], Rain12[37], and RainDrop dataset[4]. All of these datasets are having a raindrop or rain streak image and the corresponding reference rain-free image. The details of these benchmarks are summarized in Table 1 and same samples are shown in Fig.5.

TABLE 1: DETAILS OF THE BENCHMARK DATASETS USED IN THIS STUDY

Dataset	No. of Images	Type	Resolution
Rain12	12	Rain Streak	481×321
Rain100H	1800 Training, 200 Testing	Rain Streak	481×321
Rain100L	1800 Training, 200 Testing	Rain Streak	481×321
RainDrop	861 Training, 307 Testing	Raindrop	720×480

C. Evaluation Metrics

We have used two different metrics to evaluate the performance of the proposed model. The first metric is Structural Similarity Index (SSIM) and its gives high score whenever the output image (I^{out}) is closer to the reference image (I^{ref}). It is calculated as follows:

$$\text{SSIM} = \frac{(2\mu_r\mu_o + C_1)(2\sigma_{r_o} + C_2)}{(\mu_r^2 + \mu_o^2 + C_1)(\sigma_r^2 + \sigma_o^2 + C_2)} \quad (3)$$

where μ_r and μ_o are the mean of reference image and output image, respectively. σ_o^2 and σ_r^2 are the variance of the output and reference images, respectively. C_1 and C_2 are the variables. The second metric is Peak Signal-to-Noise Ratio (PSNR) and also gives higher values when an output is in high quality. PSNR score is calculated as follows:

$$\text{PSNR} = 10 \log_{10} \left(\frac{\text{MAX}_{I^{ref}}^2}{\text{MSE}} \right). \quad (4)$$

where MSE is the Means Square Error and it is calculated as stated in Equation 1 and $MAX_{I_{ref}}$ represents the maximum pixel value of the reference image.

D. Testing Results

Performance of the proposed approach is evaluated on the test set of benchmark datasets and then compared with the state-of-the-art approaches. To compare the results with other researchers, we have used the same set of test images in evaluation from each datasets as stated in Table 1. Since there are no datasets available to evaluate the raindrops removal and rain streak removal performances together in a single image, we have evaluated the performance of proposed method in raindrop and rain streak removal benchmarks individually without changing any model parameters. Raindrop removal performance of the proposed approach is evaluated and compared on RainDrop dataset, and the results are summarized in Table 2.

TABLE 2: PERFORMANCE COMPARISON ON RAINDROP BENCHMARK

Approach	SSIM	PSNR
Ours	0.95	32.60
Lin et al., [5]	0.93	30.79
Kaihao et al., [31]	0.93	30.63
Liu et al., [30]	0.92	31.24
Duc et al., [6]	0.92	31.56
Duc et al., [12]	0.92	31.57
Hao et al., [7]	0.91	30.17
Qian et al., [4]	0.90	29.57
Horia et al., [8]	0.90	31.55
Ren et al., [29]	0.90	29.46
Qiag et al., [9]	0.90	28.48
Fu et al., [28]	0.84	25.23
Yang et al., [27]	0.82	27.52
He et al., [13]	0.80	24.76
Li et al., [26]	0.78	24.85
Zhang et al., [10]	0.73	21.35

Rain streak removal performance of the proposed method is given in Table 3 based on the test results on Rain100H, Rain100L, and Rain12 datasets. Same set of test images are used in this comparison.

TABLE 3: PERFORMANCE COMPARISON ON RAIN STREAK BENCHMARKS

Approach	Rain100H		Rain100L		Rain12	
	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR
Ours	38.91	0.98	38.91	0.98	37.46	0.97
[29]	37.45	0.97	37.45	0.97	36.66	0.96
[27]	36.61	0.97	36.61	0.97	33.92	0.95
[22]	31.27	0.94	39.95	0.97	-	-
[13]	29.45	0.93	37.41	0.95	-	-
[28]	32.38	0.93	32.38	0.93	34.04	0.93
[25]	25.93	0.85	36.34	0.97	-	-
[19]	25.25	0.84	33.16	0.96	29.45	0.94
[16]	25.11	0.83	35.33	0.97	35.85	0.96
[14]	15.33	0.74	30.24	0.93	31.24	0.94
[20]	22.47	0.72	32.37	0.93	34.02	0.93
[24]	17.51	0.62	25.93	0.72	-	-
[23]	18.21	0.54	27.52	0.76	-	-
[17]	17.91	0.53	29.75	0.84	35	0.96
[21]	14.62	0.45	28.54	0.85	33.1	0.93
[26]	15.05	0.43	28.66	0.86	32.02	0.86
[15]	13.77	0.32	27.34	0.85	30.07	0.87

E. Discussion

It is clearly seen that proposed method outperforms the existing raindrop and rain streak removal approaches in publicly available benchmark datasets. It gains the rain-removal knowledge from a dual CNN architecture. To justify



Fig. 6: First row: Real-world images which are having raindrops and rain streaks together, Second row: Corresponding outputs of the proposed model.

the design of the architecture, we have evaluated the performance of individual branches in the RainDrop dataset and reported the results in Table 4.

TABLE 4: PERFORMANCES OF INDIVIDUAL BRANCHES

CNN Architecture	SSIM	PSNR
Encoder-Decoder Branch only	0.83	28.62
Dense Block Branch only	0.89	30.76
Proposed Architecture with both branches	0.95	32.60

Based on the results, we can clearly see that Dense block branch contributed more than the encoder-decoder branch in the proposed model architecture. In addition, it is seen that the combined dual CNN architecture produces better results than the individual branches. We have also evaluated the performances of skip connections in the blocks of these branches and the results are reported in Table 5.

TABLE 5: COMPARISONS OF SKIP CONNECTIONS

CNN Architecture	SSIM	PSNR
Dual CNN architecture without any skip connections	0.81	28.28
Dual CNN architecture with internal skip connections	0.86	29.39
Dual CNN architecture with internal and external skip connections	0.95	32.60

Based on the results, we can understand that internal and external skip connections in the blocks of these branches are used to increase the performance significantly. In addition to this quantitative comparison, qualitative performance of proposed method is tested on few real-world images which are having the raindrops and rain streaks together. As shown in Fig.6, proposed model is successfully able to remove the raindrops and rain streaks in a single image and produces the output with high visibility and natural colours.

V. CONCLUSION

In this study, we have proposed a dual CNN architecture for raindrop and rain streak removal. Since most of the previous approaches are focused on either raindrop removal or rain streak removal, this study objective to develop a model to handle the both issues. The proposed CNN model architecture has two branches: encoder-decoder network, and dense block network. We fed raw rainy image and its enhanced image in these branches and then their outputs are fused to obtain a high-quality rain-free image with more realistic colours. Proposed approach trained and tested on RainDrop, Rain100H, Rain100L, and Rain12 benchmarks

datasets and showed 0.95, 0.98, 0.98, and 0.97 of structural similarity indexes, respectively. Qualitative evaluation on some real-world images showed that the proposed model significantly removed the rain drops and rain streaks together from the single images by combining the outputs of encoder-decoder network and dense CNN blocks.

REFERENCES

- [1] K. Wang, L. Chen, T. Wang, Q. Meng, H. Jiang, and L. Chang, "Image Deraining and Denoising Convolutional Neural Network For Autonomous Driving," in International Conference on High Performance Big Data and Intelligent Systems pp. 241-245, 2021.
- [2] C. H. Bahnsen and T. B. Moeslund, "Rain removal in traffic surveillance: Does it matter?," IEEE Transactions on Intelligent Transportation Systems, vol. 20, no. 8, pp. 2802-2819, 2018.
- [3] M. Li, X. Cao, Q. Zhao, L. Zhang, and D. Meng, "Online rain/snow removal from surveillance videos," IEEE Transactions on Image Processing, vol. 30, pp. 2029-2044, 2021.
- [4] R. Qian, R. T. Tan, W. Yang, J. Su, and J. Liu, "Attentive generative adversarial network for raindrop removal from a single image," in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2482-2491, 2018.
- [5] H. Lin, C. Jing, Y. Huang, and X. Ding, "A 2 Net: Adjacent aggregation networks for image raindrop removal," IEEE Access, vol. 8, pp. 60769-60779, 2020.
- [6] D. M. Nguyen and S.-W. Lee, "UnfairGAN: An Enhanced Generative Adversarial Network for Raindrop Removal from A Single Image," arXiv preprint arXiv:05523, 2021.
- [7] Z. Hao, S. You, Y. Li, K. Li, and F. Lu, "Learning from synthetic photorealistic raindrop for single image raindrop removal," in Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops, pp. 0-0, 2019.
- [8] H. Porav, T. Bruls, and P. Newman, "I can see clearly now: Image restoration via de-raining," in International Conference on Robotics and Automation (ICRA), pp. 7087-7093, 2019.
- [9] Q. Guo et al., "Efficientderain: Learning pixel-wise dilation filtering for high-efficiency single-image deraining," arXiv preprint arXiv:09238, 2020.
- [10] H. Zhang, V. Sindagi, and V. M. Patel, "Image de-raining using a conditional generative adversarial network," IEEE transactions on circuits systems for video technology vol. 30, no. 11, pp. 3943-3956, 2019.
- [11] M. Roser, J. Kurz, and A. Geiger, "Realistic modeling of water droplets for monocular adherent raindrop recognition using bezier curves," in Asian conference on computer vision, pp. 235-244, 2010.
- [12] D. M. Nguyen and S.-W. Lee, "M2GAN: A Multi-Stage Self-Attention Network for Image Rain Removal on Autonomous Vehicles," arXiv preprint arXiv:06164, 2021.
- [13] H. Zhang and V. M. Patel, "Density-aware single image de-raining using a multi-stream dense network," in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 695-704, 2018.
- [14] X. Fu, J. Huang, X. Ding, Y. Liao, and J. Paisley, "Clearing the skies: A deep network architecture for single-image rain removal," IEEE Transactions on Image Processing, vol. 26, no. 6, pp. 2944-2956, 2017.
- [15] Y. Luo, Y. Xu, and H. Ji, "Removing rain from a single image via discriminative sparse coding," in Proceedings of the IEEE international conference on computer vision, pp. 3397-3405, 2015.
- [16] T. Wang, X. Yang, K. Xu, S. Chen, Q. Zhang, and R. W. Lau, "Spatial attentive single-image deraining with a high quality real rain dataset," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 12270-12279, 2019.
- [17] Y. Wei et al., "Deraincyclegan: Rain attentive cyclegan for single image deraining and rainmaking," IEEE Transactions on Image Processing, vol. 30, pp. 4788-4801, 2021.
- [18] X. Li, J. Wu, Z. Lin, H. Liu, and H. Zha, "Recurrent squeeze-and-excitation context aggregation net for single image deraining," in Proceedings of the European conference on computer vision (ECCV), pp. 254-269, 2018.
- [19] Z. Fan, H. Wu, X. Fu, Y. Hunag, and X. Ding, "Residual-guide feature fusion network for single image deraining," arXiv preprint arXiv:07493, 2018.
- [20] W. Wei, D. Meng, Q. Zhao, Z. Xu, and Y. Wu, "Semi-supervised transfer learning for image rain removal," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 3877-3886, 2019.
- [21] S. Gu, D. Meng, W. Zuo, and L. Zhang, "Joint convolutional analysis and synthesis sparse representation for single image layer separation," in Proceedings of the IEEE International Conference on Computer Vision, pp. 1708-1716, 2017.
- [22] J. Xiao, M. Zhou, X. Fu, A. Liu, and Z.-J. Zha, "Improving De-Raining Generalization via Neural Reorganization," in Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 4987-4996, 2021.
- [23] H. Wang, Q. Xie, Q. Zhao, and D. Meng, "A model-driven deep neural network for single image rain removal," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3103-3112, 2020.
- [24] Y. Liu, Z. Yue, J. Pan, and Z. Su, "Unpaired Learning for Deep Image Deraining With Rain Direction Regularizer," in Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 4753-4761, 2021.
- [25] R. Yasarla and V. M. Patel, "Uncertainty guided multi-scale residual learning-using a cycle spinning cnn for single image de-raining," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8405-8414, 2019.
- [26] Y. Li, R. T. Tan, X. Guo, J. Lu, and M. S. Brown, "Rain streak removal using layer priors," in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2736-2744, 2016.
- [27] W. Yang, R. T. Tan, J. Feng, J. Liu, Z. Guo, and S. Yan, "Deep joint rain detection and removal from a single image," in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 1357-1366, 2017.
- [28] X. Fu, J. Huang, D. Zeng, Y. Huang, X. Ding, and J. Paisley, "Removing rain from single images via a deep detail network," in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 3855-3863, 2017.
- [29] D. Ren, W. Zuo, Q. Hu, P. Zhu, and D. Meng, "Progressive image deraining networks: A better and simpler baseline," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3937-3946, 2019.
- [30] X. Liu, M. Suganuma, Z. Sun, and T. Okatani, "Dual residual networks leveraging the potential of paired operations for image restoration," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7007-7016, 2019.
- [31] K. Zhang, D. Li, W. Luo, and W. Ren, "Dual attention-in-attention model for joint rain streak and raindrop removal," IEEE Transactions on Image Processing, vol. 30, pp. 7608-7619, 2021.
- [32] S. Li et al., "A comprehensive benchmark analysis of single image deraining: Current challenges and future perspectives," International Journal of Computer Vision, vol. 129, no. 4, pp. 1301-1322, 2021.
- [33] H. Wang, Q. Xie, Y. Wu, Q. Zhao, and D. Meng, "Single image rain streaks removal: a review and an exploration," International Journal of Machine Learning Cybernetics, vol. 11, no. 4, pp. 853-872, 2020.
- [34] J. Xu, W. Zhao, P. Liu, and X. Tang, "Removing rain and snow in a single image using guided filter," in IEEE International Conference on Computer Science and Automation Engineering (CSAE), vol. 2, pp. 304-307, 2012.
- [35] J.H. Kim, C. Lee, J.Y. Sim, and C.S. Kim, "Single-image deraining using an adaptive nonlocal means filter," in IEEE international conference on image processing, pp. 914-917, 2013.
- [36] K. He, J. Sun, and X. Tang, "Guided image filtering," IEEE transactions on pattern analysis machine intelligence, vol. 35, no. 6, pp. 1397-1409, 2012.
- [37] R. Li, R. T. Tan, and L.-F. Cheong, "All in one bad weather removal using architectural search," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 3175-3185, 2020.