How effective is a convolutional neural network that combines a deblurring and denoising subnetwork at recovering the latent image from an image that is affected by both blur and noise?

AP Research

Word Count: 4631

May 18, 2021

NOMENCLATURE

convolutional neural network	A deep neural network where convolution is used as the propagation function within neurons.
module	A distinct subset of layers within a neural network.
convolution	The process where a kernel is applied to an image by iteratively by sliding it left to right, top to bottom.
kernel	A small matrix used as an image filter which is applied through convolution.
image prior	Existing knowledge about images used to improve output accuracy.
iterative	Code that is repeated continuously until an exit condition is fulfilled.
parametric	Has a fixed number of independent variables.

***Abstract* — State-of-the-art image recognition neural networks are currently trained on high quality, clean images and as such, are significantly impacted by the presence of noise and blur. As a result, there has been significant research into the developing neural networks that restore images affected by noise or blur. This paper investigates the effectiveness of combining a denoising and deblurring network architecture at restoring images. To accomplish this, a model in which a denoising module feeds into a deblurring one was proposed and then implemented in Python. It was then tested on a dataset of noisy, blurred images, the results of which were compared to those from a denoising neural network and a deblurring one. Analyzing the output from the three network architectures, I found that the deblurring neural network struggled immensely with restoring images while slight differences were observed between the proposed combined model and the denoiser, specifically in the details and edges of the objects. Additionally, the proposed model struggled at higher noise levels and with darker regions. I concluded that my proposed model slightly outperformed the state-of-the-art denoiser in recovering the sharp image from one affected by a combination of noise and blur.**

I. INTRODUCTION

Image recognition software is poised to become ubiquitous over the next decade as it is integrated into numerous fields, from speeding up medical diagnoses [1] to guiding autonomous vehicles to performing food inspections [2]. Most of the research into image recognition neural networks, however, relies upon benchmark datasets for their training. These consist of relatively high-quality, sterile images. [3] In contrast, the overwhelming majority of real-world images contain various types of distortion, which can lead to significant drops in the accuracy of image recognition software, with blur and noise being the most problematic. [4] As such, the removal of blur and noise remain active problems within the field of image processing and there has been significant research dedicated to developing both deblurring and denoising algorithms. This research has primarily focused on algorithms that simply tackle one type of image reconstruction. However, real-world images are unlikely to only be affected by only a singular type of distortion, rather they are likely to contain a combination of blur and noise degradation; while there exist state-of-the-art methods to remove a singular type of noise, there remains a lack of research into the effects of merging two distinct algorithms into one on their effectiveness at restoring distorted images.

Noise is an unwanted signal that affects the original image and is typically modeled as either a random additive or multiplicative component on top of the latent image. [3] As such, the two forms of noise are respectively modeled by

$$\hat{s} = \eta + s, \tag{1}$$
$$\hat{s} = \eta s, \tag{2}$$

where $\hat{s}$ is the noisy output image, η is a random variable described by a probability function, and s is the latent sharp image. Noise is typically added to raw image data as it passes through a camera's image signal processing (ISP) pipeline. This typically consists of automatic white balancing, demosaicing, denoising, tone mapping, gamma correction, and data compression [5], in addition to the sensor noise added as the raw data is collected [6]. Sensor noise is caused by photon arrival statistics and imprecision in the circuitry, which are known as shot and read noise respectively, [6] and is described by a Poisson-Gaussian model, since shot noise follows a Poisson distribution while read noise is modeled by Gaussian curve [5].

Blur is a degradation that occurs when a camera's sensor pixels average light from different regions in a scene [7]. It is modeled as a convolutional function where a blur kernel has been applied to the entirety or a region of a latent sharp image and, consequently, is described as

$$\hat{s} = k * s, \tag{3}$$

where $\hat{s}$ is the vectorized output blurred image, s is the latent image, and k represents a large, sparse matrix that contains local blur kernels which act upon s. [8] It occurs when either an object in frame or the camera itself moves as an image is being taken and the sensor is exposed.

Real-world image restoration requires the application of multiple techniques to remove different forms of distortion. This research aims to address the lack of research into using different image restoration neural networks in combination by providing information and data on the efficacy of combining deblurring and denoising network architectures when implementing image restoration methods and identifying any possible negative effects that arise from it. This will support the application of image restoration neural networks in real-life scenarios, their use in conjunction with image recognition networks, and provide insight to future researchers on how combining the architecture of image restoration networks affects their combined performance, creating the possibility of investigating neural networks that can remove multiple types of distortion or performing two distinct actions.

II. RELATED WORK

In the last few years, there has been a significant amount of effort dedicated to research into deep learning and image recognition problems which has led to numerous breakthroughs and significant advances in both. However, most of this research was conducted in optimum conditions, with the training sets and benchmarks used containing relatively small amounts of distortion. However, noise is often unavoidable in practical applications; as such, there has also been research conducted into the limitations of our current image recognition models and how distortions affect image recognition accuracy. In [4], Dodge and Karam examined how different forms of image degradation, specifically noise, blur, contrast, and compression artifacts, impacted the accuracy of various state-of-the-art convolutional neural networks in image recognition tasks. Comparing the top-1 accuracy of the tested networks, they found that the presence of even small amounts of noise or blur in an image has a significant negative impact on the classification accuracy of an identification neural network. As such, numerous techniques have been proposed and developed to minimize the presence of these distortions in images.

A. Noise Reduction Methods

Deep learning-based models remove noise from an image by creating a denoiser neural network that can generate a clean output from a noisy input image. This is either accomplished directly, by attempting to restore a clean output image, or indirectly, by first estimating the residual image or noise added and then using it to determine the clean image prior. Convolutional neural networks were first used in 2009 by Jain and Seung [9] when they trained a five-layer convolutional neural network to denoise grayscale images. A notable improvement to this model was made by Zhang et al. [10] with their proposed denoising convolutional neural network (DnCNN) which incorporated residual learning and batch normalization into its architecture. This resulted in an image denoiser that was faster to train and outperformed traditional non-CNN methods at blind Gaussian denoising [10]. DnCNN is limited when it comes to denoising real images due to the synthetic Gaussian noisy images used to train it deviating from the noise present in real images [11], and its reliance on blindly predicting the amount of noise present without taking in the underlying textures of the image. This problem was solved by Guo et al. [11] with their convolutional blind denoising network (CBDNet) which was an amalgamation of a noise estimation subnetwork which fed into a non-blind deblurring module that was trained on a more realistic noise model. However, CBDNet required manual intervention to improve results at high noise levels. This problem was solved by Anwar and Barnes [12] when they proposed an end-to-end neural network architecture that could handle denoising real images without requiring separate subnetworks or human intervention.

B. Deblurring Techniques

Blind image deblurring was first tackled in the seminal paper [13] by Fergus et al. which estimated the blur kernel of an image in a coarse-to-fine fashion and then used it to recover the latent unblurred image. This method was improved upon in numerous studies that designed natural image priors that were capable of supressing visual artifacts which improved the quality of the output. Some examples include a hyper-Laplacian prior [14], a l_0-regularized intensity and gradient prior for text developed by Pan el al. [15], and a prior that modeled point light sources and light streaks for deblurring low-light images [16]. However, as can be surmised from the specificity of the given examples, this method of deblurring requires highly tailored priors which limit their ability to be generalized for other situations. This led to the development of methods involving iterative priors that successively improve their estimation of the blur kernel using a parametric prior model. Examples of these types of models include the network architecture presented in [17] which uses the difference between the strengths of cross-scale patch recurrence in sharp and blurry images to determine the blur kernel and that shown in [18] which used a heavily tailed gradient prior to iteratively produce the latent image. While more accurate and generalizable, iterative algorithms have orders of magnitude longer training times than those mentioned earlier due to their repetitive nature.

Schuler et al. first proposed the application of convolutional neural networks to the task of blind image deblurring in [19]. Their proposed network's architecture closely mimics the iterative deblurring methods mentioned above as it iteratively performed the blur kernel estimation and latent image production in a coarse-to-fine manner; it simply replaced the optimization algorithms found in [17], [18] with convolutional neural networks. This resulted in a similar poor training time as the conventional iterative approaches due to the same recursiveness existing in the network architecture. This was solved by Sun et al. [20] when they proposed a sequential approach to deblurring. It determined the likelihood for each of the seventy-three pre-determined candidate blur kernels to apply to a local patch and used that information to obtain a varying blur kernel which was applied with a conventional optimization algorithm to produce the latent image. However, this method requires the accurate estimation of an image's blur kernel which involves several problems. Nah et al. [8] addressed these issues in their proposed model, DeepDeblur, which introduced a multi-scale architecture to progressively determine the sharp image without the requiring an accurate estimation of the blur kernel.

C. Previous Models

This problem has been tackled in the past using traditional methods in which a linear system of equations that model the problem is setup and then solved computationally. Vogel and Oman tackle restoration of a noisy, blurred image as a total-variation problem. However, total-variation is a nonlinear reconstruction method [21] and as such, must be linearized by formulating it into a minimization problem whose associated Euler-Lagrange equation can then be solved. This is accomplished by applying half-quadratic regularization with this process comprising their reconstruction algorithm. [21] In contrast, Lagendijk *et al.* approaches blur reconstruction as a maximum likelihood problem which requires the point-spread function to be determined, something that is impacted by the introduction of noise. As such, they incorporate an expectation-maximization algorithm to estimate the point-spread function and consequently, the parameters for the maximum likelihood problem, which is then solved. [22]

However, as was mentioned earlier, both papers approach the problem traditionally and, as such, there remains a lack of research into the efficacy of using neural networks to solve this problem that combine two forms of image distortion.

D. Gap in the Existing Research

While this problem has been examined traditionally and neural networks have shown state-of-the-art performance in removing their respective types of image distortions to recover the latent sharp image, there is a lack of research into how they interact with one another when used in conjunction to recover an image that is affected by both blur and noise. While image restoration researchers often look for and adopt methods and architectures that have been successfully employed in other areas of image processing, there remains a gap in how these state-of-the-art neural networks might interact with one another after they have been trained. This has led me to develop the following research question:

How effective is a convolutional neural network that combines a deblurring and denoising subnetwork at recovering the latent image from an image that is affected by both blur and noise?

Considering the overlapping forms of image distortions that the proposed model is explicitly designed to tackle, I expect it will prove more effective than utilizing only a denoising or deblurring neural network to restore an image. Furthermore, I predict that the magnitude of the distortion left after the restoration process to be significantly less than the sum of that after each individual neural network is run individually.

III. Proposed Model

My proposed model is composed of two main modules, the first consisting of the multi-scale deblurring network architecture from [8] with second being based upon the RIDNet denoiser [12]. Since blur is introduced while an image is being taken and noise is introduced by the ISP post-processing pipeline used to convert the raw data captured by the camera into a digital file format, the modules must be placed in the reverse order of these transformations when attempting to undo them. Furthermore, due to the data loss from to the introduction of noise, restoring noisy images results in a loss of detail and smoother edges when compared to the latent image, something evident in [12, Fig. 4]. Thus, I decided that the denoising component of the neural network would be before the deblurring section.

The process of blurring an image results in a loss of its features [23]. This would have an adversarial impact on the performance of RIDNet due to it rendering the residual-on-residual component useless. This was measured in [12] when an ablation study was conducted and the removal of the feature attention components had a negative impact on the network's capabilities. As such, a modified version of the architecture outlined in [12] where the feature attention capabilities were removed was employed as the denoising module for the proposed model.

This resulted in a module composed of a feature extraction, feature learning residual, and reconstruction components. The input image is first processed by the feature extraction component which consists of a single convolution layer. The output is then passed to the feature learning residual component, composed of four consecutive enhancement attention modules EAMs and a long skip connection. Each EAM consists of three blocks with the first covering the full receptive field of features, the second learning on those features, and the third compressing the features. The first task is accomplished with a merge-and-run unit where the input features are passed through two dilated convolutional filters, the results of which are then combined and convoluted once more. Then, the features are learned within a final residual block and compression is achieved by three convolutional layers, with the last applying a 1×1 convolution kernel. Finally, the output features are given to the reconstruction module, which is composed of one convolution layer, and outputs the denoised image.

This denoised image is then used as the input for the deblurring module, which employs the multi-scale neural network architecture of [8] to deblur it. The architecture employs a Gaussian pyramid structure with consecutive patch sizes of 64×64, 128×128, and 256×256 pixels. A modified version of a residual network at each scale level was used as it was found that the removal of the rectified linear unit after the skip connection from the typical residual blocks decreases the network's training time without negatively impacting performance [8]. The first convolution layer transforms the 64×64 image into sixty-four separate feature maps. Then, each of the maps is consecutively passed through nineteen residual blocks with a final convolution layer transforming each feature map back to its input dimensions. Each residual block consists of two convolutional layers and a skip connection from its input to the output of the second layer. As a result, each scale level has a total of forty convolutional layers with the overall network containing one hundred twenty layers. All the convolutional layers preserve the initial resolution and have no padding. Finally, at the end of each stage, the latent sharp image is generated and upscaled using transpose convolution [24] to be used as the input for the first convolutional layer of the finer scale. The structure of the finer scale level networks is identical to that of the first, with the exception that the first convolution layer also takes the sharpened feature from the previous level in a concatenated form. The same number of feature maps are created as the coarsest scale level with the same convolutional filter size. The last scale level skips the final transpose convolution step and simply restores the latent sharp image in the original resolution.

IV. Methodology

To evaluate the efficacy of the proposed model in image restoration, a post-test experimental study was conducted. The model was implemented, trained, and then tested, with its output images being compared quantitatively and qualitatively to those of the state-of-the-art neural networks for both denoising and deblurring. This methodology was chosen as it allows for the correlation between a independent and dependent variable to be tested in addition to being the predominant method employed in the field, used in both the papers from which the proposed model is derived ([8], [12]) and in those that tested a proposed solution to a similar problem ([21], [22]).

The proposed model was implemented in Python 3.7 using the PyTorch framework and all three models were tested on an AMD Ryzen 3600 CPU and a Nvidia GeForce RTX 3070 GPU. A filter size of 3×3 pixels was used to train the convolutional layers of the denoising modules (unless otherwise specified) while a 5×5 convolutional filter was employed in the training

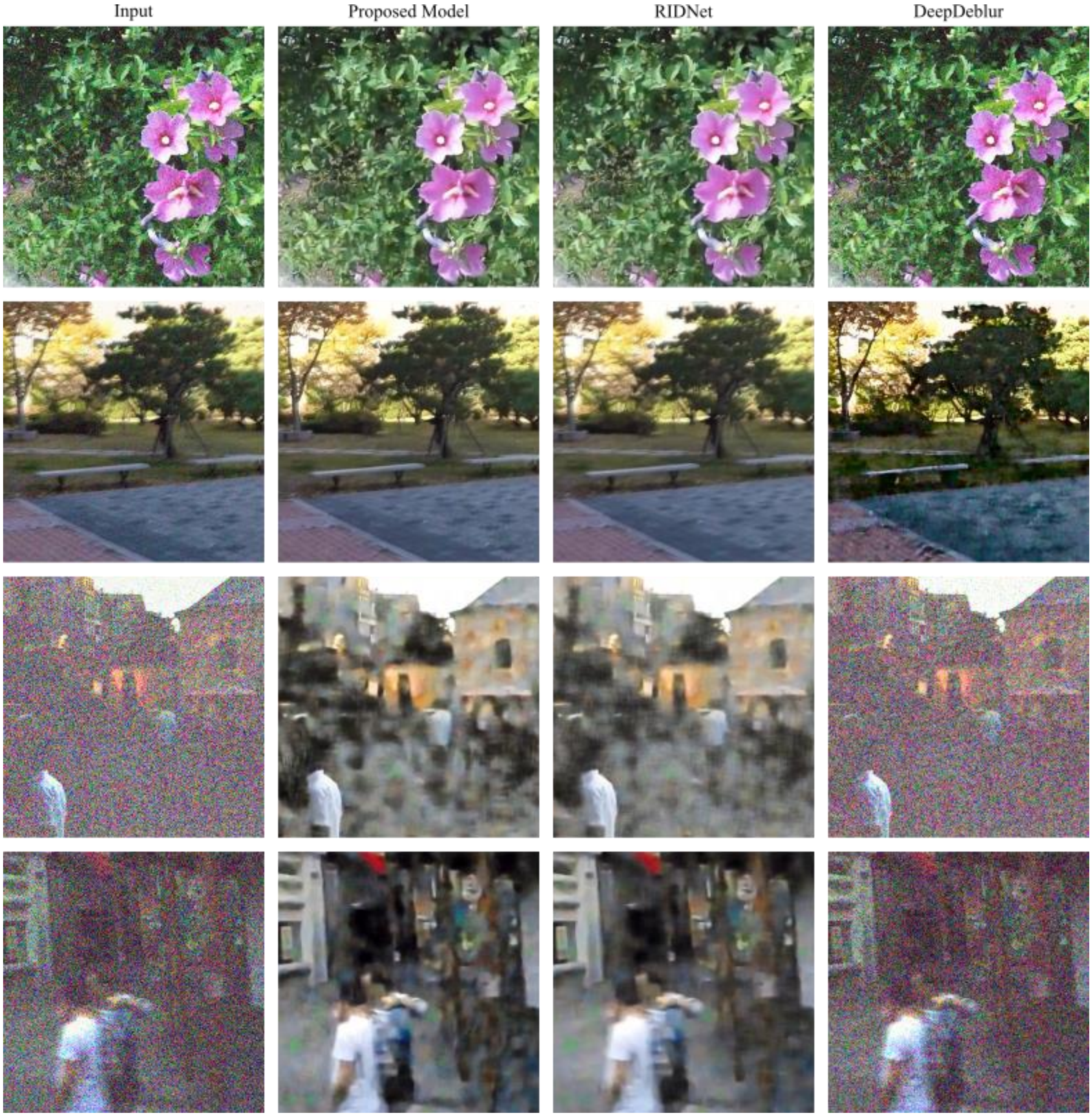

Figure 1. Representative samples of the results on the GOPRO dataset with synthetic signal noise added at different blur and noise levels. The first row shows results on a low noise, low blur image, the second on a low noise, high blur image, the third on a high noise, low blur image, and the final on a high noise, high blur image.

of the deblurring module, as was suggested by the respective papers from which the models were derived. A batch size of 8 images was chosen over the suggested size of 16 images due to memory constraints on the GPU during the training process.

To form the testing and training datasets of blurred and noisy images, signal noise was introduced to blurry images, using the process outlined in [22]. This process was performed on 1112 images from the GOPRO dataset [8], which contains pairs of blurred and sharp images, using the OpenCV library to form the testing dataset and approximately an additional 4000 images from the GOPRO dataset to form the training data. The GOPRO dataset was chosen for how it more closely mimics real-life blur with its synthetic blurry images when compared to those where a blur kernel was simply applied to a sharp image [8]. The image processing pipeline outlined in [6] and its associated code was chosen to synthetically add noise over a simple Poisson-Gaussian noise model due to it introducing a more realistic level of noise [6]. The program used simulates the functions performed within a conventional camera's image processing pipeline to convert raw sensor data into a final image. This was accomplished by adding shot and read noise, demosaicing the raw image, applying digital gain and a white balance algorithm, and, lastly, performing colour correction, gamma compression, and tone mapping. The model first recovers the raw data from the input image by running it through the mock processing pipeline in reverse and then through the pipeline again to increase or decrease the level of noise.

The peak signal-to-noise ratio (PSNR) index and structural similarity index measure (SSIM) were used to quantitatively evaluate the output images from the proposed model and the state-of-the-art denoising and deblurring algorithms. PSNR was chosen due to its widespread usage in the field [25] and the incorporation of mean squared error in its formula which allows it to measure the absolute error between the reference and restored images [26]. SSIM was chosen because of how it measures the relative similarity of two images by mimicking the visual quality perception of humans, [26] and as such, provides a more accurate measurement of how a human would perceive the difference between the restored image and the original.

While thorough, several limitations arise from the method employed in this study, however. Due to time constraints, other combinations of denoising and deblurring neural network architectures were unable to be trained and tested to allow for more comparisons to be made to the performance of my proposed model which would provide additional context on its performance. Furthermore, constructing an additional training and testing dataset composed of real-life noisy, blurred images in addition to the synthetic one that was employed in my study would improve the real-world applicability of the model presented and the accuracy of my data. However, I believe that this shortcoming will not affect the relationship between the efficacy of the different models, as they all have been trained and tested with identical datasets and thus, allow me to make conclusions upon the efficacy of my proposed model by comparing its performance relative to the other image restoration models.

V. RESULTS

In this experiment, my proposed model was tested on a synthetic noisy, blurred dataset based on the GOPRO dataset. Table 1 presents the average difference between the output images from DeepDeblur, RIDNet and my proposed model, and the original sharp images, as measured by the PSNR and SSIM values. The architectures of both DeepDeblur and RIDNet were explained in the literature review of this paper. Overall, DeepDeblur performed poorly, with its average PSNR and SSIM values well below that of both RIDNet and my proposed model. The slight difference between the performance of RIDNet and the proposed model further demonstrates the struggles of DeepDeblur as the incorporation of its architecture into an image restoration neural network only marginally improved its performance over that of just a denoiser on noisy, blurred images.

Comparing the results from the three networks qualitatively reinforces these findings and provides additional context for their performance. Figure 1 demonstrates how all the models were able to successfully denoise and deblur images

Figure 2. Representative samples of the results from the proposed model on dark regions of images containing various levels of noise and blur.

Metrics	RIDNet	DeepDeblur	Proposed Model
PSNR	21.573	14.479	21.706
SSIM	0.602	0.163	0.609

Table 1. Quantitative performance results on test images from the GOPRO dataset with synthetic signal noise added.

affected by a low level of blur and signal noise. The results from the proposed model and RIDNet are visually similar at all levels of noise and blur, with noticeable differences only appearing when closely examining the edges of object. This phenomenon is demonstrated by the man wearing a white shirt in the third row and the flowers in the top right corner of the first row. It also shows where DeepDeblur struggled to perform when compared to the other two models. In images affected by a high level of noise, it proved significantly less effective at restoring the sharp image, providing an explanation for its poor results in Table 1. This explains the marginal improvements the proposed model showed over RIDNet as it incorporates the deblurring network architecture of DeepDeblur which struggles to restore the distorted images output by the denoising module. Moreover, at higher noise levels, there is significant discolouring and regional distortions in the output image from RIDNet; thus, it can be surmised that due to the denoising module of my proposed model employing the architecture of RIDNet, the deblurring module was fed distorted images, further decreasing its effectiveness. Furthermore, dark regions of an image appear to pose a significant challenge to the proposed model as they saw more prominent false colours and unwanted distortions being introduced. Figure 2 shows that no matter the level of noise or blur that affects an input image, when comparing the darker regions of the restored image from the proposed model to the brighter areas, the darker regions will be more heavily affected by unwanted discolourations and distortions than the lighter areas.

VI. Analysis

This data demonstrates the importance of the architecture chosen when combining image restoration neural networks to tackle images affected by more than one type of distortion and tailoring a model's architecture for both the problem it is trying to solve and the nature of its input.

While RIDNet was able to denoise a sharp image affected by a high level of noise without introducing noticeable artifacts to an image [12, Fig. 8], this is not the case with my proposed model, especially in images with a high level of signal noise. I believe this is because the reconstruction layer of RIDNet, and consequently, the denoising module of my proposed model, relies upon an output layer of features from the neural network to reconstruct the latent sharp image which the presence of blur would diminish and distort. The difficulties that the proposed model experienced with dark regions supports this conclusion as it is a similar problem that results from the fact that low-light regions lack the features [16] that the deblurring module's coarse-to-fine architecture relies upon. Additionally, their low pixel values obscure the context required to denoise them [27]. Furthermore, the impact of the loss of information explains the unpredicted inability for my proposed model to restore details in an image. While the addition of blur and noise to an image both result in a loss of detail and information, blur convolutes an image, spreading information through a wider area which allows details to be reconstructed, while noise is an additive or multiplicative component, which, especially at higher levels, results in information lost as the total value of a pixel exceeds the 256 bits of information that can be stored.

Likewise, a problem in the architecture of the model I based my denoising module explains the minimal improvement in performance my proposed model demonstrated over that of RIDNet. The course-to-fine approach and multi-scale training employed by DeepDeblur explains its poor performance when working with a noisy or distorted input image. The additional distortion obscured the structural features that appear at lower scale levels that its architecture relies upon to initially restore blurred images and it confuses the distortions with the details of the latent image it tries to recover at higher scale levels. This confusion of unwanted noise with recoverable details is evident in the output of DeepDeblur in the first and second rows of Table 1 where it has made the original noise more prominent in the restored image. As was noted earlier, even at higher noise levels, the proposed model was able to reconstruct edges of objects unaffected by regional discoloration or distortions, implying that deblurring module can restore the output from the denoising module if blur is the only form of image distortion the input region is affected by.

VII. Conclusion

In this paper, I proposed a model to restore noisy, blurred images which, unlike previous attempts which approached the problem traditionally, combined two separate denoising and deblurring architectures to form a convolutional neural network that recovers the latent sharp image. Regarding my hypothesis, my findings support the first claim that combining a denoising and a deblurring module is more effective than employing either a noise or blur restoration neural network when attempting to restore a noisy, blurred image. However, the second half of my hypothesis proved to be false as the results from my proposed model only slightly outperformed those from just a denoiser. Additionally, it appears that the denoising architecture I employed in my model adds regional distortion and discolouring when working with images impacted by a high level of blur or noise.

This study into restoring noisy, blurred images has numerous implications for both the real world and further research into the topic. The proposed model can be employed as a preprocessor for real-life images being input into object recognition neural networks trained on relatively high-quality datasets. Additionally, it can be used by devices on the "internet of things", as they typically contain low resolution, low quality cameras with a large depth of field and as such, are prone to adding both noise and blur to a raw image. Finally, my proposed model could be used to restore images taken in low-light situations, with long exposure times, or by older camera models since each of these factors contribute to higher levels of noise and/or blur in an image.

The limitations of my method and findings from my data raise several possible avenues for further exploration in future research. Investigations into the use of deblurring and denoising neural networks in conjunction with one another should analyze other combinations that were not tested in this paper. A focus on architectures that are not reliant upon the features of an image to restore it may prove more effective than my proposed model and help prevent the unwanted distortions that appeared at higher noise levels. Furthermore, future research should explore other possible architectures that would minimize the introduction of unwanted distortion in the output image at high noise levels or in dark regions. For example, the development of an end-to-end neural network architecture, which eliminates the need for compartmentalized modules, should be attempted as it would decrease the complexity and training time of such a multipurpose neural network and could improve on the performance of my proposed model at high noise levels in a similar fashion to how the end-to-end architecture of RIDNet improved on older denoising neural networks. Finally, while this paper examines image restoration with regards to images affected by a combination of noise and blur, there remains a lack of research into the use of other forms of image restoration neural networks in conjunction with one another.

References

[1] 'IBM Watson Health, Merge launch new personalized imaging tools at RSNA', *Healthcare IT News*, Nov. 30, 2016. https://www.healthcareitnews.com/news/ibm-watson-health-merge-launch-new-personalized-imaging-tools-rsna (accessed Jan. 08, 2021).

[2] R. Ciora and C. Simion, ‘Industrial Applications of Image Processing’, *ACTA Universitatis Cibiniensis*, vol. 64, Dec. 2014, doi: 10.2478/aucts-2014-0004.
[3] M. Koziarski and B. Cyganek, ‘Image recognition with deep neural networks in presence of noise – Dealing with and taking advantage of distortions’, *Integrated Computer-Aided Engineering*, vol. 24, no. 4, pp. 337–349, Jan. 2017, doi: 10.3233/ICA-170551.
[4] S. Dodge and L. Karam, ‘Understanding how image quality affects deep neural networks’, in *2016 Eighth International Conference on Quality of Multimedia Experience (QoMEX)*, Jun. 2016, pp. 1–6, doi: 10.1109/QoMEX.2016.7498955.
[5] O. Bielova, R. Hänsch, A. Ley, and O. Hellwich, ‘A Digital Image Processing Pipeline for Modelling of Realistic Noise in Synthetic Images’, in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, Jun. 2019, pp. 2905–2914, doi: 10.1109/CVPRW.2019.00350.
[6] T. Brooks, B. Mildenhall, T. Xue, J. Chen, D. Sharlet, and J. T. Barron, ‘Unprocessing Images for Learned Raw Denoising’, *arXiv:1811.11127 [cs]*, Nov. 2018, Accessed: Jan. 21, 2021. [Online]. Available: http://arxiv.org/abs/1811.11127.
[7] I. Vasiljevic, A. Chakrabarti, and G. Shakhnarovich, ‘Examining the Impact of Blur on Recognition by Convolutional Networks’, *arXiv:1611.05760 [cs]*, Nov. 2016, Accessed: Apr. 13, 2021. [Online]. Available: http://arxiv.org/abs/1611.05760.
[8] S. Nah, T. H. Kim, and K. M. Lee, ‘Deep Multi-scale Convolutional Neural Network for Dynamic Scene Deblurring’, *arXiv:1612.02177 [cs]*, May 2018, Accessed: Dec. 07, 2020. [Online]. Available: http://arxiv.org/abs/1612.02177.
[9] V. Jain and H. Seung, ‘Natural Image Denoising with Convolutional Networks.’, Jan. 2008, pp. 769–776.
[10] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, ‘Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising’, *IEEE Transactions on Image Processing*, vol. 26, no. 7, pp. 3142–3155, Jul. 2017, doi: 10.1109/TIP.2017.2662206.
[11] S. Guo, Z. Yan, K. Zhang, W. Zuo, and L. Zhang, ‘Toward Convolutional Blind Denoising of Real Photographs’, *arXiv:1807.04686 [cs]*, Jul. 2018, Accessed: Dec. 03, 2020. [Online]. Available: http://arxiv.org/abs/1807.04686.
[12] S. Anwar and N. Barnes, ‘Real Image Denoising with Feature Attention’, *arXiv:1904.07396 [cs]*, Mar. 2020, Accessed: Nov. 05, 2020. [Online]. Available: http://arxiv.org/abs/1904.07396.
[13] R. Fergus, B. Singh, A. Hertzmann, S. Roweis, and W. Freeman, ‘Removing camera shake from a single photograph’, *ACM Trans. Graph.*, vol. 25, pp. 787–794, Jul. 2006, doi: 10.1145/1179352.1141956.
[14] D. Krishnan and R. Fergus, ‘Fast Image Deconvolution using Hyper-Laplacian Priors’, *Advances in Neural Information Processing Systems*, vol. 22, pp. 1033–1041, 2009.
[15] J. Pan, Z. Hu, Z. Su, and M.-H. Yang, ‘Deblurring Text Images via L0-Regularized Intensity and Gradient Prior’, in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2014, pp. 2901–2908, doi: 10.1109/CVPR.2014.371.
[16] Z. Hu, S. Cho, J. Wang, and M. Yang, ‘Deblurring Low-Light Images with Light Streaks’, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 10, pp. 2329–2341, Oct. 2018, doi: 10.1109/TPAMI.2017.2768365.
[17] T. Michaeli and M. Irani, ‘Blind Deblurring Using Internal Patch Recurrence’, in *Computer Vision – ECCV 2014*, Cham, 2014, pp. 783–798, doi: 10.1007/978-3-319-10578-9_51.
[18] Q. Shan, J. Jia, and A. Agarwala, ‘High-quality Motion Deblurring from a Single Image’, *ACM Trans. Graph.*, vol. 27, Aug. 2008, doi: 10.1145/1360612.1360672.
[19] C. J. Schuler, M. Hirsch, S. Harmeling, and B. Schölkopf, ‘Learning to Deblur’, *arXiv:1406.7444 [cs]*, Jun. 2014, Accessed: Dec. 07, 2020. [Online]. Available: http://arxiv.org/abs/1406.7444.
[20] J. Sun, Wenfei Cao, Zongben Xu, and J. Ponce, ‘Learning a convolutional neural network for non-uniform motion blur removal’, in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015, pp. 769–777, doi: 10.1109/CVPR.2015.7298677.
[21] C. R. Vogel and M. E. Oman, ‘Fast, robust total variation-based reconstruction of noisy, blurred images’, *IEEE Transactions on Image Processing*, vol. 7, no. 6, pp. 813–824, Jun. 1998, doi: 10.1109/83.679423.
[22] R. L. Lagendijk, J. Biemond, and D. E. Boekee, ‘Identification and restoration of noisy blurred images using the expectation-maximization algorithm’, *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 38, no. 7, pp. 1180–1191, Jul. 1990, doi: 10.1109/29.57545.
[23] D. G. Lowe, ‘Distinctive Image Features from Scale-Invariant Keypoints’, *Int. J. Comput. Vision*, vol. 60, no. 2, pp. 91–110, Nov. 2004, doi: 10.1023/B:VISI.0000029664.99615.94.
[24] M. Mathieu, C. Couprie, and Y. LeCun, ‘Deep multi-scale video prediction beyond mean square error’, *arXiv:1511.05440 [cs, stat]*, Feb. 2016, Accessed: Mar. 03, 2021. [Online]. Available: http://arxiv.org/abs/1511.05440.
[25] D. R. I. M. Setiadi, ‘PSNR vs SSIM: imperceptibility quality assessment for image steganography’, *Multimedia Tools and Applications*, vol. 80, pp. 1–22, Mar. 2021, doi: 10.1007/s11042-020-10035-z.
[26] A. Horé and D. Ziou, ‘Image Quality Metrics: PSNR vs. SSIM’, in *2010 20th International Conference on Pattern Recognition*, Aug. 2010, pp. 2366–2369, doi: 10.1109/ICPR.2010.579.
[27] K. Xu, X. Yang, B. Yin, and R. W. H. Lau, ‘Learning to Restore Low-Light Images via Decomposition-and-Enhancement’, in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA, Jun. 2020, pp. 2278–2287, doi: 10.1109/CVPR42600.2020.00235.