A Study on the Generalization of Deep Learning Approaches to Low-Light Imaging Enhancements

ABSTRACT

Deep learning based image processing pipelines have recently started to become capable of surpassing the performance of traditional algorithm based approaches. The “See in the Dark” low-light image enhancement pipeline out of Intel Labs has shown promise of expanding the low-light performance of sensors by learning to enhance extreme low-light raw exposures, increasing exposure while suppressing sensor noise. The authors formed their own unique training and testing dataset for specific Sony and Fuji sensors. While they claim their models trained on these sensors are generalized to work with the raw data of other sensors, they provide little evidence. In our work we form our own training and testing dataset for the Canon 6D sensor in order to evaluate how well their trained model generalizes to our sensor data, as well as train the network with our sensor data and see how our trained model performs against theirs. We further seek to modify their original See-in-the-Dark (SID) network, both pruning and making it deeper, and evaluate performance among the modified networks for a standard testing dataset. With our original dataset obtained with an ISO of 200, we seek to form another training and testing dataset at ISO 12600 to push the camera near its low-light capturing capability to evaluate how well the trained network suppresses high ISO noise and amp-glow.

INTRODUCTION

Imaging in extreme low light conditions is challenging due to the need of balancing exposure time, sensor gain, and aperture size to acquire a properly exposed image without excessive noise. Balancing these three variables allows for acquisition of an image with a high signal-to-noise ratio (SNR) in any lighting situation, though with major tradeoffs. Increasing exposure length will allow the sensor to integrate more photons from a dark scene but will increase vulnerability to blurring due to motion in the scene or camera vibrations. Increasing sensor gain will amplify both signal and noise and will not produce a pleasing image if exposure time is too low for conditions. Increasing aperture size for the imaging system allows more photons to enter the image train, increasing SNR at the cost of added weight and size of a high aperture lens. There are certain imaging applications where these three variables are constrained, such as imaging at video frame rates (30 Hz, 60 Hz), or where the imaging system has tight size constraints.

In situations where an image is not or cannot be properly exposed it will have unacceptable noise levels due to low SNR. Traditional denoising image processing pipelines aim to apply global or local filtering of the image in order to suppress noise artifacts, typically at the expense of blurring high frequency details in the image. These denoising pipelines tend to fail when the SNR of an image is extremely low, where the color noise in the image is so strong that color balance cannot be accurately recovered.

RELATED WORK

Learning based techniques have recently been applied to solving the problem of extreme low-light imaging and have achieved performance that competes with or exceeds that of traditional methods [1]. The work of Chen Chen et al. [1] in their “Learning to See in the Dark” paper showed that the end-to-end training of a fully-convolutional neural network with image pairs of extremely underexposed images and properly exposed reference images achieved the ability to dramatically correct underexposed image exposure while suppressing sensor noise. Other researchers have proposed techniques for denoising, deblurring, and enhancement of low-light images [2, 3, 4]. These techniques fall short as they assume only a moderate amount of noise in the image. author’s formed training and testing datasets for both the Sony α7S Bayer and Fujifilm X-T2 X-Trans imaging sensors.

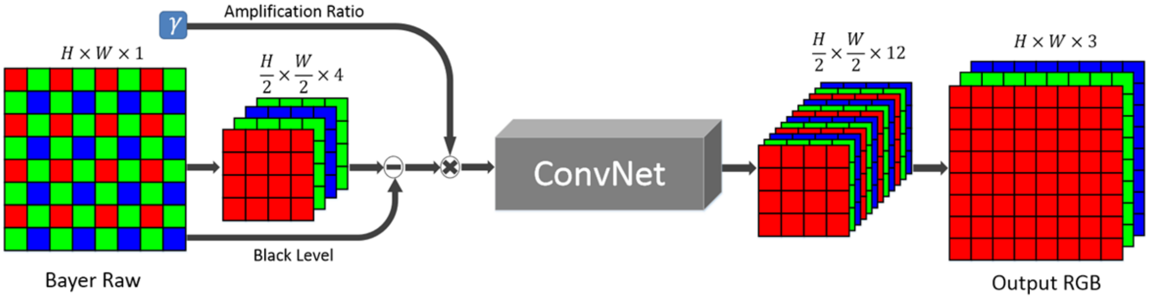

Figure 1: See-in-the-Dark (SID) pipeline.

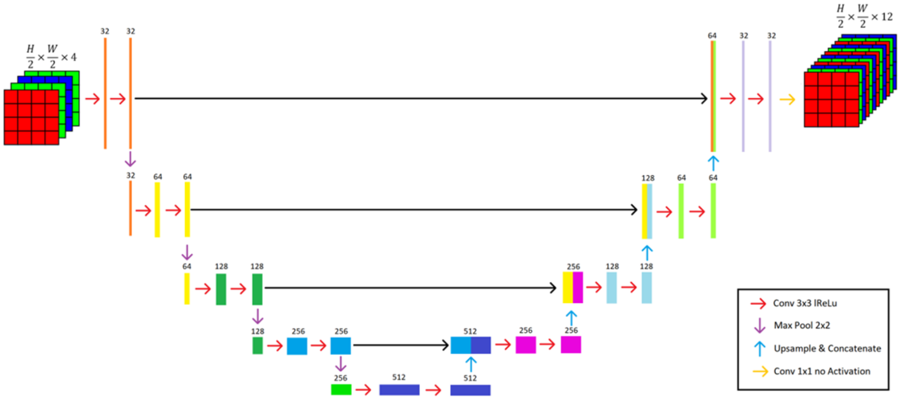

Figure 2: U-Net used in SID pipeline.

They trained their See-in-the-Dark (SID) pipeline with these training sets, and evaluated performance of the networks with corresponding testing datasets, using qualitative metrics as well as the quantitative metric of Peak-Signal-to-Noise-Ratio (PSNR). Since the Fujifilm sensor uses the proprietary X-Trans Bayer pattern, in this paper we focus on the results for the Sony α7S that uses the ubiquitous RGGB Bayer filter. The work of Chen Chen et al. [1] in their "Learning to See in the Dark" paper proposes a fully-convolutional image processing pipeline that is trained end-to-end to enhance low-light raw data while suppressing sensor noise, performing demosaicing and outputting a corrected image in RGB color space.

OVERVIEW

The pipeline shown in Figure 1 takes in raw RGGB Bayered sensor data and packs the data for each Bayer filter into a separate channel, thereby reducing the spatial resolution by half. The black level is then subtracted from the data, and the data is multiplied by the amplification ratio used to scale exposure. The data is then passed to a fully-convolutional network which outputs 12 channels of half the resolution of the original input. The 12 channels are then processed by a sub-pixel layer that forms the final three channel RGB image of full resolution.

The convolutional network used in the SID pipeline, shown in Figure 2, accepts packed Bayer data and is referred to as a U-Net. The data passes through two convolution layers and then a max-pool layer which downsamples the resolution by half. The next convolution layers use double the number of filters used previously. This convolution and downsampling scheme continues several times. The data is then upsampled using transposed convolution and concatenated with a skip connection from a previous convolution layer output from the downsampling side. This upsampled concatenated data then passes through two convolution layers of the same dimensions as at the same level in the downsampling path. This upsampling, concatenation with skip connections, and convolution occurs several times until the first convolution layer dimensions are returned. This HxWx32 data is then passed to a convolution layer with 12 - 1x1 filters, which produces the 12 channels that are then passed to the sub-pixel layer which forms the final RGB output image.

The See-in-the-Dark Sony α7S dataset consists of 232 unique training image pairs of underexposed images and properly exposed reference images (referred to as ground truth). The testing dataset consists of 51 testing image pairs, with standard exposure time gains between the dark and ground truth images of 100x, 250x, and 300x.

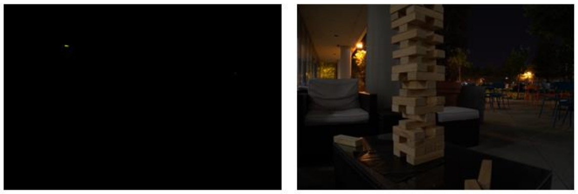

Figure 3: Example of underexposed image (Left) and properly exposed reference image (Right).

The SID network is trained using an L1 loss and the Adam optimizer. During training, the input to the network is raw data from an underexposed image, and the ground truth is the raw data of the corresponding image's properly exposed counterpart, as seen in Figure 3. The amplification ratio is set to the ratio of the ground truth and underexposed image exposure lengths. In each iteration of training a random 512x512 crop of each image is taken and randomly flipped and rotated to augment the training data. The training runs for 4000 epochs. The learning rate starts at 10^{-4} and at epoch 2000 goes down to 10^{-5}. The training seeks to optimize the convolution filter weights for each Conv layer in the U-Net, as well as the weights for transpose-convolution upsampling sections.

EXPERIMENT DETAILS

To verify the results of the paper we forked the repository and downloaded the Sony α7S dataset (25 GB). We ran training on a workstation with an Intel Xeon E5-2630V3 CPU, 64 GB of RAM, and a GTX 1080 ti GPU. Training through 4000 epochs on 232 image pairs from the author's dataset took one day. We then ran a testing script which passes images from the author's testing dataset to the network and forms the output results, which ran at a rate of one image result per second on our setup. An example image set is shown in Figure 4.

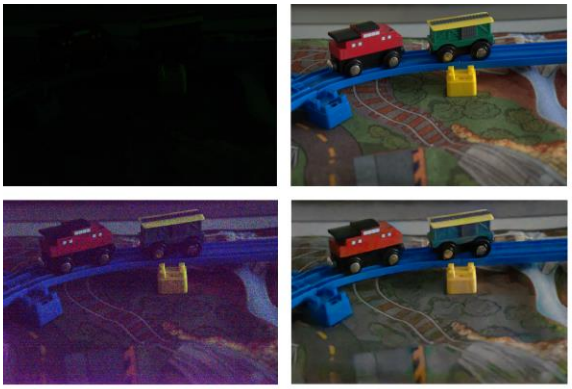

Figure 4: Output sample from author's testing set passed through author's trained model. Upper-left: underexposed input. Upper-right: ground truth. Lower-left: scaled input. Lower-right: network output.

We next investigated the generalization of the network model trained on Sony α7S data with raw bayer data from our Canon 6D camera. We obtained 21 unique ground truth images, each with three different short exposure input images, for a total of 63 input pairs. We passed these images through the authors trained Sony network and calculate the average PSNRs of the scaled image vs ground truth and network output vs ground truth .

We found some of the results remarkable, being able to bring out considerable detail from such a terribly dark image with low SNR. We also noticed that certain images have inaccurate color casts in low frequency areas of the network output image. The improvement over traditional methods is very clear in Figure 5, as the art on the pillow can now be seen.

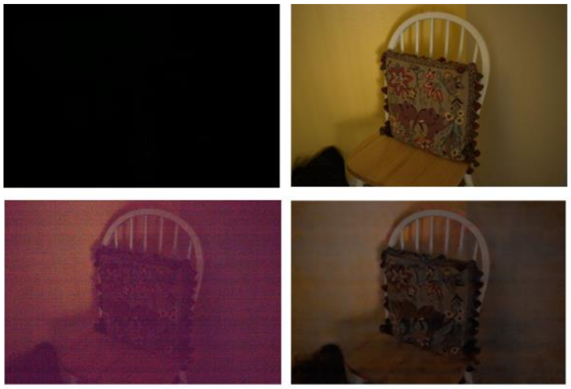

Figure 5: Output sample from our Canon 6D testing set passed through author's trained model. Upper-left: underexposed input. Upper-right: ground truth. Lower-left: scaled input. Lower-right: network output.

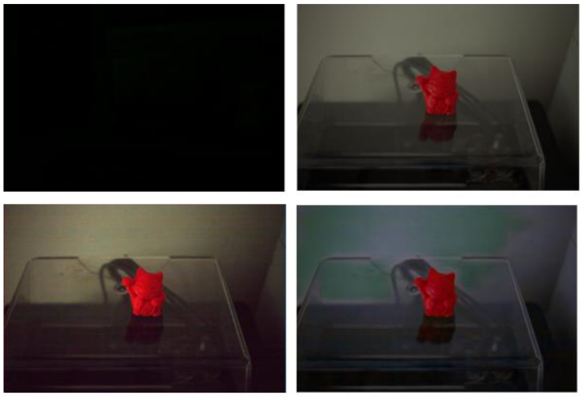

Figure 6: Output sample from our Canon 6D testing set passed through author's trained model, showing inaccurate green color cast in network output. Upper-left: underexposed input. Upper-right: ground truth. Lower-left: scaled input. Lower-right: network output.

In order to explore how well the trained model for the Sony α7S is generalized for other standard Bayer sensors, we sought out to create our own training set for a Canon 6D DSLR. The 6D is used with 40mm f2.8 and 200mm f2.8 lenses and constant ISO of 200 to capture various scenes in extreme low lighting, such as at night or in a dimly lit room. We kept the 6D on a rugged ball head with a stable carbon fiber tripod and used a remote app to control the camera and enabled mirror lockup when taking image pairs to eliminate any misalignment between consecutive images. The authors claimed to have used mirrorless cameras to avoid mechanical shaking from a mirror flipping when imaging. While our Canon 6D is not mirrorless, it was determined by calculating correlations between consecutive frames that our setup and imaging procedure did not introduce any misalignment between frames.

We formed a training set of 53 unique image pairs, and 12 images for testing. We discovered that the process of finding something unique to image, pointing the camera at the target, ensuring focus, determining image pair exposures, and acquiring the image pairs takes a considerable amount of time and effort. While the training set is smaller than that used by the authors, the training procedure uses randomized crops of training images in each iteration, thereby augmenting the training set. Training the SID network with the set of 53 image pairs for 4000 epochs took 7 hours.

Every 500 epochs the training script outputs several random crops of the training images passed through the network so that you can observe the network's performance over time.

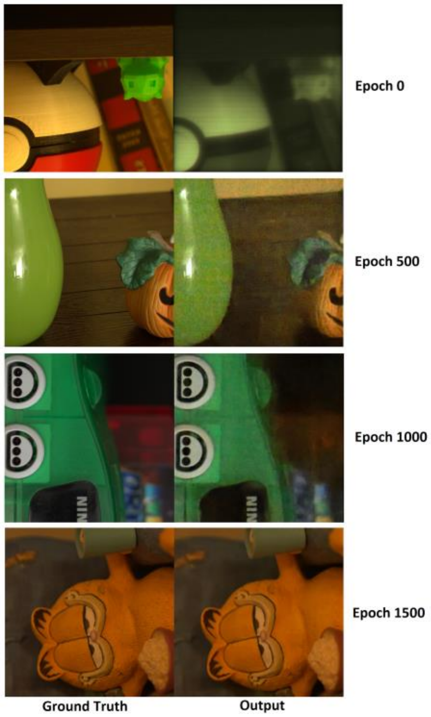

Figure 7: Sample of random crop of training images passed through the network at different levels of training with corresponding ground truth crop for comparison.

We next passed our Canon 6D testing dataset through the Canon 6D trained network, as well as through the network with the author's Sony model in order to see which had better results for testing data not used for training either model. The average PSNRs of the scaled image vs ground truth and network output vs ground truth are calculated using Equations 1 and 2 for the testing dataset for both the Sony and Canon 6D models.

$$MSE = \frac{\sum_{M,N}[I_1(m,n) - I_2(m,n)]^2}{M * N} \tag{1}$$

$$PSNR = 10\log_{10}\left(\frac{R^2}{MSE}\right) \tag{2}$$

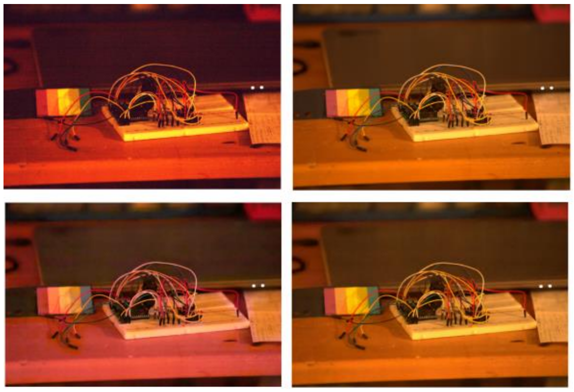

Figure 8: Output sample from our Canon 6D testing set passed through network with our Canon 6D model as well as author's Sony model. Upper-left: scaled input. Upper-right: ground truth. Lower-left: Sony network output. Lower-right: Canon 6D network output.

PERFORMANCE RESULTS

We found that the results when using Canon 6D model were substantially superior to those from using the author's Sony model. It can be seen in the images in Figure 8 above that while the result from using the Sony Model has inaccurate yellow-orange color casts throughout the image, the result from using the Canon 6D model has no color cast artifact, and also has an accurate color balance comparing against the ground truth.

The difference in performance between the Sony and Canon 6D model for the testing dataset we used can also be seen in the average PSNR calculations which are given in Table 1.

Table 1: PSNR results when using networks trained for various sensors.

Dataset	Amplified Image	Neural Net Image
Sony Sensor (Author's Figure)	Not Given	28.8
Sony Sensor (Our Figure)	14.37	28.61
Canon Sensor Final Testing (trained for Canon)	7.40	24.93
Canon Sensor Final Testing (trained for Sony)	7.40	10.85

Figure 10: Average PSNR table for quantitatively comparing network results vs ground truth, as well as scaled input image vs ground truth.

PROPOSED MODIFICATIONS

We next investigated how pruning of the network affects its performance, as well as if the network would perform better if it went deeper. Going in the direction of pruning the network to be smaller we first removed the final downsampling and first upsampling and concatenation step, the final level in the U-Net, to produce the network in Figure 11.

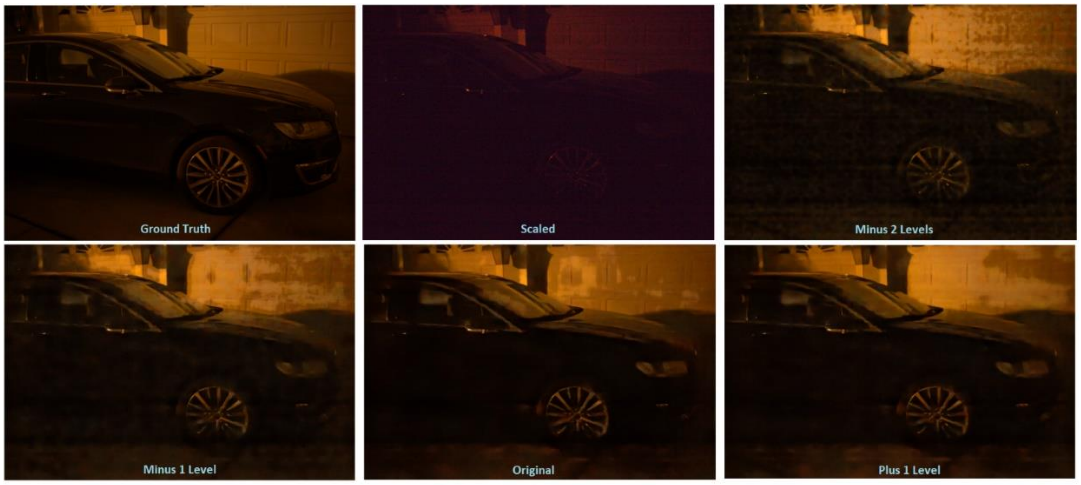

Figure 14: Comparison of ground truth, scaled, and network output images for original and modified SID networks.

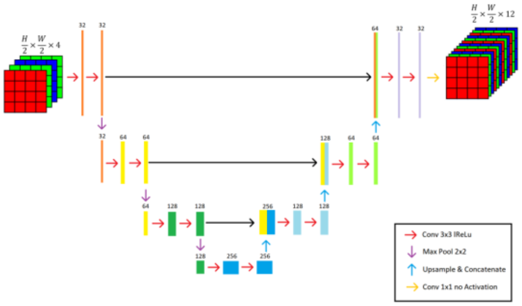

Figure 11: SID network with the removal of the final level of the original U-Net.

Going further we removed the final two downsampling and first two upsampling and concatenation steps, the final two levels in the U-Net, to produce the network in Figure 12.

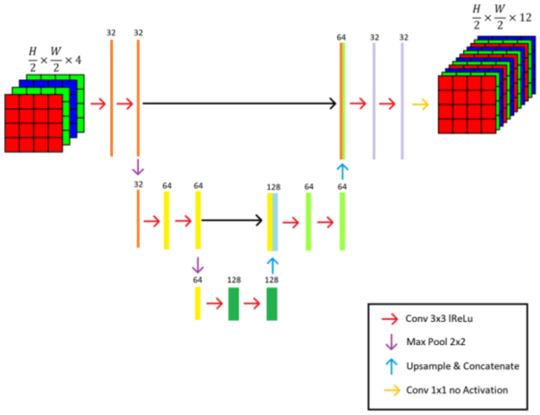

Figure 12: SID network with the removal of the final two levels of the original U-Net.

We finally made the original SID network go deeper by adding an additional downsampling section from the end of the bottom level as shown in Figure 13, reducing the resolution by half again and going from 512 to 1024 convolutional layer filters.

Unfortunately, as seen in Table 2 and Figure 14, our proposed modifications did not improve performance compared to the default network. Results were still better than using the traditional scaled approach, but more investigation is needed to see if the small improvement in PSNR is worth the high computational cost of using this network.

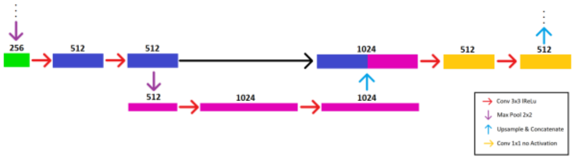

Figure 13: Deeper level appended to original SID network.

Table 2: PSNR results for our proposed modifications.

Network Type	PSNR
Original Network Minus 2 Bottom Levels	12.47
Original Network Minus 1 Bottom Level	12.36
Original Network	24.93
Original Network Plus 1 Deeper Level	12.40

CONCLUSION

In this work we have shown that despite claims of generality, the network in [1] has small performance losses when used with images taken by different sensors than what the network was trained with. We investigated the use of this network with a Canon 6D camera, and what improvements could be achieved by training the network to this specific sensor. We also explored depth modifications to the original network, we found that these modifications degraded performance. This could be due to improper training of the modified networks.

In the future we would like to continue building our dataset and doing a detailed explorations of possible network modifications.

ACKNOWLEDGEMENTS

We would like to thank Dr. Sedat Ozer for teaching the UCF Computer Vision course which helped give us the foundation to perform this exploratory study. We would also like to thank FLIR for giving us access to a workstation on which to train our network.

REFERENCES

[1] C. Chen, Q. Chen, J. Xu, V. Koltun – Learning to See in the Dark. arXiv:1805.01934

[2] J. Anaya and A. Barbu. RENOIR – A dataset for real lowlight image noise reduction. arXiv:1409.8230, 2014. 2

[3] H. C. Burger, C. J. Schuler, and S. Harmeling. Image denoising: Can plain neural networks compete with BM3D? In CVPR, 2012. 2

[4] Q. Chen and V. Koltun. Photographic image synthesis with cascaded refinement networks. In ICCV, 2017. 6

GITHUB

https://github.com/jconenna/Canon-6D-Datasets-For-Learning-to-See-in-the-Dark