Feature-Align Network with Knowledge Distillation for Efficient Denoising - Supplemental -

Lucas D. Young* Fitsum A. Reda*† Rakesh Ranjan
Jon Morton Jun Hu Yazhu Ling
Xiaoyu Xiang David Liu Vikas Chandra
Facebook Inc.

1. Noise Modeling

Image noise originates at the Bayer RAW domain. The observed signal of a pixel is gaussianly distributed about the noiseless intensity of the pixel. Noise originates from a variety of sources, but can be broken down into two categories - signal independent noise and signal dependent noise. The variance of the gaussian distribution of signal independent noise per-pixel does not vary depending on the intensity of a pixel. The variance of signal dependent noise per-pixel is proportional to that pixel's intensity. Read noise is a prominent example of signal independent noise and shot noise is a prominent example of signal dependent noise.

$$y \sim \mathcal{N}(\mu = x, \sigma^2 = ax + b) \tag{1}$$

Equation 1 models both signal independent noise and signal dependent noise as a single heteroscedastic gaussian, treating y as a variable whose variance is a function of the true signal x, where a and b represent the signal dependent and signal independent noise respectively [2, 11, 3]. This description of noise is only partially complete because it does not account for clipping of the signal. Due to clipping, the per-pixel distribution of a pixel is a censored gaussian distribution, and the expected variance at the low and high ends of the signal are decreased compared to Equation 1 [2]. To accurately model noise for a specific sensor, we choose realistic a and b parameters, sample the corresponding gaussian distribution for each pixel, add the noise to the ground truth image, and then clip the image.

To pick realistic as and bs for the target sensor, we conduct an empirical analysis. In a lab-controlled lighting environment, we capture samples of frames of a color checker in a variety of lighting conditions, holding exposure time constant. We scale the image's intensities to a 0-1 range in accord with the camera's white and black levels. After segmenting color boxes in the checker, we approximate the noise-free intensity of each pixel as the mean intensity of its corresponding color box. This allows us to observe the relationship between noise-free intensity and variance for each image (see Figure 2).

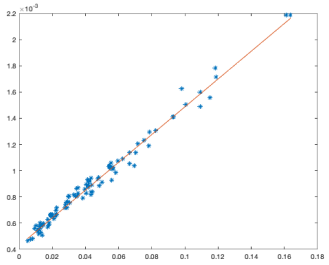
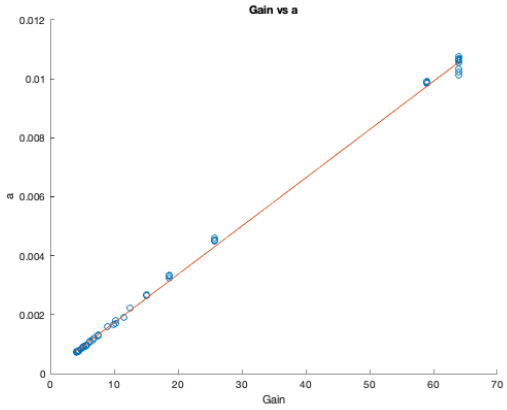


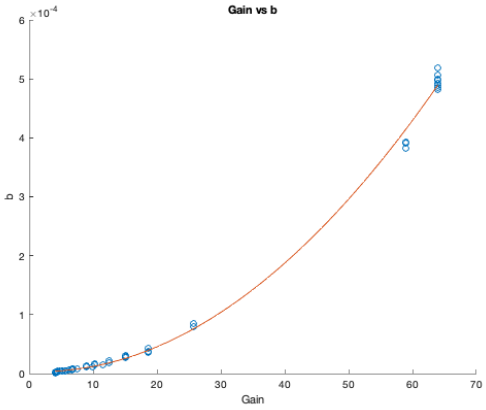
Figure 1. Example of intensity (x-axis) vs variance (y-axis) for a single image of a color checker.

We use the algorithm proposed by [2] to fit Equation 1 to this plot. Note that while the images used for this analysis are clipped, the algorithm takes this discrepancy into account when fitting the theoretical unclipped noise model. This yields the parameters a and b for the image, which can be used to generate artificial noise to be subsequently added to a ground truth image and clipped.

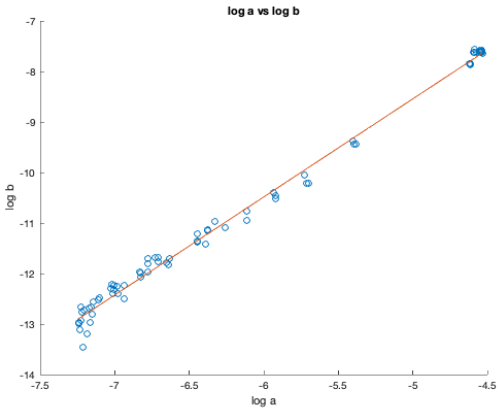
In training, we model the distribution of these a and b pairs in logarithmic space, randomly choose $\log a$ along a uniform distribution, and then pick $\log b$ based on a linear regression. The resulting regression is unique to each type of sensor. Our noise modeling was done for target camera's sensor, the Sony IMX258. Since the image's gain is known at inference, to predict the noise levels at inference we create regressions between a and gain and between b and gain. We find that a linear regression fits the relationship between a and gain and that a quadratic regression fits the relationship between b and gain. As demonstrated in [11], superior denoising performance can be achieved by having this information available to the algorithm.



(a) Linear regression between gain and a



(b) Quadratic regression between gain and b



(c) Regression between $\log a$ and $\log b$

Figure 2. The regressions in (a) and (b) are used to estimate noise parameters in inference. The regression in (c) is used to randomly choose noise parameters in training.

2. Full Training and Implementation Details

To train our array of models we use the following configuration:

- The training ground truth images consist of unprocessed MIRFLICKR [5] with modified white balance gains for our target sensor, SIDD [1], and the Learning to See in the Dark training dataset [4]. The latter two datasets are unified into an RGGB pattern with Bayer Unification [8]. These images are randomly cropped into 128 x 128 patches.
- Bayer Augmentation [8] is applied to the training data.
- The input to the model is generated by adding artificial noise to the ground truth image. A description of our experiments to realistically model artificial noise is included in supplemental Section 1.
- The k-sigma transform of [11] is implemented in training and testing. Note that k and σ^2 refer to our a and b noise parameters respectively.
- Training examples are collated into batches of size 16.
- The Charbonnier Loss variant of our models uses a loss weight of 393.5. The Feature Matching Loss variant of our models uses a loss weight of 78.7. These weights were derived from a hyperparameter sweep on RAW PSNR.
- Models are trained using the Adam [7] optimizer with the maximum learning rate of 1e-4.
- Training occurs in 2,500,000 iterations scheduled with cosine learning rate decay.

3. Test Dataset Details

We collect noisy-clean image pairs on the targeted camera of our method which uses a Sony IMX258 sensor. Pairs

are collected by taking a short exposure with a random gain between 1.0 and 64.0 accompanied by a long exposure of the same scene with a gain of 1.0. The long exposure's exposure time is adjusted such that the brightness of both images are equivalent. The gain and exposure time is selected programatically so that the camera is not moved slightly between image captures. Similar to the Darmstadt Noise Dataset [10], to account for small environmental vibrations occurring between the short and long exposure that misalign the pair, we predict and correct for a global 2D translation estimated by averaging the Lucas-Kanade [9] optical flow of features detected by the Shi-Tomasi [6] algorithm. Finally, noise level annotations a and b for the short exposure are estimated from the regression described in supplemental Section 1.

References

[1] Abdelrahman Abdelhamed, Stephen Lin, and Michael S. Brown. A high-quality denoising dataset for smartphone cameras. In *CVPR*, June 2018.

[2] L. Azzari and A. Foi. Gaussian-cauchy mixture modeling for robust signal-dependent noise estimation. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5357–5361, 2014.

[3] Tim Brooks, Ben Mildenhall, Tianfan Xue, Jiawen Chen, Dillon Sharlet, and Jonathan T Barron. Unprocessing images for learned raw denoising. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11036–11045, 2019.

[4] Chen Chen, Qifeng Chen, Jia Xu, and Vladlen Koltun. Learning to see in the dark. In *CVPR*, 2018.

[5] Mark J. Huiskes and Michael S. Lew. The mir flickr retrieval evaluation. In *MIR '08: Proceedings of the 2008 ACM International Conference on Multimedia Information Retrieval*, New York, NY, USA, 2008. ACM.

[6] Jianbo Shi and Tomasi. Good features to track. In *CVPR*, pages 593–600, 1994.

[7] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017.

[8] Jiaming Liu, Chi-Hao Wu, Yuzhi Wang, Qin Xu, Yuqian Zhou, Haibin Huang, Chuan Wang, Shaofan Cai, Yifan Ding, Haoqiang Fan, and Jue Wang. Learning raw image denoising with bayer pattern unification and bayer preserving augmentation, 2019.

[9] Bruce D. Lucas and Takeo Kanade. An iterative image registration technique with an application to stereo vision. In *Proceedings of the 7th International Joint Conference on Artificial Intelligence - Volume 2*, IJCAI'81, page 674–679, San Francisco, CA, USA, 1981. Morgan Kaufmann Publishers Inc.

[10] Tobias Plötz and Stefan Roth. Benchmarking denoising algorithms with real photographs, 2017.

[11] Yuzhi Wang, Haibin Huang, Qin Xu, Jiaming Liu, Yiqun Liu, and Jue Wang. Practical deep raw image denoising on mobile devices, 2020.