Enhancing Low Light Videos by Exploring High Sensitivity Camera Noise

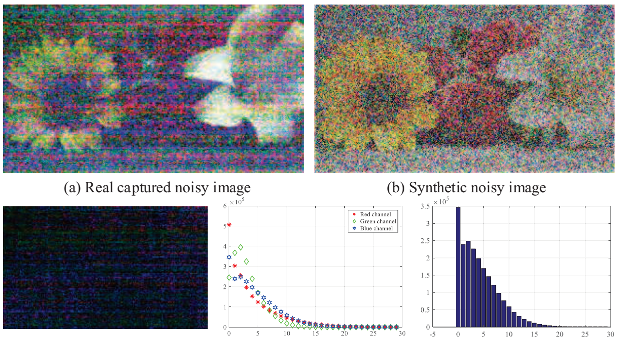
Wei Wang[1] Xin Chen[1] Cheng Yang[1,2] Xiang Li[1,2] Xuemei Hu[1] Tao Yue[1,2]
[1]Nanjing University, Nanjing, China
[2]NJU institute of sensing and imaging engineering, Nanjing, China
{MF1723052, MF1723008, DG1623027}@smail.nju.edu.cn {xiangli, xuemeihu, yuetao}@nju.edu.cn

Abstract

Enhancing low light videos, which consists of denoising and brightness adjustment, is an intriguing but knotty problem. Under low light condition, due to high sensitivity camera setting, commonly negligible noises become obvious and severely deteriorate the captured videos. To recover high quality videos, a mass of image/video denoising/enhancing algorithms are proposed, most of which follow a set of simple assumptions about the statistic characters of camera noise, e.g., independent and identically distributed (i.i.d.), white, additive, Gaussian, Poisson or mixture noises. However, the practical noise under high sensitivity setting in real captured videos is complex and inaccurate to model with these assumptions. In this paper, we explore the physical origins of the practical high sensitivity noise in digital cameras, model them mathematically, and propose to enhance the low light videos based on the noise model by using an LSTM-based neural network. Specifically, we generate the training data with the proposed noise model and train the network with the dark noisy video as input and clear-bright video as output. Extensive comparisons on both synthetic and real captured low light videos with the state-of-the-art methods are conducted to demonstrate the effectiveness of the proposed method.

1. Introduction

Under extremely low light conditions, high ISO setting is commonly adopted for capturing videos. However, with a high sensitivity, the dynamic streak noise (DSN), color channel heterogeneous and clipping effect, which are commonly negligible, become significant and deteriorate the quality of captured videos dramatically. For example, Fig. 1(a) is a real-captured high ISO image in low light environment, and Fig. 1(b) is the corresponding clear image under sufficient light condition, deteriorated with common Gaussian noise model. The noise distribution in Fig. 1(a) is obviously different than that in Fig. 1(b), including dynamic streak noise (Fig. 1(c)), color heterogenous (Fig. 1(d)) and clipping effect (Fig. 1(e)).



(a) Real captured noisy image (b) Synthetic noisy image

(c) Dynamic streak noise (d) Channel heterogeneous (e) Clipping effect

Figure 1. Characteristic of noise distribution in low-light imaging. (a) Video frame captured by Canon 5D mark III with ISO 25600 under low light condition. (b) Video frame synthsized with i.i.d. AWGN of approximately equal noise variance. (c) The dark field frame of Canon 5D mark III, with obvious horizontal streak noise, changing from frame to frame. (d) Intensity histograms of different color channels of a certain line, showing color heterogeneous among channels. (e) The intensity histogram of a low light image, with observed clipping effect.

In this paper, we explore the physical origin of the high sensitivity noise of both rolling and global shutter cameras, propose a novel noise model for handling the high sensitivity noise (i.e. dynamic streak noise, color heterogeneous, and clipping effect) in low light imaging, and build the estimation methods for estimating the model parameters of practical cameras. With the calibrated high sensitivity noise model, we generate the training dataset that could preciously simulate the real acquisition process of videos under low light environments, and train a LSTM-based video denoising network for enhancing the low light videos. We thoroughly evaluate the proposed method on both synthetic and real-captured videos and demonstrate the superiority of the proposed method via comparing with the state-of-the-art methods. In all, the main contributions of this paper are:

- We *propose* a practical high sensitivity noise model based on the capturing process and the hardware characteristic of sensor, which could accurately model the complex noises in the low-light imaging scenario.

- We *propose* the estimation method for the high sensitivity noise model, and synthesize the noisy/enhanced training dataset in low light environment that could well-approximate the practical capturing process.
- We *develop* an LSTM-based video enhancing neural network and *d*emonstrate the effectiveness of our method on both synthetic and real captured videos.

2. Related Work

Noise modeling. In the past decades, plenty of methods are proposed to denoise images/videos, such as nonlocal self-similarity [2, 6], sparse representation [6, 12, 16, 23] and Markov random field [11, 21]. Most of them are built on a simple noise model, i.e., the independent and identically distributed (i.i.d.) additive white Gaussion noise (AWGN).

To describe the noise more precisely, a set of complex noise models have been proposed. Liu *et al*. [13] extend the AWGN assumption by exploring the relationship between noise level and image brightness. Zhu *et al*. [26] model the pixel-wise noise with mixture of Gaussian distribution (MoG), which can approximate large varieties of continuous distributions. Mäkitalo and Foi [18] use the mixture of Poisson and Gaussian distributions to model the pixel-wise noise. Luisier *et al*. [14] as well as Mäkitalo and Foi [17] also assume the noise follows the mixture of Poisson and Gaussian distributions. Besides, the signal dependent Gaussian distribution is studied as well [1, 7] and the noise clipping effect is eliminated by reproducing the non-linear response of the sensor. Plötz and Roth [20] utilize a similar model and build a benchmark dataset by rectifing the noise bias of the model. However, the specific noise characters of practical high sensitivity noises are not explored well, limiting their application for noisy images/videos captured in extremely low light conditions.

Recently, the deep learning based methods achieve significant progress in many image processing tasks, including denoising as well. Chen *et al*. [4] directly get the training data by capturing noisy and clean image pairs with short and long exposures by a specific camera, so that their network can handle the real noise of that camera at high ISO setting. However, due to the requirement of long exposure for capturing clean images (10 to 30 seconds as in the paper), this method cannot be applied to the video denoising scenario. Chen *et al*. [5] take the advantage of generative adversarial network (GAN) to generate noisy images as training data with the similar noise distribution of real captured image. However, since the GAN network utilized in the paper is locally supported, it cannot deal with DSN which has the large scale spatial correlations.

Video denoising. Traditional video denoising methods usually take advantages of the self-similarity and redundancy of adjacent frames, and meanwhile, incorporate motion estimation into processing, such as [3, 15, 23]. Maggioni *et al*. [15] propose a multi-frame method (so-called VBM4D) based on finding groups of similar patches across the entire video sequence. Buades *et al*. [3] make use of motion estimation algorithms and patch-based methods for denoising, which introduce patch comparison and adapted PCA based transform to help motion estimation and global registration. Wen *et al*. [23] propose a video denoising method based on an online tensor reconstruction scheme with a joint adaptive sparse and low rank model.

Recently, the deep neural network has been applied in various low level vision tasks including video denoising and achieves impressive results. Clement *et al*. [9] and Mildenhall *et al*. [19] propose to recover a single clear frame from a set of frames in a burst manner. Since both of these methods assume that the image sequence is captured in a burst mode, they could not process videos with very large motion. Xue *et al*. [24] propose to denoise videos using a task-oriented flow (TOFlow) based network, which learns the motion representation in a self-supervised and task-specific manner, and thus can handle the large motion much better.

All of these methods are based on the traditional noise model, i.e., AWGN, Poisson or Mixture model, which cannot handle the noisy videos captured with large ISO setting in low light environment.

3. High Sensitivity Noise Model

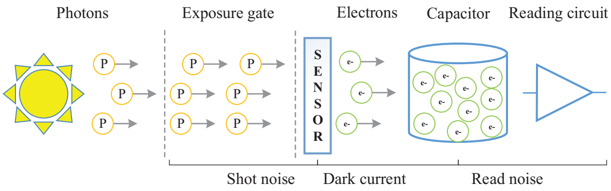


Figure 2. Noise sources of the entire imaging process.

Fig. 2 shows the common noise sources of the entire imaging process: 1) During the exposure time, N_p photons arrive on a pixel area. These photons generate N_e electrons with a quantum efficiency rate η. Both the photon arrival and photoelectron generation process are stochastic processes and follow the Poisson distribution, leading to a signal dependent Poisson noise, i.e. shot noise. 2) Due to the character of semiconductor device, there is little electric current generated from the random generation of electrons and holes within the depletion region, but not caused by the photons arrived at the pixel, leading to a signal independent Poisson noise, a.k.a. dark current. 3) The electrons are read out and amplified in the form of voltage or current signal, where the readout noise is introduced.

Based on this physical process, the basic practical noise model taking the main types of noise into consideration is:

$$y^i = K(S^i + D^i + R^i), \tag{1}$$

where i is the pixel index, y^i is the captured pixel value. $S^i \sim \mathcal{P}(N_e^i)$ denotes the shot noise and N_e^i is the expected number of photoelectrons in pixel i. $D^i \sim \mathcal{P}(N_d)$ denotes the dark current and N_d is the expected number of dark current electrons per pixel. $R^i \sim \mathcal{G}(0, \sigma_r^2)$ represents the read noise, of Gaussian distribution, and σ_r^2 is the variance. K is the system gain (unit: DN/e^-), commonly assumed to be of a uniform value over pixels and channels.

3.1. Noise Model in low-light imaging

In low light imaging, high sensitivity camera setting (i.e. high ISO) is required and previously negligible noise becomes significant in the captured videos. In this section, we model those observed high sensitivity noise and developed a comprehensive noise model under low light condition.

Dynamic streak noise. As shown in Fig. 1(a), dynamic streak noise (DSN) commonly becomes significant in low light imaging, deteriorating the quality of images with horizontal streaks and changing dynamically from frame to frame. It appears in both rolling-shutter consumer cameras (e.g., Canon 5D Mark III) and global-shutter consumer cameras (e.g., Grasshopper3 GS3-U3-32S4C).

In different shutter mode, the DSN possesses slightly different statistical characteristics, i.e. white or colored on the row-wise frequency domain. In another word, regarding a certain column as a 1-D noise signal, it has either equal or unequal intensity at different frequences. Specifically, for the rolling shutter, since the sensor is exposed and readout row by row, the operation sequences of rows are sequentially delayed and the sensor has more time to read the pixels, leading to a slower reading speed. In this case, the $1/frequence$ noise (i.e., the power spectral density is inversely proportional to the frequency of the noise, $1/f$ for short) which often appear in relative low-frequency circuits becomes significant and non-negligible, exhibiting colored noise in the vertical frequency domain. In contrast, the global shutter sensors have to read faster and thus resulting in a much smaller $1/f$ noise, which therefore can be neglected in our model, presenting a white vertical frequency character.

According to the streak fluctuation characteristic of the DSN, we propose to handle it by using a row-wise gain fluctuation model,

$$K^r = K + \lambda K^r_{1/f} + (1-\lambda)K^r_{white}, \tag{2}$$

where K is the globally constant system gain, $K^r_{1/f}$ is the $1/f$ gain fluctuation following a colored Gaussian distribution, and K^r_{white} is the white gain fluctuation with the white Gaussian distribution. λ is the weight between the $1/f$ and white components. In this paper, we find that the $1/f$ component dominate the DSN for rolling shutter cameras while the white component becomes major for the global shutter ones. Accordingly, we choose $\lambda = 1$ for rolling shutter cameras and $\lambda = 0$ for global shutter.

Besides, since both $K^r_{1/f}$ and K^r_{white} follow zero-mean Gaussian distribution, the expectation of K^r is exactly K, to facilitate the parameter estimation we rewrite Eq. 2 as

$$K^r = K\beta^r, \tag{3}$$

where β^r is the fluctuation factor of DSN, and equals to either $1 + K^r_{1/f}/K$ for rolling shutter or $1 + K^r_{white}/K$ for global shutter. Since both $1/f$ noise and white noise obey to Gaussian distribution, β^r follow a colored or white Gaussian distribution $\mathcal{G}(1, \sigma_{beta})$ respectively.

Noise relationship between channels. Here, we propose to model the noise relationship between channels by exploring the physical characters of sensors, which takes both the pixel uniformity and channel difference into consideration.

Specifically, most of color cameras capture three channels by covering a Color Filter Array (CFA) on a uniform silicon sensor[1], which means that all the noise sources after the photons arriving the semiconductor device should be consistent. But considering the silicon devices has unbalanced responses in different color channels, many camera amplify three channel with different gains to correct color bias, so that we change Eq. 3 to its color channel version, i.e., $K^r_c = K_c\beta^r_c$.

Clipping effect. The digital sensors always have positive measurements. However, when the read noise which follows zero-mean Gaussian distribution is considered, the negative values may appear for very weak signals theoretically, leading to a clipped noise distribution as shown in Fig. 1(e) in practice. Mathematically, the clipping operation $\mathcal{T}(\cdot)$ can be expressed by

$$\mathcal{T}(x) = \begin{cases} x, & if \quad x > 0 \\ 0, & otherwise. \end{cases} \tag{4}$$

According to the above analysis, our proposed practical noise model in low-light conditions becomes,

$$y^i = \mathcal{T}(K_c\beta^r_c(S^i + D^i + R^i))|_{c\in\{r,g,b\}}, \tag{5}$$

3.2. Model parameter estimation

In the following, we will show how to estimate the parameters of our high sensitivity noise model for a real camera. To facilitate the inference, we discuss the parameter estimation method without considering the clipping operation first. Then, we propose a 2D look-up table based method to correct the estimation bias caused by clipping effect.

[1] Note that in this paper we mainly focus on the three channel color cameras with bayer CFA, but the method can be easily extended to other kinds of CFAs.

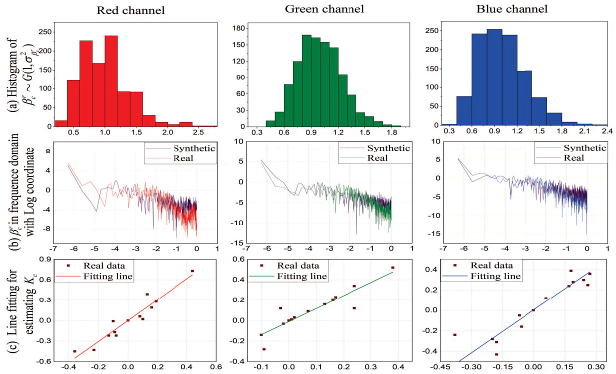


Figure 3. Parameter calibration for Canon 5D Mark III.

Calibrating β_c^r. Capturing a dark field image, we have the measurements $y^i = K_c\beta_c^r(D^i + R^i)|_{c\in\{r,g,b\}}$. The expectation of y^i of whole image is $(\mathrm{E}[y] = K_c\mathrm{E}[\beta_c^r](\mathrm{E}[D] + \mathrm{E}[R]) = K_c(\mathrm{E}[D] + \mathrm{E}[R])$, and the expectation of $y^{i\in r}$ of r-th row is $\mathrm{E}[y]_{i\in r} = K_c\beta_c^r(\mathrm{E}[D] + \mathrm{E}[R]))$. Thus, by dividing the mean of y^i in row r by the global mean of y^i, i.e., $\beta_c^r = \mathrm{mean}(y^{i\in r})/\mathrm{mean}(y^i)$, β_c^r can be computed. Given β_c^r, we can remove the DSN from dark field image by dividing the measurements of each row $y^{i\in r}$ by β_c^r, deriving a DSN corrected measurement ${y'}^i = K_c(D^i + R^i)|_{c\in\{r,g,b\}}$ (Please note that the following calibrations are all based on the corrected measurements). As shown in Fig. 3(a), we calibrate the β_c^r of Canon 5D Mark III, and present the histograms of three color channels accordingly. According to the shutter type of the camera, the random β_c^r can be generated with either $1/f$ (like Canon 5D which is rolling shutter, as shown in Fig. 3(b). Like the $1/f$ noise, the envelope of the Fourier transform of β_c^r presents as a descending line in the logarithm coordinate system.) or white (for global shutter) Gaussian distribution.

Calibrating K_c. Since $D^i \sim \mathcal{P}(N_d)$ and $R^i \sim \mathcal{G}(0, \sigma_r^2)$, the expectation and variance of corrected measurement y' can be denoted by

$$\begin{aligned}\mathrm{E}[y'] &= K_c N_d \\ \mathrm{Var}[y'] &= K_c^2(N_d + \sigma_R^2).\end{aligned} \tag{6}$$

Substituting N_d with $N_d = \mathrm{E}[y']/K_c$, we have

$$\mathrm{Var}[y'] = K_c\mathrm{E}[y'] + K_c^2\sigma_R^2. \tag{7}$$

Considering the expectation of dark current N_d varies with different exposure time, we can have two equations by applying Eq. 7 on two dark field images with different exposure time, and the constant term $K_c^2\sigma_R^2$ can be eliminated from the difference between them,

$$\Delta\mathrm{Var}[y'] = K_c\Delta\mathrm{E}[y'], \tag{8}$$

where $\Delta\mathrm{Var}[y'] = \mathrm{Var}[y'_{t_1}] - \mathrm{Var}[y'_{t_2}]$ and $\Delta\mathrm{E}[y'] = \mathrm{E}[y'_{t_1}] - \mathrm{E}[y'_{t_2}]$, in which t_1, t_2 denote exposure time. In practice, by replacing $\mathrm{E}[y']$ by $\mathrm{mean}(y')$, and $\mathrm{Var}[y']$ by $\mathrm{var}(y')$, we can derive both $\Delta\mathrm{Var}[y']$ and $\Delta\mathrm{E}[y']$ from two dark field videos with different exposure times. By capturing a set of differently exposured dark field images, we can compute a set of points $(\Delta\mathrm{E}[y'], \Delta\mathrm{Var}[y'])$, and K_c can be estimated by fitting a line from these points, as shown in Fig. 3(c).

Calibrating N_d **and** σ_R^2. Given K_c, N_d and σ_R^2 can be computed easily from Eq. 6.

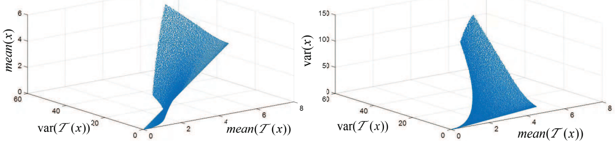


Figure 4. 2D look-up tables for correcting the clipping effect of $\mathrm{mean}(\cdot)$ (left) and $\mathrm{var}(\cdot)$ (right).

Clipping correction by looking up tables. Further, let's consider the clipping operation. Obviously, $\mathrm{mean}(\cdot)$ and $\mathrm{var}(\cdot)$ of a clipped variable $\mathcal{T}(x)$ can significantly deviate from unclipped $\mathrm{mean}(x)$ and $\mathrm{var}(x)$. Generally, it is difficult to eliminate this effect without any knowledge of x. However, in our cases, all the random variables inside $\mathcal{T}(\cdot)$ can be divided into two components, i.e., a Poisson component and a zero-mean Gaussian component. Therefore, the expectation and variance of the combined distribution can describe the entire distribution sufficiently. Therefore, we can generate a set of x with different expectations and variances, and compute two 2D table from the clipped estimates $\mathrm{mean}(\mathcal{T}(x))$ and $\mathrm{var}(\mathcal{T}(x))$ to the real $\mathrm{mean}(x)$ and $\mathrm{var}(x)$. Fig. 4 shows the 2D look-up table for correcting $\mathrm{mean}(\cdot)$ (left) and $\mathrm{var}(\cdot)$(right). Then, by looking up the two 2D tables according to $\mathrm{mean}(\mathcal{T}(x))$ and $\mathrm{var}(\mathcal{T}(x))$ computed from the real data, we can derive the real $\mathrm{mean}(x)$ and $\mathrm{var}(x)$ easily. Specifically, for calibrating β_c^r, $\mathrm{mean}(y^{i\in r})$ and $\mathrm{mean}(y^i)$ are required, and both $y^{i\in r}$ and y^i follow mixture of Poisson and Gaussian, can thus can be corrected by looking up the 2D table according to $\mathrm{mean}(\mathcal{T}(y^{i\in r}))$, $\mathrm{var}(\mathcal{T}(y^{i\in r}))$, $\mathrm{mean}(\mathcal{T}(y^i))$ and $\mathrm{var}(\mathcal{T}(y^i))$. Similarly, we can correct $\mathrm{mean}(y')$ and $\mathrm{var}(y')$ in Eq. 8 for calibrating K_c as well.

Training data generation. In this paper, we generate the training data for video enhancing network according to our practical noise model. Given the model in Eq. 5 and the parameters calibrated from a certain camera, we can generate noisy videos from a clean video by Monte Carlo simulation. Before that, we should first derive the expected photonelectron number N_e. Given an illuminance I of a certain low light environment, the expected number of photoelectrons $\mathrm{E}[N_e]$ generated on a certain pixel can be computed by

$$\mathrm{E}[N_e] = \frac{I \cdot S_{pixel}}{c_{lum2radiant} \cdot E_p \cdot \eta}, \tag{9}$$

where S_{pixel} is the area of a pixel, $c_{lum2radiant}$ is the transfer constant from luminous flux to radiancy, E_p is the energy of a single photon[2] and η is the quantum efficiency of the camera. By adjusting the average value of an image to $\mathrm{E}[N_e]$, we can derive the expected incoming photon numbers of all the image pixels. Then, we can get the noisy image by Monte Carlo simulation according the Eq. 5.

4. Our LSTM-based Network

In this section, we present an LSTM-based video enhancing network which recovers the clear and bright videos from the noisy dark ones captured by real digital cameras.

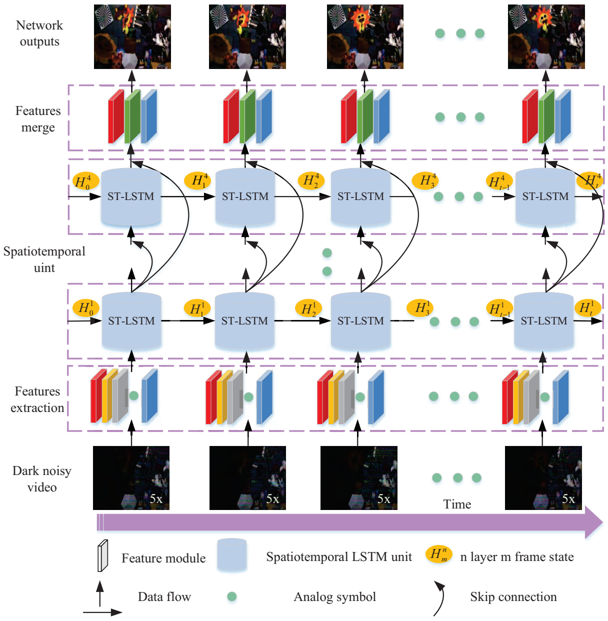


Figure 5. Our proposed LSTM-based network. The network extracts both the short-term and long-term dependencies from videos and combines spatial-temporal information to reconstruct high quality videos form dark noisy ones. To make the dark frame visible, we demonstrate the right half of each input frame with 5x amplification.

Network Architecture. Fig. 5 shows an overview of the proposed LSTM-based network. Our model takes the low-light videos with arbitrary length as input and generates output frames in an online manner. To simultaneously and adaptively extract the short-term and long-term dependencies from videos, we integrate a Spatio-Temporal LSTM (ST-LSTM) unit [22] into our video denoising network. ST-LSTM can model both spatial and temporal representations in a unified memory cell and convey the memory both vertically across layers and horizontally over states. Our network consists of two strided convolutional layers and four ST-LSTM layers. First, the convolutional layer extract features of the input frames, and then pass the features into ST-LSTM layers. Skip connection is added in the spatial correlations. The last convolutional layer transforms the reconstructed information into standerd RGB image format to derive the final result. We use filters with 3×3 kernel size for the 1st and 4th ST-LSTM layers, and filters with 5×5 kernel size for the 2nd and 3rd ST-LSTM layers. Each ST-LSTM layer has 64 feature maps in our network. Zero padding is adopted to ensure the consistence between dimensions of the input and output.

Loss Functions. We train the proposed LSTM-based network by minimizing the whole loss defined in Eq.10 between the network output frame I and the corresponding ground truth I^*. We define the basic loss function as the weighted average of $\mathcal{L}_2$ and $\mathcal{L}_1$ losses. Both $\mathcal{L}_2$ and $\mathcal{L}_1$ distances focus on pixel intensity consistency, and the former makes the output smooth while the latter keeps more details. In order to further improve the perceptual quality, we introduce the perceptual loss $\mathcal{L}_{per}$ [8] which constrains the difference between high-level features of I and I^* extracted by a pre-trained Visual Geometry Group(VGG) Network. Besides, the total variation regularizer $\mathcal{L}_{tv}$ is added in our loss functions as a smooth regularization term. Then, our final loss function becomes

$$\mathcal{L}=\sum_{i=1}^{N}\alpha\mathcal{L}_2(I_i,I_i^*)+\beta\mathcal{L}_1(I_i,I_i^*)+\gamma\mathcal{L}_{per}(I_i,I_i^*)+\delta\mathcal{L}_{tv}(I_i), \tag{10}$$

where α, β, γ and δ are hyper-parameters for the training process, and N is the number of frames in a sequence. Here, we set $\alpha = 5$, $\beta = 1$, $\gamma = 0.06$, $\delta = 2 \times 10^{-6}$ and $N = 8$ in the training process.

Training details. The network is implemented with Pytorch. We train the network from scratch using the loss function above and the Adam optimizer [10] under a learning rate of 1×10^{-3}. We collect large number of clean videos, and select about 900 sequences which are abundant in moving scenes. Then, based on the practical noise model, we generate both dark and noisy sequences from these clean videos. Considering that each camera has a unique set of noise parameters, we train the network with different training data for different cameras. In this paper, two representative cameras, Canon 5D Mark III and Grasshopper3 GS3-U3-32S4C, are used to calibrate the parameters and train the network. To guarantee the generalization performance, we introduce a slight fluctuation in the noise parameters randomly when generating the training data. The network is trained by $8-$frame sequences, and can deal with videos with infinite frames in testing. In the training process, we set batch size to 8 and patch size to $[64, 64, 3]$.

5. Experiments

In this section, we conduct exhaustive comparisons, both quantitatively and qualitatively, based on the Canon 5D

[2] In this paper, we use the energy of $E_p = hc/\lambda, \lambda = 555nm$, and $c_{lum2radiant} = 683lm/w@555nm$ for a rough estimation.

Mark III camera. In addition, to demonstrate the feasibility of proposed method, the enhancing results of videos captured by Grasshopper3 GS3-U3-32S4C are presented as well. The noise parameters of the camera model are calibrated according to Sec. 3.2 (shown in Tab. 2), and the training datasets are generated with these parameters through Monte Carlo method respectively. The proposed network are trained upon the training datasets and extensive experimental comparisons are implemented to demonstrate the superiority of the proposed method. Specifically, we analyze the effect of the proposed noise model and the network individually, demonstrate the proposed method under different luminance levels and thoroughly demonstrate the effectiveness of our method on real captured videos over various scenes. In our experiments, we compare our method with both representative conventional methods, i.e. VBM4D [15], and state-of-the-art learning based methods, i.e. FFDNet [25], KPN [19], and TOFlow [24].

First, we analysis the effect of the proposed noise model and the proposed network individually. Limited by the paper length, only part of the compairsons about the Canon 5D Mark III camera model are given here. Additional analysis about Grasshopper3 camera model and more results of Canon camera can be found in the supplementary material.

Effect of network individually. To demonstrate the superiorities of the proposed network individually, we first compare our network with VBM4D [15], FFDNet [25], KPN [19] and TOFlow [24] upon the proposed noise model. Note that for fair comparison, the parameters of VBM4D [15] are choosen at the best performance and the learning based method, i.e. FFDNet [25], KPN [19], TOFlow [24] and our network, are trained on the same dataset and tested on the same noisy videos. Here we introduced 'refer' (short for reference) as the comparison baseline. 'refer' is the scaled input video and the scaling factor is the total intensity ratio of the groundtruth images over the input images. As shown in Fig. 6, at least 2dB/0.05 improvement in PSNR/SSIM are introduced compared with the other methods. As shown in Fig. 7, the proposed network could help to recover the videos with much higher visual quality, with more structural details such as the animal face contour and the sharp textures on the shoulder, and in higher color fidelity. As a whole, the performance improvement introduced by the proposed network is demonstrated both quantitatively and qualitatively.

Effect of noise model. Then, to further verify the effectiveness of the proposed noise model individually, we train the network of TOFlow [24] and our network model upon different noise models, i.e. AWGN model, mixture of Gaussian and Poisson noise model, and the proposed noise model. The real captured dark noisy video is denoised by the trained network models as shown in the 1-3 th columns in Fig. 8. Note that we capture an additional image of the same scene with the light turned on for reference of scene structure (Fig. 8, marked with 'Real data with light'). With the same network (i.e. TOFlow or our network), the results of proposed noise model are of the best quality in terms of much less chrominance artifacts, more structual details and higher contrast, validating the superiorities of the proposed noise model for enhancing videos in low light condition.

Table 1. Camera settings and environment conditions applied in the experiments.

Indoor conditions and settings				
Camera	ISO/Gain	Exposure time	Illuminance	F-number
Canon 5D	25600	50ms	0.05-0.2Lux	5.6
Grasshopper3	48dB	30ms	0.01-0.03Lux	1.8

Table 2. Parameters of our practical noise model calibrated of Canon 5D Mark III and Grasshopper3.

Parameters	Grasshopper3			Canon 5D		
	R	G	B	R	G	B
$\sigma^2_{\beta^r_c}$	0.0081	0.0069	0.026	0.197	0.039	0.013
K_c	3.54	3.11	3.95	1.53	1.23	1.31
N_d	118/s			51.7/s		
σ^2_R	5.43			12.5		

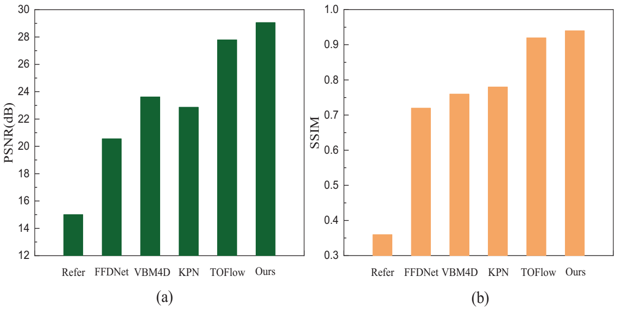


Figure 6. Quantitative comparisons on the effect of network individually.

In all, we demonstrate that the superiorities of the proposed method are attributed to both the proposed network and the proposed noise model.

Performance Analysis on Synthetic Data. We test our method on the synthetic test dataset generated with our practical noise model, and compare it with the other state-of-the-art algorithms. The comparisons are conducted on six test videos generated by simulating the environment illuminance from 0.05 to 0.2 Lux. Since our network is trained to recover the bright clear videos from dark noisy inputs, the brightness of the output has been adjusted adaptively by the network itself. For fair comparison, we add brightness scaling of input frames and results of other methods to the truth frames and the fidelity metrics(PSNR and SSIM) of the other methods are computed after light enhancement.

As shown in Fig. 9, the proposed method achieves the highest scores in both PSNR and SSIM, over all the test videos and different luminance levels. Further scrutinizing the comparisons among our method, TOFlow and TOFlow with our noise model, we could find that the performance improvement comes from both the proposed network and

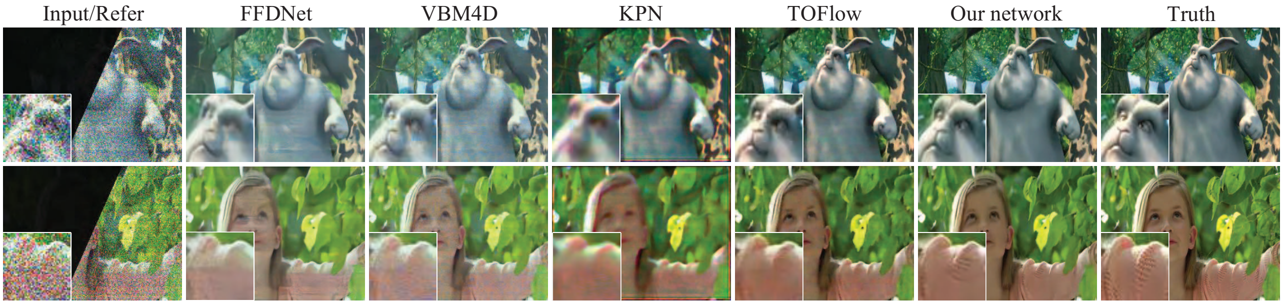


Figure 7. Qualitative comparisons on the effect of network individually. To facilitate visualization of the original input frames, we show in the first column the left halves of dark noisy input frames and the right halves of the brightness-scaled 'refer' frame.

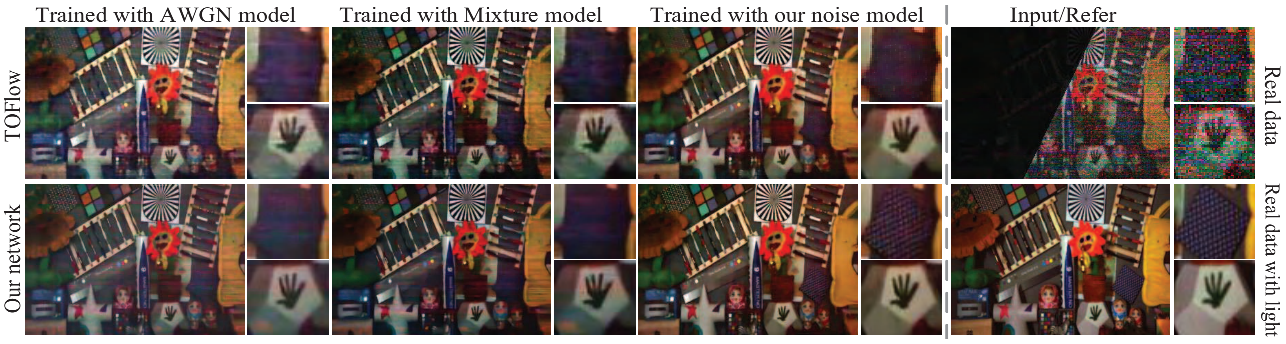


Figure 8. Comparisons on the effect of the proposed noise model. 'Real data with light' denotes the same scene with the light turned on and the same camera setting parameters.

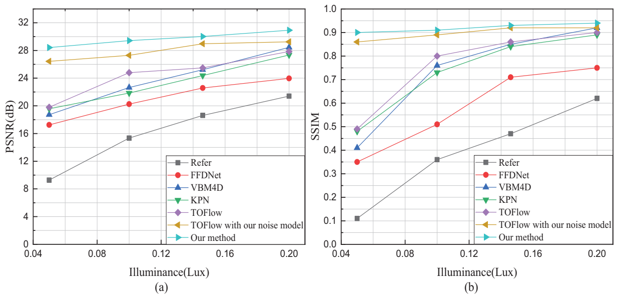


Figure 9. Quantitative comparisons with the other methods under different illuminance intensities.

the proposed noise model, and the improvement from the proposed noise model are larger than from the proposed network, demonstrating the importance of our noise model.

Fig. 10 presents the performance of our method on individual frames. It is obviously that the proposed method can memory the information of previous frames and handle the motion between frames well to elevate the results of current frame. Thus at the very begining of the sequence, the PSNR of proposed method rise frame by frame. After about 20 frames, the curve become flat because the information of 20 frames before cannot provide much more useful informaiton for current frame processing.

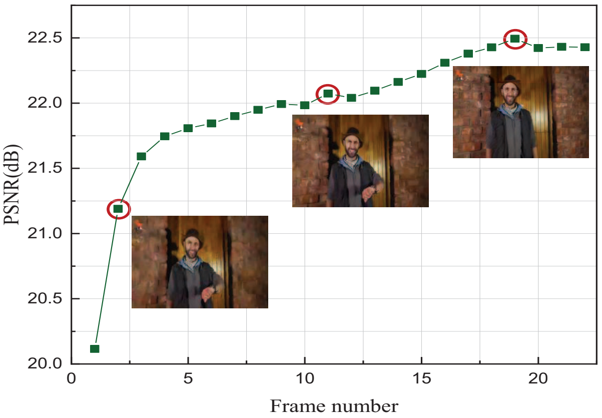


Figure 10. PSNR of frames enhanced by proposed method.

Experiments on Real Captured Videos. As shown in Fig. 11, we show the experimental results with both the Canon 5D MarkIII and Grasshopper3 GS3-U3-32S4C camera models. As can be seen, the results of the other methods lose many details on the extremely noisy region and cannot deal with the stripe noise well in the flat region, while our method could remove the noise effect clearly while preserving the details of images well. Color bias are also obviously observed in the other methods, which could probably be caused by the inaccuracy of the noise model. Since our method utilize a more accurate noise model under low light condition, our method could recover the images with higher color fidelity. In all, the effectiveness of our method is demonstrated in both cameras (including rolling shutter and global shutter), over real captures scenes.

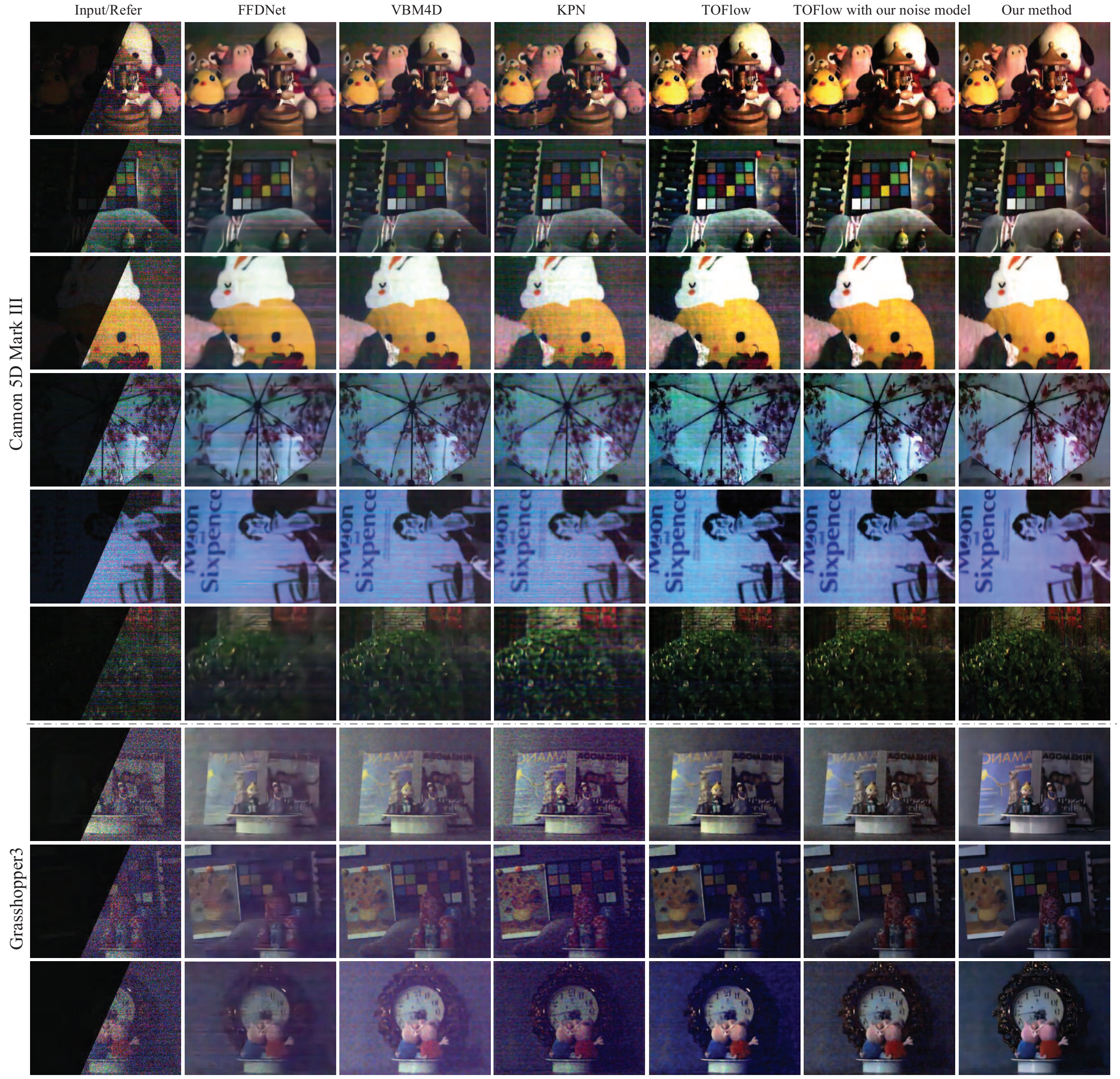


Figure 11. Results on real videos captured by Canon 5D MarkIII and Grasshopper3 GS3-U3-32S4C.

6. Discussion and Conclusion

In this paper, we propose a low light video enhancing method by exploring the high sensitivity camera noise in low light imaging. A high sensitivity noise model and the corresponding estimation method are proposed to generate dark-noisy/enhanced training datasets. An LSTM-based neural network is proposed to trained upon the generated dataset and enhance the real-captured low light noisy videos. We conduct thorough experiments and demonstrate the effectiveness of the proposed method.

Currently, the proposed method requires to capture several videos by the camera beforehand for noise model calibration. In the future, we will investigate how to blindly estimate the noise parameters from input noisy videos.

Acknowledgments

We would like to acknowledge funding from NSFC Projects 61671236, 61971465, and National Science Foundation for Young Scholar of Jiangsu Province, China (Grant No. BK20160634), and Fundamental Research Funds for the Central Universities, China (Grant No.0210-14380128).

References

[1] Lucio Azzari and Alessandro Foi. Gaussian-Cauchy mixture modeling for robust signal-dependent noise estimation. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2014. 2

[2] Antoni Buades, Bartomeu Coll, and J-M Morel. A non-local algorithm for image denoising. In *International Conference on Computer Vision and Pattern Recognition*, 2005. 2

[3] Antoni Buades, Jose-Luis Lisani, and Marko Miladinović. Patch-based video denoising with optical flow estimation. *IEEE Transactions on Image Processing*, 25(6):2573–2586, 2016. 2

[4] Chen Chen, Qifeng Chen, Jia Xu, and Vladlen Koltun. Learning to see in the dark. In *International Conference on Computer Vision and Pattern Recognition*, 2018. 2

[5] Jingwen Chen, Jiawei Chen, Hongyang Chao, and Ming Yang. Image blind denoising with generative adversarial network based noise modeling. In *International Conference on Computer Vision and Pattern Recognition*, 2018. 2

[6] Weisheng Dong, Lei Zhang, Guangming Shi, and Xin Li. Nonlocally centralized sparse representation for image restoration. *IEEE Transactions on Image Processing*, 22(4):1620–1630, 2013. 2

[7] Alessandro Foi, Mejdi Trimeche, Vladimir Katkovnik, and Karen Egiazarian. Practical Poissonian-Gaussian noise modeling and fitting for single-image raw-data. *IEEE Transactions on Image Processing*, 17(10):1737–1754, 2008. 2

[8] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *International Conference on Computer Vision and Pattern Recognition*, 2016. 5

[9] Clément Godard, Kevin Matzen, and Matt Uyttendaele. Deep burst denoising. In *European Conference on Computer Vision*, 2018. 2

[10] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2014. 5

[11] Xiangyang Lan, Stefan Roth, Daniel Huttenlocher, and Michael J Black. Efficient belief propagation with learned higher-order markov random fields. In *European Conference on Computer Vision*, 2006. 2

[12] Huibin Li and Feng Liu. Image denoising via sparse and redundant representations over learned dictionaries in wavelet domain. In *International Conference on Image and Graphics Processing*, 2009. 2

[13] Ce Liu, Richard Szeliski, Sing Bing Kang, C Lawrence Zitnick, and William T Freeman. Automatic estimation and removal of noise from a single image. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2):299–314, 2008. 2

[14] Florian Luisier, Thierry Blu, and Michael Unser. Image denoising in mixed Poisson-Gaussian noise. *IEEE Transactions on Image Processing*, 20(3):696–708, 2011. 2

[15] Matteo Maggioni, Giacomo Boracchi, Alessandro Foi, and Karen Egiazarian. Video denoising, deblocking, and enhancement through separable 4-d nonlocal spatiotemporal transforms. *IEEE Transactions on Image Processing*, 21(9):3952–3966, 2012. 2, 6

[16] Julien Mairal, Francis R Bach, Jean Ponce, Guillermo Sapiro, and Andrew Zisserman. Non-local sparse models for image restoration. In *International Conference on Computer Vision*, 2009. 2

[17] Markku Mäkitalo and Alessandro Foi. Optimal inversion of the generalized anscombe transformation for Poisson-Gaussian noise. *IEEE Transactions on Image Processing*, 22(1):91–103, 2013. 2

[18] Markku Mäkitalo and Alessandro Foi. Noise parameter mismatch in variance stabilization, with an application to Poisson-Gaussian noise estimation. *IEEE Transactions on Image Processing*, 23(12):5348–5359, 2014. 2

[19] Ben Mildenhall, Jonathan T Barron, Jiawen Chen, Dillon Sharlet, Ren Ng, and Robert Carroll. Burst denoising with kernel prediction networks. In *International Conference on Computer Vision and Pattern Recognition*, 2018. 2, 6

[20] Tobias Plötz and Stefan Roth. Benchmarking denoising algorithms with real photographs. In *International Conference on Computer Vision and Pattern Recognition*, 2017. 2

[21] Stefan Roth and Michael J Black. Fields of experts: a framework for learning image priors. *International Journal of Computer Vision*, 82(2):205–229, 2009. 2

[22] Yunbo Wang, Zhifeng Gao, Mingsheng Long, Jianmin Wang, and Philip S Yu. Predrnn++: Towards a resolution of the deep-in-time dilemma in spatiotemporal predictive learning. In *International Conference on Machine Learning*, 2018. 5

[23] Bihan Wen, Yanjun Li, Luke Pfister, and Yoram Bresler. Joint adaptive sparsity and low-rankness on the fly: An online tensor reconstruction scheme for video denoising. In *International Conference on Computer Vision*, 2017. 2

[24] Tianfan Xue, Baian Chen, Jiajun Wu, Donglai Wei, and William T Freeman. Video enhancement with task-oriented flow. *International Journal of Computer Vision*, Feb 2019. 2, 6

[25] Kai Zhang, Wangmeng Zuo, and Lei Zhang. Ffdnet: Toward a fast and flexible solution for cnn-based image denoising. *IEEE Transactions on Image Processing*, 27(9):4608–4622, 2018. 6

[26] Fengyuan Zhu, Guangyong Chen, and Pheng-Ann Heng. From noise modeling to blind image denoising. In *International Conference on Computer Vision and Pattern Recognition*, 2016. 2