Low-light image restoration using discrete cosine transform and conditional generative adversarial network

Banglian Xu, Yao Fang, Zhixiang Bian, Yu Huang, Yaoyao Tan, Xue Cheng, Jiale Song and Zhang Leihong

College of Communication and Art Design, University of Shanghai for Science and Technology, Shanghai 200093, China

Abstract. In the process of low-light imaging, some part of useful information of an image is overwhelmed by a noise. When interference is large, the signal-to-noise ratio (SNR) detected in a system is reduced to a very low level. We study the low-light imaging under condition when the detection SNR is equal to 1 dB. Taking into account that the noise is often located in the high-frequency spectral part, we use discrete cosine transform (DCT) to remove the noise or, at least, filter out its some part. Then we use an algorithm of conditional generative adversarial network (CGAN) to improve the image quality. The simulation results testify that the DCT and CGAN algorithms combined together improve significantly the restoration results and the final quality of images. The latter is high enough, with the average peak SNR being higher than 22 dB and the structural similarity index measure amounting to about 0.8.

1. Introduction

Digital image is an important means for transmitting information. The images received by final customers are often affected by various factors, resulting in their decreasing quality. In particular, noise interference affects seriously the image quality during imaging process. At present, the researches on the image denoising of additive white Gaussian noise are mature. However, this is only partly true of photon-counting imaging systems, e.g. those for medical, remote-sensing, lidar and night imagings [1]. In any low-light imaging system, a receiver can only detect a small number of photons and the Poisson noise has a strong correlation with the useful signal, a situation which is difficult to deal with [2].

Image denoising is performed either in spatial or transform domains. The spatial-denoising methods use mainly some filters to process image pixels, e.g. such common ones as a mean filter, a Wiener filter or a bilateral filter [3]. Although this type of methods can remove the image noise to a certain extent, the edge information of the image becomes blurred after denoising. According to the principles of denoising in the transform domain, different characteristic coefficients of the image and the noise are being converted to the transform domain, and then the denoising process is performed on them. This technique includes a fast Fourier transform, a discrete cosine transform (DCT), a wavelet transform [4], etc. In other words, a noisy image is converted into the transform domain for its further processing. By retaining the coefficients of image and filtering out the coefficients of noise in the transform domain, one then performs the inverse transform to obtain a denoised image. At present, a number of computer technologies have been developed for this aim

and some novel image-denoising algorithms have been suggested. The examples are denoising algorithms based on sparse representation, non-local mean denoising [5], three-dimensional block-matched filtering [6], etc. These algorithms typically reveal an excellent denoising effect, although they are very complex computationally and require sophisticated optimization algorithms.

In the recent years, deep-learning technologies, flexible neural networks and the facilities to learn big data have been applied to the image processing, too. In 2009, Jain [7] et al. have used neural networks to obtain relatively good denoising results for different natural images. Burger [8] et al. have suggested a multi-layer perceptron method to achieve image denoising. Denoising convolutional neural network has been developed by Zhang et al. [9]. It combines a batch normalization and a residual learning network. It can be successfully applied to image denoising and solve some problems of gradient dispersion and information loss, so that the network can converge faster. In 2019, Jia [10] et al. have proposed a fractional optimal control-network model, which is designed basing on discretization of fractional differential equations, introduces multi-scale features and achieves quite good results. Finally Zamir et al. [11] have trained a new image-denoising network on some realistic synthetic data and achieved a state-of-the-art performance on real camera-benchmark datasets.

Below we will address the problems of the low-light imaging. Under low-light conditions, the final quality of images is typically extremely poor and, therefore, the images are to be restored. For this purpose, we first build an imaging model, where the noise conforms to the Poisson distribution. Then we employ a DCT-based denoising method to enhance the quality of low-light images. It considers the characteristics of both the image information and the noise and so filters the noise in the transform domain. Finally, we use an algorithm of conditional generative adversarial network (CGAN) to improve further the image quality. The technique developed by us improves greatly the image quality under the condition of as low signal-to-noise ratio (SNR) as 1 dB. Therefore, we have testified that the traditional denoising method combined with the deep learning method can improve significantly the overall image restoration results.

2. Low-light imaging model

In a low-light imaging system, the noise model should conform with the Poisson distribution, where the noise intensity is related to the pixel value of original image. When the illumination gets lower, a detector collects still fewer photons, and the Poisson noise becomes more difficult to remove. This is why our imaging model considers only the Poisson noise. As a result, our low-light imaging model is given by

$$I = P + N,\tag{1}$$

where I denotes the image acquired, P the original image and N the noise. We remind that the noise conforms to the Poisson distribution. Within a period T, its statistical model is given by

$$p(r,\lambda) = \frac{e^{-\lambda}\lambda^r}{r!},\tag{2}$$

where r is the number of photons received by a detector and λ the average rate of random events. Then the SNR of detection process for the entire system is as follows:

$$SNR = \frac{\text{mean}(P)}{D},\tag{3}$$

where $\text{mean}(P)$ refers to the original image (i.e., it is the mean value of the signal) and D is the standard deviation of the noise N.

3. DCT algorithm

The basis of the DCT is Fourier transform. Among orthogonal transformations applied to speech and image signals, the DCT transformation is considered to be optimal. The DCT and the inverse DCT are described respectively by the formulae

$$F(u,v) = c(u)c(v)\sum_{x=0}^{N-1}\sum_{y=0}^{M-1} f(x,y)\cos\left(\frac{\pi u(2x+1)}{2N}\right)\cos\left(\frac{\pi v(2y+1)}{2M}\right) \qquad (4)$$

and

$$f(x,y) = \sum_{u=0}^{N-1}\sum_{v=0}^{M-1} c(u)c(v)F(u,v)\cos\left(\frac{\pi u(2x+1)}{2N}\right)\cos\left(\frac{\pi v(2y+1)}{2M}\right), \qquad (5)$$

with

$$c(u) = \begin{cases} \sqrt{\dfrac{1}{N}}, u = 0 \\ \sqrt{\dfrac{2}{N}}, u \neq 0 \end{cases}, \quad c(v) = \begin{cases} \sqrt{\dfrac{1}{M}}, v = 0 \\ \sqrt{\dfrac{2}{M}}, v \neq 0 \end{cases}. \qquad (6)$$

Here u and v represent generalized frequency variables associated with two dimensions of image, $F(u,v)$ refers to the direct DCT transform and $f(x,y)$ to the pixel value in the original image. Concerning the other notations, M and N imply respectively the row and column sizes of $f(x,y)$, and $c(u)$ and $c(v)$ are the compensation coefficients that make the DCT-transformation matrix an orthogonal one.

4. CGAN algorithm

In the present work, we suggest a deep-learning method based on the CGAN algorithm to restore low-quality images and increase their quality [12]. The CGAN is a novel way to train generative models which can be constructed by simply feeding the data to both generator and discriminator. It has a very prominent improvement over the original algorithm of generative adversarial network and is more efficient than this counterpart. As a consequence, the CGAN plays an important role in many fields like image conversion and restoration, video prediction and data enrichment. The network framework consists of two parts: a restoration network G for restoring image quality and a discriminant network D for conditional adversarial learning. For a given data set $\{t(x), T(x)\}$, the low-quality image $t(x)$ represents an input to obtain a resulting image $G(x)$. This algorithm makes $G(x)$ as close to the original image $T(x)$ as possible, thus removing the noise and improving the image quality.

4.1. Network structure

The generator uses a known 'U-Net' architecture, as shown in Fig. 1. The input image $t(x)$ is encoded and then decoded into the image $G(x)$. U-Net uses a skip connection method to connect the i-th layer and the n-i-th layer, where n is the total number of layers. At the down-sampling end, the network takes a 256×256 image as an input and processes it in a down-sampling module. One of the down-sampling modules includes Conv2D and LeakyReLU activation layers (see Ref. 15). The first down-sampling module does not include a normalization layer, and the rest ones include the latter in order to prevent gradient explosion and gradient disappearance. After several down-sampling operations, the size becomes $1 \times 1 \times 512$. On the up-sampling side, there are multiple up-sampling layers. A single up-sampling module includes Conv2DTranspose and ReLU layers (see
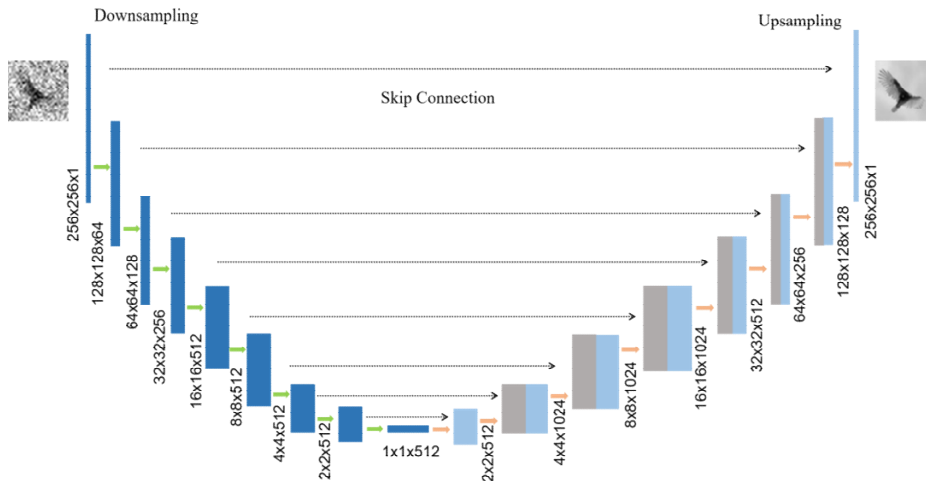
Fig. 1. Structure of generator network.

Ref. 16). Only the first three up-sampling modules include a Dropout [17] layer, with the dropout being equal to 0.5. The image sizes for the *i*-th and *n-i*-th layers are the same, and the information cutting and copying operations at the two ends of the linkage skip the connection. They implement a deep neural network layer that crosses the adjacent-layer nodes to perform information interaction. This can help the deep neural network to perform feature fusion for the multi-scale image features and obtain more accurate output results, thus improving the resolution. Finally, the output gives a predicted 256×256 image. The discriminator uses a "PatchGAN" classifier. In order to better judge a part of the image, the latter is divided into multiple fixed-size area blocks, and the true and false values of each area block are judged separately. The average value of the regional-block discrimination results is used as a final output value, as demonstrated in the discriminator part of Fig. 2.
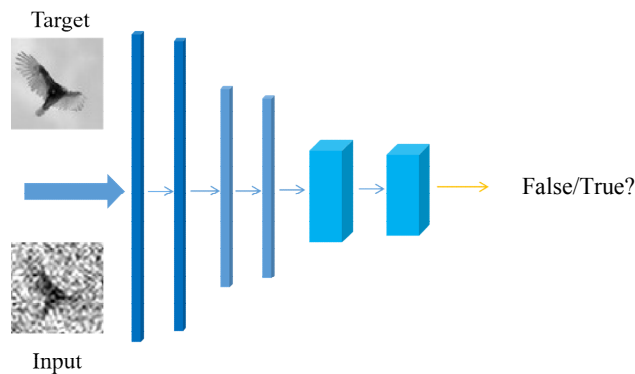

Fig. 2. Discriminator network structure.

First, the real image of the label and the image obtained by the generator are inputs into the discriminator at the same time. The appropriate size is 256×256, and the connection is merged. After three down-sampling operations with the step size 2, the image block with the dimension size 32×32×256 is obtained and the padding operation is carried out. After two-dimensional convolution processing and batch normalization, we obtain an input into the known Leaky ReLU activation function. Then it passes the padding operation to become a 33×33×512 image, and the two-dimensional convolution operation finally produces a 30×30 patch, to finally discriminate it.

4.2. Loss function

For network training, a reasonable objective loss function needs to be set. For any general CGAN, the adversarial loss function can be defined as

$$L_{CGAN}(G,D) = E_{x_c,y_c}\left[\log D(x_c,y_c)\right] + E_{x_c,z}\left[\log(1 - D(x_c, G(x_c,z)))\right], \tag{7}$$

where E denotes mathematical expectation, x_c the input GI image, y_c is the the 'Ground Truth', which means real image, z the random-noise distribution, G the information of the generator, and D the information of the discriminator.

Since the network is used to process standard image-mapping problems, our problem is transformed into a regression one. The related studies have shown that mixing the CGAN objective loss function with some traditional losses (such as those given by L_1 and L_2 distances) provides better results. Therefore, in order to make the low-quality images more similar to the target images, a regression loss function is needed to calculate the similarity. Compared with the L_2 loss, the L_1 loss has higher robustness. It is given by

$$L_{L1}(G) = E_{x_c,y_c,z}\left\| y - G(x_c,z) \right\|_1. \tag{8}$$

In the relevant comparative experiments, it has been deduced that the effect is better when the weight parameter of the L_1 loss function is $\lambda = 100$. Therefore, the objective loss function finally used in our network reads as

$$G^* = \arg\min_G \max_D L_{CGAN}(G,D) + \lambda L_{L1}(G), (\lambda = 100), \tag{9}$$

where 'argmin' and 'argmax' represent the variables at which the objective function takes a minimum and a maximum, respectively. When argmax over D and argmin over G are obtained, the loss function reaches a final convergence. Since the purpose of training the discriminator D is to make D as large as possible, its discriminative ability is better. At the same time, the loss function of the generator G should be extremely small, so that the distributions of the generated and real images become close to each other.

5. The essence of our method

Aiming at the low-light imaging, we note that the existence of a large amount of noise leads to degraded image quality. We first simulate the noise based on the Poisson distribution and construct an appropriate low-light imaging model. Then we use the two main steps to perform the image restoration: (i) to remove a large amount of noise in the image, transform it into the frequency domain through the DCT, and perform high-frequency shielding for removing some part of the noise and (ii) to process the image obtained in the first step. We input the trained condition to generate the adversarial neural network and estimate the closeness of the low-light image and the real image. The whole process is illustrated schematically in Fig. 3.
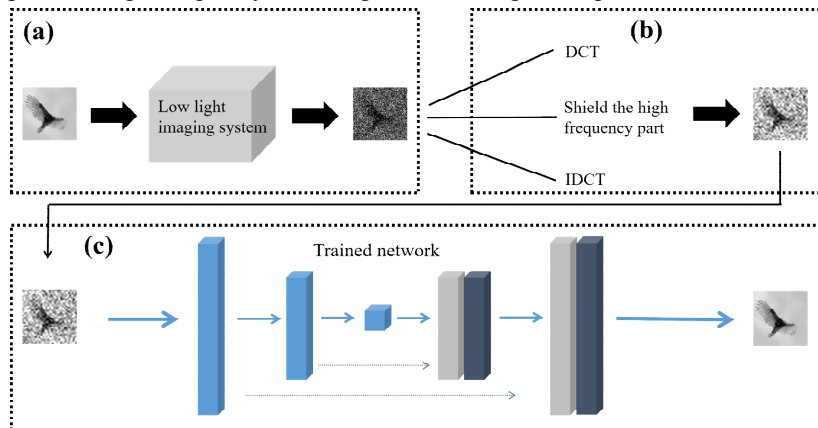

Fig. 3. Low-light image restoration process.

6. Simulation analysis

In this section we use simulation experiments to study the following: (1) the image quality under different detection SNRs, (2) low-light image denoising based on traditional methods, and (3) low-light image restoration based on the DCT and the CGAN combined together. When evaluating the imaging quality, we use an objective evaluation method for the quantitative analysis, with the peak SNR (PSNR) unit being dB. The larger its value, the smaller the distortion is. The corresponding formula is as follows:

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

Given a clean $m \times n$ image I and a noised image K, one can define the mean-square error MSE as

$$MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}\left\|I(i,j) - K(i,j)\right\|^2, \tag{11}$$

where MAX_I^2 is a maximal possible pixel value in the image. If a pixel value is represented as a binary n-bit quantity, then we have $MAX_I = 2^n - 1$. A structural similarity index measure (SSIM) evaluates a similarity of two images and can change in the range [0, 1]. The larger its value, the smaller the distortion is. The corresponding formulae read as

$$SSIM(x,y) = [l(x,y)^\alpha c(x,y)^\beta s(x,y)^\gamma], \tag{12}$$

$$l(x,y) = \frac{2\mu_x\mu_y + c_1}{\mu_x^2 + \mu_y^2 + c_1}, \quad c(x,y) = \frac{2\sigma_x\sigma_y + c_2}{\sigma_x^2 + \sigma_y^2 + c_2}, \quad s(x,y) = \frac{\sigma_{xy} + c_3}{\sigma_x\sigma_y + c_3}, \tag{13}$$

where $l(x,y)$ denotes the luminance comparison, $c(x,y)$ the contrast comparison, $s(x,y)$ the structural comparison, μ_x and μ_y are the mean values of respectively x and y, and σ_x and σ_y the standard deviations of respectively x and y. Here σ_{xy} represents the covariance of x and y, and c_1, c_2 and c_3 are small constants used to avoid a zero denominator. Finally, α, β, γ in Eq. (12) imply the weights of different evaluation factors.

6.1. Imaging under different detection SNRs

In this section we study the performance of a low-light imaging system and its imaging quality under different detection SNRs. The influence of the detection SNR on the imaging quality can be quantitatively analyzed in this way. Below, we select five different original pictures and perform imaging under the detection SNRs equal to 1, 3, 5, 7 and 10 dB. The appropriate results are shown in Fig. 4. One can see that the image quality is extremely poor when the detection SNR is only 1 dB. For example, the image of a Man under this condition has both outlines and details very blurred. It is almost impossible to identify a person and a background. With improving detection SNR, the quality of images becomes higher and higher (see Fig. 4 from left to right).

Fig. 5 shows the PSNR and SSIM curves obtained for the five image groups corresponding to different SNRs. It is evident from Fig. 5a that the SSIM values for the Bird, Frog, Horse and Man images are very close (~ 0.3) when the detection SNR amounts to 1 dB. With increasing SNR, the SSIM reveals an upward trend and reaches 0.9 at 10 dB. The SSIM values for all of the Hawk images remain rather low under the same detection SNRs (about 0.1 lower on average, if compared with the other four images).
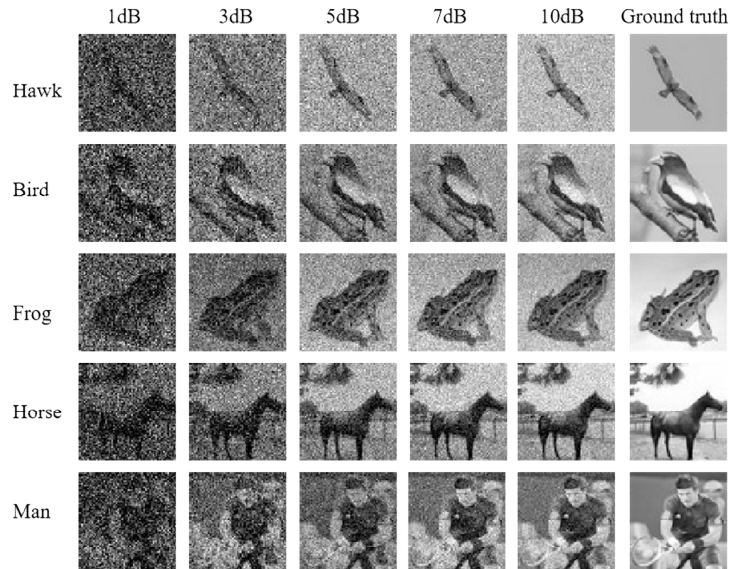
Fig. 4. Five sample images observed at different detection SNRs shown in the legend.

The PSNR values calculated for different images are quite different under the same detection SNRs (see Fig. 5b). This is because different types of the images have different information and so different final quality. When the detection SNR for the same group of images gradually increases, the PSNR also increases. When the detection SNR becomes 10 dB, the PSNR value improves by about 14 dB on average, if compared to the initial case of 1 dB. Fig. 5 testifies a great effect of the detection SNR upon the image quality, especially in case of the lowest level, 1 dB.
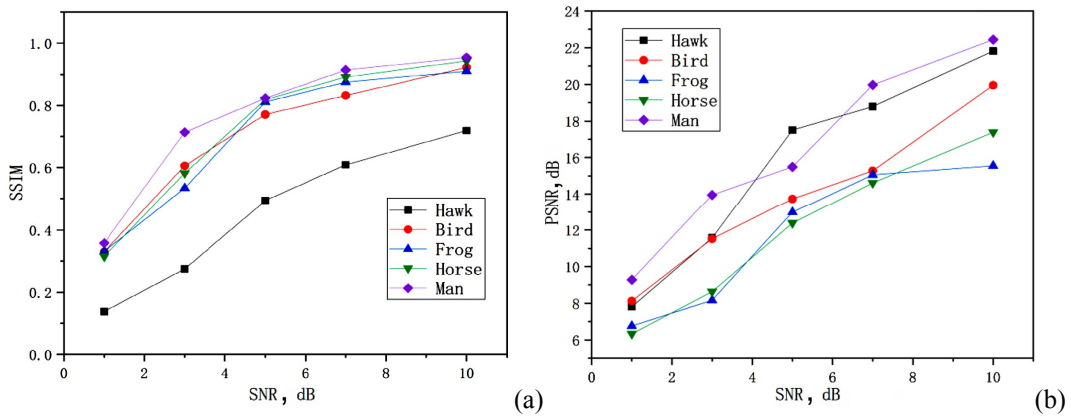

**Fig. 5.** SSIM (a) and PSNR (b) curves calculated for the five groups of images shown in Fig. 4.

6.2. Image-denoising analysis based on DCT

It is well known that the discrete Fourier transform is a complex and low-speed operation. Although the fast Fourier transform can increase the operation speed, it is rather inconvenient in image coding, especially in real-time processing. These shortcomings can be removed when using the DCT. In addition to general orthogonal-transformation properties of the DCT, the basis vector of its transformation matrix is very similar to the eigenvector of Toeplitz matrix, which embodies relevant characteristics of human language and image signals. In this section, we first use the DCT denoising method to process a part of noise. This is done in three steps:

(1) Perform the DCT transformation on the image to be processed and convert it into the frequency domain. Note that panels (b) and (e) in Fig. 6 show respectively a low-light image to be processed and a corresponding DCT-coefficient image;

(2) Process the coefficients in the frequency domain. Panels (a) and (d) in Fig. 6 correspond respectively to the original image and its coefficient map in the frequency domain. It is obvious that most of the energy in the original image is concentrated in the upper left corner, which means that the image information corresponds mainly to the low-frequency region. For the image that contains a lot of noise (see Fig. 6b), the energy distribution is rather messy (see Fig. 6e) and some energy is located in the high-frequency region. Therefore a high-frequency shielding is required.

(3) The coefficient diagram after high-frequency shielding is shown in Fig. 6f. Here the low-frequency part is retained. After DCT, a processed low-light image can be obtained (see Fig. 6c).
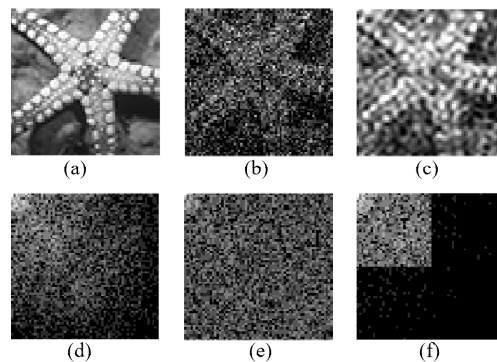
Fig. 6. Schematic diagram of DCT denoising algorithm (see explanations in the text).

Below we analyze the denoising effects of different traditional methods. The object under test is again detection of low-light images with the low SNR, 1 dB. We will compare the DCT denoising with the three standard image-denoising algorithms denoted commonly as Statisticsfilt, Wnfilt and Meanfilt. These algorithms have a sufficient denoising effect and their running time is about 1 s. Notice that the non-local mean denoising algorithm cannot remove a serious Poisson noise. Moreover, it also loses some original information so that its results are not good. The three-dimensional block-matched filtering algorithm and the other related algorithms do not result in outstanding denoising effect. Moreover, the appropriate calculations are cumbersome and the time cost is high. Therefore these methods are not suitable for pre-processing undertaken prior to deep learning. For this reason we perform no further comparison for these methods.

As seen from Fig. 7, Statisticsfilt, Wnfilt and Meanfilt reveal some denoising and filtering capabilities. Obviously, the processed images become smoother, though they still remain somewhat blurry. Statisticsfilt is slightly better than Wnfilt and Meanfilt. Among all of the methods, the DCT denoising algorithm manifests the best performance. It is seen from Fig. 7 that this method removes more Poisson noise, while the quality of the processed images is significantly improved when compared to the initial 1 dB image.

Comparing Fig. 8a with Fig. 8b, one concludes that the SSIM for the Hawk map after denoising by each method is about 0.3 and the corresponding results are rather close to each other. In addition, the DCT has obvious advantages in denoising performance, being followed by the Statisticsfilt method. The Wnfilt and Meanfilt methods reveal a similar denoising effect. When compared with the 1 dB image, the SSIM value for the image denoised by the DCT method increases by more than 100% and the corresponding PSNR value increases by more than 90%.
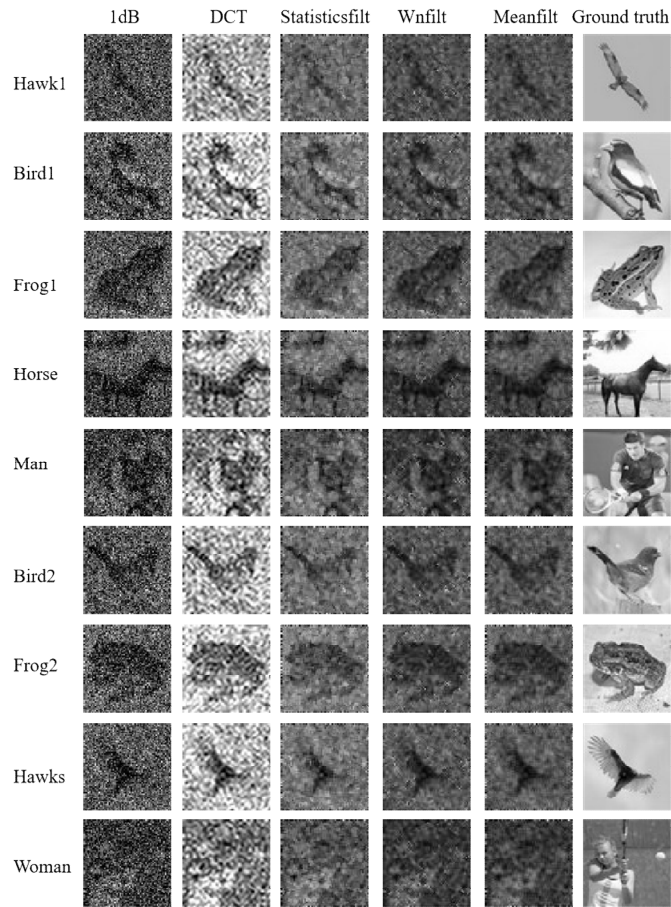
Fig. 7. Comparison of different denoising algorithms applied after pre-processing.
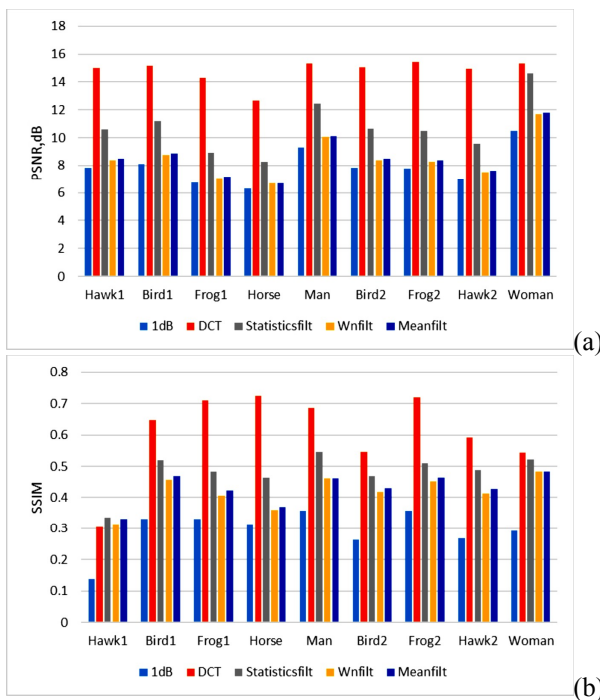

(a)

(b)

Fig. 8. Histograms of PSNR (a) and SSIM (b) values calculated for the images of Fig. 7 after denoising performed by different methods (see the legend).

6.3. Restoration based on DCT and CGAN

Now we discuss the CGAN network model built under the framework of Tensorflow 2.0 [18], which runs on a Python 3.7 platform. The training adopts an alternate training method of generator and discriminator. Further on, we use a known MSRA data set [19] as a basis. In order to construct the data set, we have selected 5000 images from MSRA. These original images are taken as $\{T(x)\}$. As above, the low-quality image data set $\{t_1(x)\}$ is generated one-to-one through a low-light imaging. The image-detection SNR used in our simulations is again equal to 1dB. The processed data set $\{t_2(x)\}$ is obtained through the DCT denoising processing. We have obtained a data set $\{t_2(x), T(x)\}$ composed of 5000 pairs of images. Finally, 3000 image pairs have been selected as a training set, 1000 pairs as a verification set and 1000 pairs as a test set.

In the training process, we have employed the batch size 32 and the buffer size 400. A known Adam's optimization method has been used to optimize all the networks in the generator, with the batch size set to 4 and the initial learning rate set to 0.0002. When testing, only the generation network is needed. Therefore, we input directly the low-quality images into the generation network for calculations and then obtain the restored images.
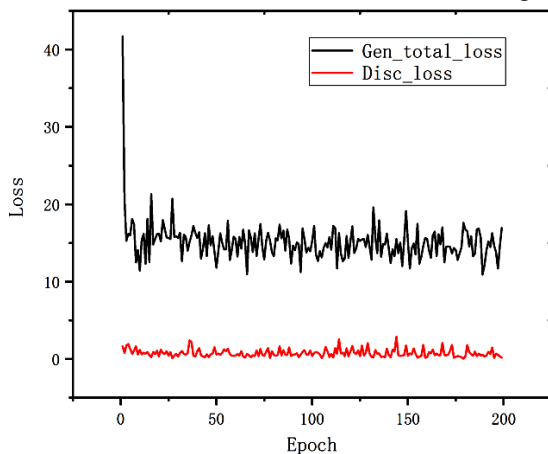
Fig. 9. Behaviour of loss function value observed during training. The upper and lower curves correspond respectively to generator and discriminator loss functions.

Fig. 9 displays a change in the loss functions occurring with increasing number of training rounds during the training. Our simulation experiments demonstrate that the test effect becomes fairly good whenever the number of training rounds reaches ~ 200. Therefore we set this number to be 200 further on. It is seen from Fig. 9 that the initial value of the generator loss function is larger because there is a large gap between the input image and the target original image. As the number of trainings increases, the generator loss value oscillates slightly and decreases steadily, thus reaching gradually a convergence. The advantage of the loss function is a combination of adversarial and regression losses. The value of the discriminator loss function remains relatively stable, with only slight fluctuations, and it finally converges to the region [0, 1].

The main experimental results are shown in Fig. 10. Here we compare the 1 dB image, the image processed using the DCT denoising procedure alone and the restored image processed by the combination of the DCT denoising and the CGAN network, as suggested in our study. It is evident that the image quality at 1 dB is extremely poor and most of the information is submerged by the noise. After the DCT denoising algorithm is applied, some information of the target image can be seen and its main content can be identified, although this result is far from satisfactory. To further restore it, we employ the CGAN-network method. The results confirm that our combined method can achieve good enough results. The restored images are very close to the original ones,

whereas the noise is almost completely removed. Nonetheless, there are some details that have not been fully restored (e.g., the legs of a frog and the wings of an eagle – see the lowest row of images in Fig. 10).
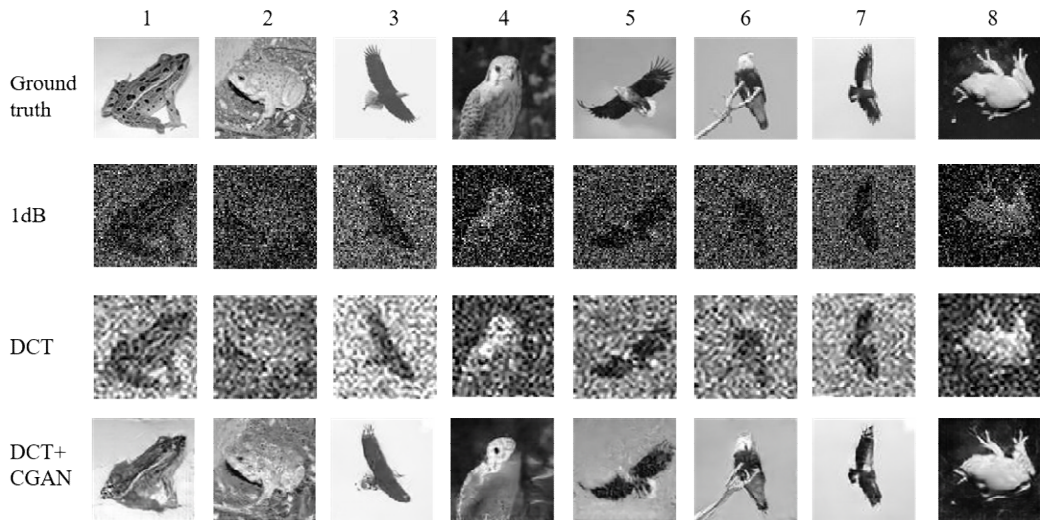

Fig. 10. Images restored by different methods referred to in the legend.
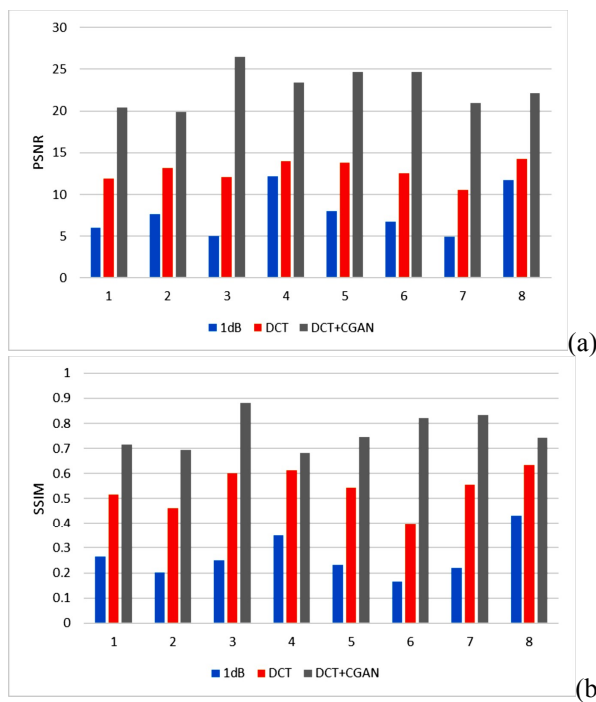

(a)

(b)

Fig. 11. Histograms of PSNR (a) and SSIM (b) values calculated for the images processed and restored by different methods (see the legend).

Finally, we evaluate the imaging results, using the PSNR and SSIM histograms (see Fig. 11). It is evident that the images obtained after DCT denoising has higher PSNR and SSIM values. The improvement is nearly equal to 100%. Table 1 shows the average PSNR and SSIM values derived for different processing and restoration methods. The combination DCT+CGAN improves greatly the restoration effect. When compared with the case of 1 dB image, the PSNR increases by about 15 dB and the SSIM by about 0.5. Comparison with the DCT-processed image demonstrates the increase in both the PSNR (about 8 dB) and the SSIM (~ 0.3). Due to our method, the average

PSNR value of the final image is as high as ~ 22.8 dB and the SSIM value is ~ 0.8. This implies that our method enables one to achieve a good enough processing and restoration effects.

Table 1. Average PSNR and SSIM values obtained with different processing and restoration methods

Method	1 dB	Statisticsfilt	Wnfilt	Meanfilt	DCT	DCT+CGAN
PSNR, dB	7.786	11.636	8.510	8.681	14.743	22.830
SSIM	0.265	0.481	0.427	0.441	0.573	0.796

7. Conclusion

It is a general knowledge that the images resulted from low-light imaging situations contain a lot of noise and their quality is extremely poor. In the present work we simulate low-light imaging basing on the Poisson-distributed noise. The lower the system-detection SNR, the worse the image quality is. We have concentrated on the image restoration performed under the condition of 1 dB SNR. We have suggested using a combination of the DCT denoising algorithm and the CGAN processing network. Taking into account the nature of the noise under the above conditions, we transform the image into the frequency domain by the DCT and perform the high-frequency shielding.

First, different denoising and preprocessing algorithms have been analyzed. It has been found that our method has a better performance and lower time cost, when compared with the commonly known denoising algorithms. Then we use the CGAN algorithm to improve further the image quality. As a result, our method restores the images with greatly improved quality. In particular, the average PSNR and SSIM parameters characterizing our method are respectively over 22 dB and about 0.8.

Disclosures. The authors declare no conflicts of interest.

References

1. Remez T, Litany O, Giryes R and Bronstein A M, 2018. Class-aware fully-convolutional Gaussian and Poisson denoising. IEEE Trans. Imag. Proc. **27**: 5707−5722.
2. Marnissi Y, Zheng Y, Chouzenoux E and Pesquet J-C, 2016. A variational Bayesian approach for image restoration: application to image deblurring with Poisson-Gaussian noise. IEEE Trans. Comp. Imag. **3**: 722−737.
3. Sanjay G, Pravin N and Narayan C K, 2018. Optimized Fourier bilateral filtering. IEEE Signal Proc. Lett. **25**: 1.
4. Zhang X, 2017. Improved interscale-based SURE-LET approach to image denoising. Proc. Comp. Sci. **107**: 819−823.
5. Zachiu C, Ries M, Moonen C and Baudouin Denis de Senneville, 2017. An adaptive non-local-means filter for real-time MR-thermometry. IEEE Trans. Med. Imag. **36**: 904−916.
6. Dabov K, Foi A, Katkovnik V and Egiazarian K, 2007. Image denoising by sparse 3-D transform-domain collaborative filtering. IEEE Trans. Imag. Proc. **16**: 2080−2095.

7. Chao Dong, Chen Change Loy, Kaiming He and Xiaoou Tang, 2014. Learning a deep convolutional network for image super-resolution. Springer International Publishing. Computer Vision – ECCV, pp. 184−199.

8. Burger H C, Schuler C J and Harmeling S, 2012. Image denoising: can plain neural networks compete with BM3D? 2012 IEEE Conference on Computer Vision and Pattern Recognition, pp. 2392−2399.

9. Kai Z, Zuo W, Chen Y and Zhang L, 2016. Beyond a Gaussian denoiser: residual learning of deep CNN for image denoising. IEEE Trans. Imag. Proc. **26**: 3142−3155.

10. Jia X, Liu S, Feng X and Zhang L, 2019. FOCNet: a fractional optimal control network for image denoising. Proc. of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6054−6063.

11. Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang and Ling Shao, 2020. CycleISP: real image restoration via improved data synthesis. Proc. of the 33rd IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2020), pp. 2693−2702.

12. Mirza M and Osindero S, 2014. Conditional generative adversarial nets. Comp. Sci. **2014**: 2672−2680.

13. Zhang Leihong, Bian Zhixiang, Ye Hualong, Wang Zhaorui, Wang Kaimin and Zhang Dawei, 2021. Restoration of single pixel imaging in atmospheric turbulence by Fourier filter and CGAN. Appl. Phys. B. **127**: 45.

14. Isola P, Zhu J Y, Zhou T and Efros A, 2017. Image-to-image translation with conditional adversarial networks. IEEE Conference on Computer Vision & Pattern Recognition, pp.5967−5976.

15. Liu Yisi, Wang Xiaojun, Wang Lei and Liu Dongliang, 2019. A modified leaky ReLU scheme (MLRS) for topology optimization with multiple materials. Appl. Math. Comp. **352**: 188−204.

16. Dahl G E, Sainath T N and Hinton G E, 2013. Improving deep neural networks for LVCSR using rectified linear units and dropout. 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 8609−8613.

17. Srivastava N, Hinton G, Krizhevsky A, Sutskever I and Salakhutdinov R, 2014. Dropout: a simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. **15**: 1929−1958.

18. Rami Al-Rfou et al., 2016. Theano: a Python framework for fast computation of mathematical expressions, The Theano development team. arXiv:1605.02688 [cs.SC].

19. Mian Muhammad Sadiq Fareed, Qi Chun, Gulnaz Ahmed, Adil Murtaza, Muhammad Rizwan Asif and Muhammad Zeeshan Fareed, 2019. Salient region detection through salient and non-salient dictionaries. PLoS ONE. **14**: e0213433.

Анотація. *У процесі формування зображень при слабкому освітленні деяка частина корисної інформації зображення перекривається шумом. Коли перешкоди значні, відношення сигнал/шум (ВСШ), виявлене в системі, зменшується до дуже низького рівня. Ми вивчаємо зображення при слабкому освітленні за умови, коли ВСШ детектування дорівнює 1 дБ. Беручи до уваги, що шум часто знаходиться у високочастотній частині спектра, ми використовуємо дискретне косинусне перетворення (ДКП), щоб видалити шум*

або, принаймні, відфільтрувати його частину. Потім використано алгоритм умовної генеративної змагальної мережі (УГЗМ) для подальшого поліпшення якості зображення. Результати моделювання засвідчують, що поєднання алгоритмів ДКП і УГЗМ значно поліпшує результати відновлення та кінцеву якість зображень. Остання стає досить високою, так що середнє пікове ВСШ перевищує 22 дБ, а показник структурної подібності становить приблизно 0,8.