Learning point and contextual processing networks for low-light image enhancement

Zheng, Bowen

2021

Zheng, B. (2021). Learning point and contextual processing networks for low-light image enhancement. Master's thesis, Nanyang Technological University, Singapore.
<https://hdl.handle.net/10356/152819>

<https://hdl.handle.net/10356/152819>

<https://doi.org/10.32657/10356/152819>

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License (CC BY-NC 4.0).

Downloaded on 26 Apr 2023 20:26:09 SGT

LEARNING POINT AND CONTEXTUAL PROCESSING NETWORKS FOR LOW-LIGHT IMAGE ENHANCEMENT

ZHENG BOWEN

School of Electrical & Electronic Engineering

A thesis submitted to the Nanyang Technological University
in partial fulfillment of the requirement for the degree of
Master of Engineering

2021 JUNE

Statement of Originality

I hereby certify that the work embodied in this thesis is the result of original research, is free of plagiarised materials, and has not been submitted for a higher degree to any other University or Institution.

25-06-2021

ZHENG BOWEN

.....
Date

.....
ZHENG BOWEN

Supervisor Declaration Statement

I have reviewed the content and presentation style of this thesis and declare it is free of plagiarism and of sufficient grammatical clarity to be examined. To the best of my knowledge, the research and writing are those of the candidate except as acknowledged in the Author Attribution Statement. I confirm that the investigations were conducted in accord with the ethics policies and integrity standards of Nanyang Technological University and that the research data are presented honestly and without prejudice.

25-06-2021

.....
Date

Jiang Xudong

.....
JIANG XUDONG

Authorship Attribution Statement

This thesis **does not** contain any materials from papers published in peer-reviewed journals or from papers accepted at conferences in which I am listed as an author.

25-06-2021

.....
Date

ZHENG BOWEN

.....
ZHENG BOWEN

Acknowledgments

First of all, I am very grateful to my mentor, Prof. Jiang Xudong for his careful guidance of my graduation thesis, which greatly improved my understanding of academic writing and taught me a lot of specific research skills. Besides, I would like to offer my thanks to Dr. Li Hanhui who gives me many guidance on the experimental skills and innovation ideas. Finally, I will say thanks to my friends who give me support and help in other ways.

Table of Contents

Summary	I
List of Figures	II
List of Tables	IV
1 Introduction	1
1.1 Background and Motivation	1
1.2 Major contributions of the Thesis	3
1.3 Organization of the Thesis	4
2 Literature Review	5
2.1 Introduction	5
2.2 Traditional Enhancement Methods	5
2.3 Deep Learning-based Methods	6
2.4 Conclusion	7
3 Methodology	8
3.1 Introduction	8
3.2 Point processing techniques	8
3.3 Rethinking Convolutional Neural Networks and Point Processing	10
3.4 Point and Contextual Processing Network	10
3.4.1 Dual-branch Architecture	11
3.4.2 Point Processing Branch	12
3.4.3 Contextual Processing Branch	13
3.4.4 Loss Functions	14
3.5 Implementation Details	16
4 Discussion about Preliminary Results and Ablation Study	17

TABLE OF CONTENTS

4.1	Introduction	17
4.2	Preliminary Experiments on Color space	17
4.2.1	Discussion of preliminary results	18
4.3	Preliminary Experiments on Attention Module	20
4.3.1	Discussion of preliminary results	21
4.4	Preliminary Experiments on Total Network Architecture	22
4.4.1	Discussion of preliminary results	23
4.5	Preliminary Experiments on Hidden intensity images of PPB	24
4.5.1	Discussion of preliminary results	24
4.6	Preliminary Experiments on Point-Convolution in PPB	26
4.6.1	Discussion of preliminary results	26
4.7	Ablation study	27
5	Results of Experiments	29
5.1	Benchmark Evaluations	29
5.2	Quantitative and Qualitative Comparisons	29
5.3	Adjustable Outputs	31
6	Conclusion and Recommendations	37
6.1	Conclusion	37
6.2	Recommendations for Further Research	37
6.2.1	Improvement of the existing work	37
6.2.2	Extension of the existing work	38
	References	41

Summary

Point processing techniques, such as the gamma correction, are classical methods for low-light image enhancement. These methods are efficient, explicit, and interpretable. However, to tackle images of various light conditions, these methods need to be set meticulously and accordingly. To tackle this issue, I propose to combine these traditional methods with the powerful learning ability of convolutional neural networks (CNNs). Specifically, I propose a point and contextual processing network (PCPNet) consisting of two parallel branches: In the point processing branch, given an input image, a set of hidden intensities images (HIs) are predicted, as well as a set of parameters governing the gamma corrections performing on the HIs. Due to the nonlinearity of the gamma corrections, we can obtain diverse and complex enhancement effects merely with a shallow network (of 3 layers). This further guarantees the efficiency of the point processing branch, and allows the HIs to be generated with the full resolution and preserve the details. While in the contextual processing branch, an encoder-decoder structure is adopted to explore contextual information, which helps to alleviate the effects of noise. The outputs of these two branches are combined as the final enhancement result. Extensive experiments on multiple datasets, including LOL, MIT, and ExDark, validate the effectiveness of the proposed method.

List of Figures

1.1	Visual performance of state-of-the-art methods on processing real scene low-light images.	2
3.1	Point processing techniques.	9
3.2	The dual-branch architecture of the proposed network. The upper branch is the point processing branch that estimates a set of hidden intensity images (X), and the parameters governing the gamma corrections (c and γ). The bottom contextual processing branch is an encoder-decoder CNN which leverages the contextual information to reduce noise.	11
3.3	Visual examples of the hidden intensity images. It is noted that each hidden intensity image has the high intensity values in particular areas. This indicates the point processing branch has learned to segment the input image and conduct transformations accordingly.	12
4.1	Visual examples of the processed images on different color space. Images processed in YCbCr and CIELAB color spaces have less problem of over-enhanced than those in RGB color space.	18
4.2	Visual examples of the processed images on different color space. Images processed in RGB color space have less problem of color distortion than those in YCbCr and CIELAB color spaces.	19
4.3	CBAM module adopted in the first two layers of contextual processing branch. Channel and spatial attention help the model focus on 'what' and 'where' is important.	21
4.4	A possible series link network structure.	22
4.5	Visual comparisons of the hidden intensity images with different D values. For each example, 4 selected hidden intensity images are presented for $D = 4, 8, 16$ from top to bottom. It is clearly from these images that, the larger D is, the more fine-grained segmentation-like results we can obtain.	25
4.6	Visual Comparison of the PCPNet with different components.	28
5.1	Visual comparison with state-of-the-art methods on the LOL dataset.	33

LIST OF FIGURES

5.2	Visual comparison with state-of-the-art methods on the MIT-Adobe FiveK dataset.	34
5.3	Visual comparison with state-of-the-art methods on the ExDark dataset.	35
5.4	Visual comparison of output with different θ	36
6.1	Four basic image types and their corresponding histograms. It shows that histogram equalization is effective on processing varies types of under-enhanced images.	39
6.2	Point Processing Branch with Histogram Equalization.	39
6.3	Indoor training set (ITS) and outdoor training set (OTS) from RESIDE dataset for image dehazing task.	40

List of Tables

4.1	Results of preliminary experiments on different color space. The experiment is performed via an encoder-decoder architecture with ResNet-18 as backbone similar to the proposed CPB.	18
4.2	Results of preliminary experiments on using different attention modules.	22
4.3	Results of preliminary experiments on two different network architecture.	23
4.4	Results of preliminary experiments on the number of hidden intensity images. . .	24
4.5	Preliminary study on the kernel size of convolutions in the point processing branch.	26
4.6	Ablation study on the contribution of each component in the PCPNet.	27
5.1	Quantitative comparison on the LOL dataset.	30
5.2	Quantitative comparison on the MIT-Adobe FiveK dataset.	30
5.3	Quantitative comparison on the ExDark dataset with BRISQUE as the metric (the lower the better).	31

Chapter 1

Introduction

1.1 Background and Motivation

There is no doubt that low-light environment is an indispensable part of our daily activities. When day turns to night or we turn off the indoor lights, the light intensity of the environment decreases which means less luminous flux to the camera. Hence, in such low-light photography condition, for single lens reflex (SLR) camera, people can use flash or extend the exposure time to increase the luminous flux to obtain images with high contrast. However, using flash will introduce unbalanced lighting and unexpected highlights into photos while extending the exposure time requires objects to remain still or motion blur will greatly reduce the image visual quality. As for digital camera, although people can directly set a larger ISO to increase the photo brightness, high ISO increases the image sensor sensitivity to light, and thus amplified the noise.

In this case, images captured in low-light conditions usually suffer from low contrast, poor visibility and unexpected noise. The suppressed details and low contrast will cause unpleasant subjective feelings; also, they seriously affect the performance of many computer vision tasks that highly rely on target visibility, such as object detection and scene understanding. Therefore, in order to improve the subjective feelings and usability of other vision tasks, low-light image enhancement techniques are necessary.

In the past decades, researchers have proposed various methods [1, 2] to solve the problem of low-light image enhancement. Nevertheless, low-light image enhancement is still a challenging task, since it needs to achieve the subtle balance between over-enhancement

and under-enhancement. On the one hand, under-enhancement images still have undesirable contrast and these images are more likely to fail to recover the details. On the other hand, over-enhanced results are usually accompanied with obvious noise and color distortion. Fig. 1.1 shows the results of several well-known methods. Obviously, some results suffer from visible color or noise artifacts while some results still have improper contrast.

Figure 1.1: Visual performance of state-of-the-art methods on processing real scene low-light images.

In this study, unlike most current methods directly learning an image-to-image mapping via deep neural networks, I propose to leverage the fundamental and classical point processing techniques [3]. Point processing is a bunch of explicable and effective transforms widely-used in traditional image enhancement. Take the gamma correction for an example, it can adjust an intensity image and generate various results by varying the gamma exponent. More importantly, the gamma correction is a **nonlinear transformation**, which means it has the larger potential for tackling images with complex illumination conditions, compared with the linear methods. In fact, many popular convolutional neural networks (CNNs), especially those based on the standard convolutions and the widely-used ReLUs, are limited in nonlinearity, and thus they alleviate this problem by stacking dozens or even hundreds of layers [4]. The

advantage of the point processing in nonlinearity makes it a considerable solution. However, to leverage this advantage, selecting the proper transformation parameters is necessary and sometimes frustrating.

Therefore, I propose to integrate the point processing into the deep learning framework, and the proposed network shall be able to learn parameters for region self-adaptive gamma corrections. Particularly, I devise a point and contextual processing network (PCPNet), which is composed of a point processing branch and a contextual processing branch. The roles of these two branches are as their names suggest: In the point processing branch, we construct a shallow network to estimate a set of hidden intensity images and the parameters for gamma corrections, so that I can conduct the self-adaptive gamma corrections on the hidden intensity images. Meanwhile, an encoder-decoder architecture leverages the strengths of CNNs in exploring contextual information. With these two branches, each pixel in the enhanced image is determined not only by the nonlinear transformations of the corresponding input pixel, but also by its context. Consequently, the proposed PCPNet can handle images of various light conditions with good visual performances. Last but not least, a luminance-guided loss function, which can be adaptive to the luminance distribution of the training images, is introduced to better optimize the PCPNet.

1.2 Major contributions of the Thesis

The contributions are summarized as follows:

- I propose a learning-based image point processing framework and integrate it with a contextual processing framework, to construct a dual-branch architecture for image enhancement.
- I realize the self-adaptive gamma corrections in the point processing framework, which achieves and combines various nonlinear transformations.
- I put forward a luminance-guided loss, which effectively alleviates the over-enhancement and color distortion problems.
- The proposed method is validated on multiple datasets and achieves the state-of-the-art performance.

1.3 Organization of the Thesis

The thesis is organized as follows:

1. Chapter 1 introduces background, motivation and major contributions of the research objectives.
2. Chapter 2 introduces the related work.
3. Chapter 3 shows the main theory of the proposed method and implementation details.
4. Chapter 4 discusses about the preliminary experiments and ablation study.
5. Chapter 5 shows the quantitative and qualitative comparisons.
6. Chapter 6 draws a conclusion for the thesis and introduces recommendations for further research.

Chapter 2

Literature Review

2.1 Introduction

In the past decades, a large number of low-light image enhancement schemes have been proposed. Generally speaking, as for low-light image enhancement task, they can be divided into traditional enhancement methods and deep learning-based methods.

2.2 Traditional Enhancement Methods

Traditional methods for low-light image enhancement can be mainly divided into two categories. The first category is built upon the histogram equalization (HE) techniques. For low-light images, directly amplifying the low-light part of image is the most intuitive way to improve the contrast of the entire image. By effectively spreading out the most frequent intensity values, HE allows areas of lower local contrast to gain a higher contrast. However, focusing on improving the overall contrast of image instead of considering the real illumination factors leads to the risk of over- and under-enhancement in some regions of images. Note that there are some variants further use some additional priors and constraints to improve the HE technique. In particular, CLAHE [5] tries to use multiple histograms to emphasize local contrast and limit contrast amplification to reduce noise amplification at the same time. BPDHE [6] tries to maintain the mean input brightness dynamically; CVC [7] tries to increase the gray value difference between each pixel and adjacent pixels in the image to improve the image contrast; LDR [8] seeks a layered difference representation of 2D histograms. Nevertheless, these methods de-

scribe histogram equalization on a grayscale image. Hence, the relative distributions of the color channels may change when applying them on color images.

The other category is based on the Retinex theory [9], which assumes that an image is composed of reflectance and illumination. Typically, single-scale Retinex (SSR) [10] and multi-scale Retinex (MSR) [11], treat the reflectance as the final enhanced result; NPE [12] tries to preserve naturalness for non-uniform illumination images. MF [13] proposes a fusion-based framework to produce an adjusted illumination and compensate the adjusted illumination back to the reflectance. SRIE [14] estimates reflectance and illumination simultaneously using a weighted variational model. LIME [15] refines the illumination map with structure prior. BIMEF [16] proposes a dual-exposure fusion framework to produce accurate lightness results and Ying *et al.* [17] use the camera response model for further enhancement. However, methods based on Retinex theory suffer from the limitation in model capacity of the decomposition for reflectance and illumination since actual human color perception has been shown to be more complex [18]. Besides, these methods generally assume that input images are noise- and color distortion-free, and do not explicitly consider the degradations, which may not be applied in many scenes.

2.3 Deep Learning-based Methods

With the rapid development of deep neural network, CNN has been widely used in low-level image processing, including denoising [19, 20], super-resolution [21], deblurring [22] *et al.* As for the low-light image enhancement task, many researchers are keen to use CNN to solve this problem. LLNet [23] uses a deep autoencoder for simultaneous contrast enhancement and denoising, but it is only tested on grayscale images. Shen *et al.* construct MSR-net [24] as an end-to-end model of mapping dark images to bright images. Retinex-Net [25] provides an end-to-end framework to combine the Retinex theory and deep networks. However, these strategies ignore the fact that the noise differently affects regions with various lights. Based on Retinex theory, KinD [26] designs two branches for handling the reflectance and illumination to flexibly adjust different light exposure conditions. Lv *et al.* [27] propose a multi-branch feature extraction and fusion structure (MBLLEN) to produce high quality images. As we can see from Fig. 1.1, both KinD and MBLLEN fail to produce images with correct brightness and color. Besides, Chen *et al.* [28] propose a pipeline for processing low-light images based on

end-to-end training of a fully convolutional network, which can jointly deal with noise and color distortion. Nevertheless, this work is specific to data in RAW format, limiting its applicable scenarios. GANs [29] have proven successful in image synthesis and translation. Based on GAN, EnlightenGAN [30] and Zero-DCE [31] can get the enhanced results without using any paired data in the training process. However, they cannot balance region noise removal and light exposure power well.

2.4 Conclusion

Although a large number of researchers have proposed many algorithms on this problem, their methods still have room for improvement. Limited by the lack of model expression ability, I do not choose traditional methods for low-light image enhancement. When it comes to deep learning-based methods, we can find that all these mentioned methods are trying to utilize the contextual information of a pixel to alter the input pixel value and expect to get a good prediction value. Nevertheless, this may blur some small image details and the nonlinear transform of the pixel value is inexplicit so that its nonlinear capacity of the pixel transform may be limited. In fact, the reason why CNN is effective in dealing with many computer vision problems is closely related to its strong nonlinear expression ability.

Therefore, a feasible scheme is to add nonlinear expression structure to CNN to make the whole neural network have more powerful nonlinear expression ability, so as to complete the end-to-end learning task better. Besides, contextual information should be avoided as far as possible when adding the nonlinear expression structure into the whole neural network.

Chapter 3

Methodology

3.1 Introduction

In this chapter, details of the proposed method are illustrated. Firstly, the basic theory of point processing techniques is introduced. Then, I will illustrate the reason that I choose to combine CNN with the point processing technique and introduce the details of the point and contextual processing network (PCPNet). Finally, the specific implementation details are illustrated.

3.2 Point processing techniques

Point processing means that the gray value of each pixel in the output image s , is only determined by the value of the corresponding pixel in the input image r . Therefore, point processing does not require the neighbouring information in the image, and it is widely used in image enhancement tasks due to its high efficacy and efficiency.

Image Negatives. By using negative transformations as $s = L - 1 - r$ of an image with gray levels in the range of $[0, L - 1]$, white or gray detail embedded in dark regions of an image is enhanced and thus more obvious to see. For example, when the black areas are dominant of the whole image such as digital mammogram, using negative transformations help doctors to analyze the breast issue.

Log Transformations. The general form of log transformations is defined as $s = c \log(1 + r)$,

Figure 3.1: Point processing techniques.

where c is a constant, and $r \geq 0$ is the predefined bias. The log curve in the Fig . 3.1 shows that a narrow range of input pixels with low grey-level values would be mapped into a wider range of output pixels. Besides, the opposite is true of the inverse log transformation. Such this characteristic makes log transformation be useful in compressing the dynamic range of pixel values [3].

Power-law transformations. The power-law transformation is also called gamma correction, of which the basic form is defined as,

$$s = cr^\gamma, \quad (3.1)$$

where c is a scale factor and γ is the parameter governing the gamma correction. Different effects can be achieved by varying the γ value. A gamma correction with $\gamma < 1$ is called *gamma compression* and with it the range of bright input values is narrowed down, while that with $\gamma > 1$ has exactly the opposite effect and is called *gamma expansion*. It is noted that the effect of gamma compression with $\gamma < 1$ is very similar to the log transformation. Thus, the

gamma correction is more versatile, and in this paper I select the gamma correction as the target point processing technique.

As the effect of gamma correction varies largely with different values of γ , a single γ parameter is not sufficient to handle images of various illumination conditions. Therefore, instead of learning a single γ for the whole image, I propose to learn multiple proper values of γ and apply them on different intensity images predicted from the input image.

3.3 Rethinking Convolutional Neural Networks and Point Processing

For most current CNNs, the main component is the convolution operation that applies linear filtering. As simple nonlinear activation functions, such as ReLU, are commonly used in CNN, its nonlinear capability in terms of processing pixel value is inexplicit and limited. In order to improve the nonlinear expression ability, CNNs need a large number of layers. This is a double-edged sword that, on the one hand, it largely increases the receptive field of convolution kernel and thus spatial information in a large area is utilized and condensed. In this way, CNNs are more biased to catch contextual and structure information of the input image. On the other hand, more convolution kernels are needed, and to maintain the computational costs, it is unavoidable for CNNs to utilize downsampling/pooling, which leads to the reduction of image resolution or removal of small image details.

In contrast, point processing techniques perform nonlinear transform of pixel value explicitly. This allows me to use a very shallow CNN (3 convolutional layers in this paper) to learn the parameters for point processing. Besides, the point processing branch takes less parameters than parameters taken in contextual processing branch of which CNN is used.

3.4 Point and Contextual Processing Network

In this section I will introduce the point and contextual processing network (PCPNet) in detail. Generally speaking, the proposed PCPNet is a dual-branch architecture with its two major components, namely, the point processing branch (PCB) and the contextual processing branch (CPB).

Figure 3.2: The dual-branch architecture of the proposed network. The upper branch is the point processing branch that estimates a set of hidden intensity images (X), and the parameters governing the gamma corrections (c and γ). The bottom contextual processing branch is an encoder-decoder CNN which leverages the contextual information to reduce noise.

3.4.1 Dual-branch Architecture

The goals of the PCPNet are (i) to provide the nonlinear gamma correction with self-adaptability, and (ii) to explore structural and contextual information for image enhancement. Hence, I build the PCPNet upon two parallel branches and realize each of goals via a specific branch. As shown in Fig. 3.2, given a low-light RGB image $I \in [0, 1]^{H \times W \times 3}$ as the input, the enhanced image I_E is generated by combining the output of the point processing branch $G(I)$, and the output of the contextual processing branch $C(I)$:

$$I_E = \sigma_1(G(I) + C(I)). \quad (3.2)$$

σ_1 is the modified ReLU for normalization, which can be formulated as,

$$\sigma_b(t) = \min(\max(t, 0), b), \quad (3.3)$$

and $b > 0$ is the upper bound.

The nonlinearity of the gamma correction allows $G(I)$ to have a smooth and high-dynamic range. Besides, due to the efficiency of the gamma correction, the point processing branch can be realized by a shallow architecture and run at the full resolution without any down-sampling operation. This guarantees that $G(I)$ can maintain the smallest details of input image. Mean-

Figure 3.3: Visual examples of the hidden intensity images. It is noted that each hidden intensity image has the high intensity values in particular areas. This indicates the point processing branch has learned to segment the input image and conduct transformations accordingly.

while, I consider to adopt a deep architecture with the large receptive field in the contextual processing branch, so that $C(I)$ can leverage contextual information to alleviate the effects of noise and outliers. With the joint optimization of these two branches, $G(I)$ and $C(I)$ complement each other, and consequently make the PCPNet a robust solution for low-light image enhancement.

3.4.2 Point Processing Branch

The key issue of realizing the self-adaptive gamma correction is to select the proper γ values for different regions and/or ranges of illumination level. To tackle this issue, I further divide it into three steps: Firstly, the input image needs to be split into parts of different region and/or range of illumination level; Secondly, for each split part, a corresponding γ value is learned and predicted to govern the gamma correction; Lastly, the corrected results of split parts are combined as the output of the self-adaptive gamma correction.

Our point processing branch is devised in accordance with the above scheme. Formally, given an input image $I \in [0, 1]^{H \times W \times 3}$, I define three transformations, F_D , F_c , and F_γ as follows:

$$X = F_D(I), c = F_c(X), \gamma = F_\gamma(X). \quad (3.4)$$

$X \in [0, 1]^{H \times W \times D}$ denotes D hidden intensity images (HIIIs) I will conduct the gamma corrections on, and D is the predefined number of the HIIs. $c \in [0, 1]^{D \times 3}$ and $\gamma \in [0, 10]^{D \times 3}$ are the

parameter matrices extending those of the traditional gamma correction defined in Eq. (3.1). The output of the point processing branch is obtained as:

$$G^i(I) = \frac{1}{D} \sum_{d=1}^D c_d^i X_d \gamma_d^i, \quad (3.5)$$

where $i \in \{R, G, B\}$ is the index for the color channels of the output image.

F_D serves as the dynamic segmentation of the input image, so that the non-zero pixels in each of the HIs can focus on certain parts of the input image. In this paper, F_D is implemented by two consecutive 3×3 convolutional layers, which have 128 and D output channels, respectively. Learning F_D with the PCPNet overcomes the trouble of setting fixed illumination thresholds for segmenting the input image. Fig. 3.3 demonstrates several examples of the HIs, from which we can clearly observe the different parts of the images captured by different HIs.

With the HIs, F_γ is introduced to estimate the exponents for gamma correction, while F_c is used to determine the weight of each enhanced intensity image. I implement F_γ and F_c with two modules of the same architecture, which is a 3×3 convolutional layer with D output channels, followed by a global average pooling layer and a 1×1 convolutional block with $D \times 3$ output channels. Note that I use σ_{10} as the last activation of F_γ , to ensure F_γ can simulate the effects of both gamma compression and gamma expansion, while σ_1 is used for F_D and F_c .

3.4.3 Contextual Processing Branch

The choices of the contextual processing branch are plenty because most modern CNNs are proficient in modeling contextual information. Therefore, I realize the contextual processing branch via an encoder-decoder architecture. The encoder is built upon the first four layers of ResNet-18 [32], which produces feature maps of size $\frac{H}{16} \times \frac{W}{16} \times 256$. I also augment the first two layers of the encoder with the attention modules [33]. The decoder part consists of four 3×3 convolutional layers. In each of these four layers, I first upsample the feature maps by factors of two, and then conduct the convolution. The first three convolutions are used to refine the features and their numbers of output channels are 64, while the last one predicts $C(I)$ directly. σ_1 is used as the last activation function of the contextual processing branch, to ensure the value of $C(I)$ is in $[0, 1]$.

3.4.4 Loss Functions

There is no doubt that the main purpose of the image enhancement task is that input and target images can be well matched densely (i.e., pixel-to-pixel). As for the low-light image enhancement task, noise amplification, color distortion, loss of details, boundary artifacts, and region light conditions are all need to be considered. Hence, loss function should be designed to solve these problems as far as possible. I construct the total loss function as below:

$$\mathcal{L} = \omega_{lg}\mathcal{L}_{lg} + \omega_{tv}\mathcal{L}_{tv} + \omega_{str}\mathcal{L}_{str}, \quad (3.6)$$

where the $\mathcal{L}_{lg}$, $\mathcal{L}_{tv}$ and $\mathcal{L}_{str}$ represent the loss function of a luminance-guided loss function, total variation loss and structure loss while ω_{lg} , ω_{tv} and ω_{str} are the corresponding coefficients. The details of those loss functions are given below.

Luminance-guided loss. The proposed luminance-guided loss function can adjust its weight according to the luminance distribution of the ground-truth image. Specifically, given an image I in the RGB color space, I first obtain its luminance as $I^L = \sum_i a^i I^i$ and chrominance component $I^{C,i} = (1 - a^i)I^i$, where $i \in \{R, G, B\}$ and the coefficients $a = [0.2126, 0.7152, 0.0722]^T$ ¹. Then, for the enhanced image I_E and the ground-truth image I_G , the luminance-guided loss $\mathcal{L}_{lg}$ is calculated as follows:

$$\mathcal{L}_{lg} = \frac{1}{H \times W} \sum_{p=1}^{H \times W} d^C(p) + (\phi \hat{I}(p) + 1) \cdot d^L(p), \quad (3.7)$$

where p is the index of pixels, $d^C(p) = \|I_E^C(p) - I_G^C(p)\|_1$ and $d^L(p) = \|I_E^L(p) - I_G^L(p)\|_1$ are the l_1 distances between each component of I_E and I_G , $\hat{I}(p) = I_G^L(p)/MAX(I_G^L)$ is the normalized luminance of I_G , and ϕ is a trade-off parameter.

The d^C term in Eq. (3.7) is used to prevent color distortion, while the d^L term encourages the network to mimic the illumination distribution of the ground truth. Considering that areas with larger luminance values are prone to be over-enhanced, I choose $\hat{I}(p)$ as the weight, so that this loss can emphasize brighter areas to alleviate over-enhancement.

Total variation loss. Total variation loss is widely used in denoising work [34, 35]. In order to remove noise and keep the spatial smoothness of the enhanced images, total variation loss is calculated in all three channels:

¹https://en.wikipedia.org/wiki/Relative_luminance

$$\mathcal{L}_{tv} = \frac{1}{CHW} \|\nabla_x I_E + \nabla_y I_E\|, \quad (3.8)$$

where C , H and W are the dimensions of enhanced image I_E , ∇_x and ∇_y represent the horizontal and vertical gradient operations respectively.

Structure Loss. In order to minimize the perceptual change in the structural content of output images, I use multi-scale structural similarity measure (MS-SSIM) [36] as the structure loss and calculate it in all three channels:

$$\mathcal{L}_{str} = 1 - \frac{1}{N} \sum_{p=1}^N \text{MS-SSIM}(I_E, I_G). \quad (3.9)$$

However, MS-SSIM is based on structural similarity measure (SSIM). Thus, we need to define SSIM first. Let us assume μ_{I_E} , $\sigma_{I_E}^2$ and $\sigma_{I_E I_G}$ are the mean of image I_E , the variance of I_E , and the covariance of image I_E and I_G , respectively. So,

$$\begin{aligned} \text{SSIM}(I_E, I_G) &= \frac{2\mu_{I_E}\mu_{I_G} + C_1}{\mu_{I_E}^2 + \mu_{I_G}^2 + C_1} \cdot \frac{2\sigma_{I_E I_G} + C_2}{\sigma_{I_E}^2 + \sigma_{I_G}^2 + C_2} \\ &= l(I_E, I_G) \cdot cs(I_E, I_G), \end{aligned} \quad (3.10)$$

and finally,

$$\text{MS-SSIM}(I_E, I_G) = [l_M(I_E, I_G)]^{\alpha_M} \cdot \prod_{j=1}^M [cs_j(I_E, I_G)]^{\gamma_j}, \quad (3.11)$$

where M is the number of scales, C_1 and C_2 in Eq. (3.10) are small constants. The first term in Eq. (3.11) compares the luminance of image I_E with its reference image I_G , and it is computed only at scale M . The second term measures the contrast and structural differences at various scales. The parameters α_M and γ_j are used to adjust the relative importance of each term. To simplify parameter selection, cross-scale settings are normalized such that $\alpha_M = \gamma_j = 1$ for $j = \{1, \dots, M\}$ [36].

3.5 Implementation Details

The PCPNet is implemented with PyTorch and runs on a GTX 1080 Ti graphic card. In all experiments, I use the LOL dataset [25] for training, which includes 485 low/normal-light real-case image pairs. I split the LOL dataset into 338 training pairs, 49 evaluation pairs and 98 test pairs. I resize each image to the size of 480×320 , and training the network by the Adam optimizer [37] with its default parameters. The network is trained for 360 epochs, with the learning rate of 0.001 and the batch size of 8. The whole training process takes about 6 hours.

The number of the hidden intensity images D is 64, and the weight of the normalized luminance ϕ in Eq. (3.7) is set to 0.1. The trade-offs in the total loss function in Eq. (3.6), ω_{lg} , ω_{tv} , and ω_{str} are set to 0.4, 0.04, and 1, respectively.

I adopt the peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) as the metrics for full-reference image quality assessment. As for no-reference image quality assessment, the widely used blind/referenceless image spatial quality evaluator (BRISQUE) [38] is adopted, as a holistic measure of image quality.

Chapter 4

Discussion about Preliminary Results and Ablation Study

4.1 Introduction

In this section, I introduce the preliminary experiments and ablation study with discussion about them. With the support of these experiments, the effectiveness of the proposed method is confirmed.

4.2 Preliminary Experiments on Color space

There is no doubt to say that color distortion will greatly affect the visual quality of an image. Therefore, in order to learn the characteristics of the input image data better and reduce the color distortion problem, it is necessary to select the appropriate color space. I conduct preliminary experiments on three different color spaces as RGB, CIELAB and YCbCr. Then I compare the enhancement results obtained in the RGB color space with those in other spaces, and demonstrate that the differences among the results are highly related to the gamma correction. In this way, we can better understand how to combine the gamma correction with deep neural networks and obtain desirable enhancement results.

Specifically, I construct a baseline network with the similar structure of the proposed CPB to perform enhancement in three color spaces, including RGB, CIELAB and YCbCr. To our

surprise, the baseline obtains the best performance in the RGB space, as reported in Table 4.1. Also, some visual results are shown in Fig . 4.1 and Fig . 4.2.

Color Space	PSNR	SSIM
CIELAB	20.93	0.7701
YCbCr	20.77	0.7688
RGB	21.11	0.7959

Table 4.1: Results of preliminary experiments on different color space. The experiment is performed via an encoder-decoder architecture with ResNet-18 as backbone similar to the proposed CPB.

Figure 4.1: Visual examples of the processed images on different color space. Images processed in YCbCr and CIELAB color spaces have less problem of over-enhanced than those in RGB color space.

4.2.1 Discussion of preliminary results

From the Fig . 4.1 and Fig . 4.2, we can easily to see that images processed in CIELAB and YCbCr color spaces are less likely to be over-enhanced while images processed in RGB color space have less color distortion. As the input channels of the baseline are normalized and processed jointly, I believe the performance gap is mainly caused by the color conversion.

Figure 4.2: Visual examples of the processed images on different color space. Images processed in RGB color space have less problem of color distortion than those in YCbCr and CIELAB color spaces.

I further notice that both CIELAB and YCbCr are involved with the gamma correction. Take CIELAB as an example, to covert a RGB image into the CIELAB space, we will first represent the image via the CIEXYZ coordinate system, and then calculate its $L^* a^* b^*$ values as follows¹:

$$\begin{aligned} L^* &= 116f\left(\frac{Y}{Y_n}\right) - 16, \\ a^* &= 500\left(f\left(\frac{X}{X_n}\right) - f\left(\frac{Y}{Y_n}\right)\right), \\ b^* &= 200\left(f\left(\frac{Y}{Y_n}\right) - f\left(\frac{Z}{Z_n}\right)\right), \end{aligned} \quad (4.1)$$

where X_n , Y_n and Z_n are empirical normalization parameters depending on the reference light source. f is a piecewise function defined as follows:

$$f(t) = \begin{cases} \sqrt[3]{t} & \text{if } t > \delta^3 \\ \frac{t}{3\delta^2} + \frac{4}{29} & \text{otherwise} \end{cases}, \quad (4.2)$$

where $\delta = \frac{6}{29}$. Since $t \in [0, 1]$ and $\delta^3 \approx 0.009$, the major part of f is actually a gamma correction with $\gamma = \frac{1}{3}$.

As shown in Fig. 3.1, the difference between two arbitrary intensity values is decreased with such a gamma correction. In other words, distances measured in the CIELAB space

¹https://en.wikipedia.org/wiki/CIELAB_color_space

will be smaller than those in the RGB space. This will suppress the gradient in networks whose optimization target includes certain distances, such as the L_2 loss. In our case, the gamma correction eventually narrows down the range of values in the enhanced images. The examples in Fig. 4.1 and Fig. 4.2 can support our opinion intuitively: the baseline tends to produce under-enhanced results in the CIELAB space, while over-enhanced results in the RGB space. Therefore, the experiment on different color spaces gives me support on combining gamma correction into the CNN structure.

4.3 Preliminary Experiments on Attention Module

Recently, there have been several attempts [33, 39, 40] to incorporate attention processing to improve the performance of CNNs in large-scale classification tasks. Attention mechanism helps the model focus more on the most informative feature expressions and simultaneously suppresses the less useful ones. EnlightenGAN [30] takes the luminance channel I of the input RGB images to generate attention map, and uses it to guide the model to learn different light condition regions. However, only using luminance channel attention map may lead to the color distortion of the results since information in chrominance channel is neglected. In our low-light image enhancement task, we hope that the attention module can guide the model to learn differences between dark and bright regions in the luminance and chrominance channels meanwhile.

In this case, the model needs to learn ‘how’ and ‘where’ to enhance the image. As shown in Fig. 4.3, CBAM module is adopted to complete this task since it can be divided into channel attention module and spatial attention module. As each channel of a feature map is considered as a feature detector [41], channel attention focuses on ‘what’ is meaningful to the model. Different with the channel attention, the spatial attention focuses on ‘where’ is important to the model.

To show the effectiveness of CBAM module, I compare it with some other widely used attention modules such as SE module and ECA module. Besides, I also test which layer or layers is most suitable to use attention module. The results are shown as Table 4.2.

Figure 4.3: CBAM module adopted in the first two layers of contextual processing branch. Channel and spatial attention help the model focus on 'what' and 'where' is important.

4.3.1 Discussion of preliminary results

Note that both SE and ECA modules only focus on channel attention where channel attention is actually redistributing the weight of numerous feature maps, but spatial attention adopted in CBAM module pushes the model to focus on the area difference of one feature map. That is the possible reason why using CBAM module gets better performance than others. Another concern is the reason why adopting CBAM module at the first two layers has the best performance. As the number of layers deepens, the encoder gradually extracts more feature maps and more complex contextual information. However, information will inevitably lose due to the operation of down-sampling in the encoder part. In this case, the first several layers hold richer information than the latter ones and they also use less running memory and parameters. We also need to know that we could not directly use attention module on the input image or the first layer of encoder. It is possibly because that the input image only has three

Module	PSNR	SSIM
SE	18.34	0.7222
ECA	19.00	0.7423
CBAM (layer0)	21.19	0.7938
CBAM (layer0, 1)	21.44	0.7972

Table 4.2: Results of preliminary experiments on using different attention modules.

channels (R, G, B) making it meaningless and problematic of using channel attention and the first layer may not have enough receptive field that limits the spatial attention to find “where” to focus on. In this case, adopting CBAM attention module at the first two layers is the best choice for my architecture.

4.4 Preliminary Experiments on Total Network Architecture

Though in Sec. 3.4, I plan to use dual-branch (parallel link) as the total network architecture. Notably, series link and parallel link are two possible ways for network architecture. Hence, in order to know which structure most suits our low-light enhancement task, I make some experiments to test it. As shown in Fig .4.4, in this experiment, I replace the input of the contextual processing branch with the output of the point processing branch as the series link network structure.

Figure 4.4: A possible series link network structure.

Similarly, given a low-light RGB image $I \in [0, 1]^{H \times W \times 3}$ as the input, $G(I)$ to be the output

of the point processing branch and $C(\cdot)$ to be the output of the contextual processing branch, thus, the output of the series link structure I_E^s could be expressed as:

$$I_E^s = \sigma_1[C(G(I))]. \quad (4.3)$$

The results are shown as below:

Architecture	PSNR	SSIM
Series Link	19.21	0.7137
Parallel Link	20.44	0.7972

Table 4.3: Results of preliminary experiments on two different network architecture.

As we can see from Table. 4.3, the parallel link structure has better performance than the series link structure.

4.4.1 Discussion of preliminary results

It is an expected result that parallel link structure performs better than series link structure in this low-light image enhancement task. Note that the proposed point processing branch is designed to learn multiple proper values of γ applied on different intensity images, thus parameters in PPB are quite important.

In the process of back propagation, since the output is the sum of $G(I)$ and $C(I)$ in parallel link structure, parameters in PPB and CPB are adjusted simultaneously. However, in the series link structure, parameters in CPB are adjusted first, and then back propagate to the PPB. In the PPB, c and γ are learnable parameters that directly govern the gamma correction performances so that $G(I)$ are quite sensitive to these parameters. Thus, the parallel link structure is better because parameters in PPB could be updated at the first stage of back propagation and then be able to make more subtle adjustments. Besides, since PPB is designed to realize the self-adaptive gamma correction, it is obvious to use origin low-light image as input since it has no loss of information. Differently, CPB is designed to realize the contextual information construction, thus it also needs to use the origin low-light image as input since it has most abundant and original contextual information. In this case, dual-branch (parallel link) structure is the most suitable architecture for this low-light image enhancement task.

4.5 Preliminary Experiments on Hidden intensity images of PPB

Considering the point processing branch (PPB) is the core of the proposed method, many experiments have been taken to find the best structure of it. As shown in Fig. 3.2, the key parameter is D , the number of hidden intensity images (HIIIs). In this case, we need to know if we can perform the deep gamma corrections directly on the RGB input image and if the larger D will get better performance.

	PSNR	SSIM	Parameters
CPB	20.29	0.7180	3.421M
w/ RGB	19.75	0.6364	28.67K
$D = 4$	20.04	0.7867	8.75K
$D = 8$	20.09	0.7894	14.55K
$D = 16$	20.22	0.7897	28.46K
$D = 32$	20.02	0.7817	65.50K
$D = 64$	20.44	0.7972	176.45K
$D = 96$	20.40	0.7912	336.54K
$D = 128$	20.77	0.7960	545.79K

Table 4.4: Results of preliminary experiments on the number of hidden intensity images.

4.5.1 Discussion of preliminary results

The effects of the number of hidden intensity images are evaluated, and the results are shown in 4.4. I set D in a wide range and evaluate the point processing branch on the LOL dataset. we can see that $D = 64$ achieves the best balance between performance and complexity. This is reasonable, because intensity images with a smaller D may be insufficient to cover the areas of diverse illumination conditions, while those with a larger D are redundant.

More specifically, with the HIIs, the spatial local structures/patterns in the input image can be separated, in the way that each of the HIIs has the high response to particular local structures/pattern. Consequently, though I just estimate a global gamma exponent for each HII, combining all HIIs together allow us to enhance the local areas in the input image adaptively. The visual results are shown in Fig . 4.5

To validate this, I also implement a variant of the point processing branch, in which we remove the first two 3×3 convolutional layers and conduct gamma corrections directly on the RGB input image (we may consider it as the situation of $D = 3$). Thus, I also set D to rather

Figure 4.5: Visual comparisons of the hidden intensity images with different D values. For each example, 4 selected hidden intensity images are presented for $D = 4, 8, 16$ from top to bottom. It is clearly from these images that, the larger D is, the more fine-grained segmentation-like results we can obtain.

small values, i.e., 4, 8, and 16 for comparison. The experimental results are also reported in Table. 4.4. It is obvious that even with a small value of D , e.g., $D = 4$, the performance of the point processing branch is significantly higher than that with gamma corrections conducted on

the RGB input image (SSIM: 0.7867 vs. 0.6364). Therefore, the conclusion is that the HIs are necessary for the performance of the point processing branch.

Table . 4.4 also includes the contextual processing branch (CPB) as a baseline for comparison. It is noted that the point processing branch with $D = 4$ is still comparable to the contextual processing branch (PSNR: 20.04 vs. 20.29, SSIM: 0.7867 vs. 0.7180), and the former has 8.75K learnable parameters only. This again demonstrates the notable advantages of nonlinear transformations in the low-light image enhancement. In Fig . 4.5, these examples clearly demonstrate that the larger value of D will help us obtain the HIs with more fine-grained segmentation. For example, one of the HIs (row 5, column 4) with $D = 8$ has the high response to both the floor and seats, while in that of $D = 16$ these two classes are separated well (row 6, column 3 and 4).

4.6 Preliminary Experiments on Point-Convolution in PPB

In the implementation of the point processing branch, 3×3 convolutions are used. As traditional point processing techniques conduct transformations with a single point/pixel, I wonder if it is feasible to replace all 3×3 convolutions with 1×1 convolutions. Therefore, I implement a variant of the point processing branch with 1×1 convolutions, and the results are reported in Table . 4.5. We can see that the SSIM of this variant is lower than that of the original one by a large margin (0.1355).

	PSNR	SSIM	Parameters
CPB	20.29	0.7180	3.421M
PPB w/ 1×1 Conv	19.60	0.6617	42.30K
PPB w/ 3×3 Conv	20.44	0.7972	176.4K

Table 4.5: Preliminary study on the kernel size of convolutions in the point processing branch.

4.6.1 Discussion of preliminary results

Note that the key difference between 1×1 convolution and 3×3 convolution is that, 1×1 convolution cannot extract local structures/patterns. In this case, it is hard for the HIs to segment the input image, and consequently the performance of the variant is hindered.

Therefore, 3×3 convolutions for the point processing branch is more preferable and this experiment also indicates the importance of the HLLs.

4.7 Ablation study

I first conduct the ablation study to investigate the effectiveness of each proposed component in the PCPNet, including the point processing branch (PPB), the contextual processing branch (CPB) and the luminance-guided loss (L_{lg}). The experimental results on the LOL dataset are reported in Table 4.6. First, I train each of the two branches as an individual network and replace the luminance-guided loss with l_1 loss. I find that the performance of the point processing branch (PSNR: 20.44, SSIM: 0.7972) is visibly higher than that of the contextual processing branch (PSNR: 20.29, SSIM: 0.7180). This demonstrates the advantages of nonlinear transformations in low-light image enhancement. Moreover, the number of parameters in the point processing branch (176.4K) is one order of magnitude smaller than that of the contextual processing branch (3.421M). Second, I apply the luminance-guided loss to these two branches, and we can observe the considerable performance gains for both of them. At last, integrating all three components obtains the highest performance (PSNR: 22.17, SSIM: 0.8192).

	PSNR	SSIM	Parameters
CPB	20.29	0.7180	3.421M
PPB	20.44	0.7972	176.4K
CPB+ L_{lg}	20.65	0.7285	3.421M
PPB+ L_{lg}	20.79	0.8028	176.4K
CPB+PPB	21.39	0.8000	3.598M
CPB+PPB+ L_{lg}	22.17	0.8192	3.598M

Table 4.6: Ablation study on the contribution of each component in the PCPNet.

I also visualize the enhancement results of these components in Fig. 4.6. We can see that, due to the downsampling/pooling layers in the contextual processing branch, it is unavoidable that some details cannot be recovered. On the other hand, this problem can be alleviated with the point processing branch because of its efficient architecture. Combining both branches provides more details and sharpen edges, but I also observe the color distortion in this case. Fortunately, with the luminance-guided loss, this problem is tackled and it obtains

the enhanced image that is highly similar to the ground truth.

Figure 4.6: Visual Comparison of the PCPNet with different components.

Chapter 5

Results of Experiments

5.1 Benchmark Evaluations

Benchmark Methods. I compare the PCPNet with 9 state-of-the-art methods, including MSR [11], Dong [42], NPE [12], LIME [15], BIMEF [16], Retinex-Net [25], MBLLN [27], KinD [26], and Zero-DCE [31]. The results are reproduced by using publicly available source codes, with recommended parameters and pre-trained weights.

Testing Datasets. I perform qualitative and quantitative analysis on three real scene low-light datasets, including the LOL dataset used in previous section, MIT-Adobe FiveK [43] with images resized to $1920 \times 1440 \times 3$, and ExDark [44] with images resized to $512 \times 512 \times 3$. Following previous methods [45, 46], on the MIT-Adobe FiveK dataset, I use the images edited by Expert C as the ground truth. And considering that only a small portion of images in this dataset are underexposed, I select 380 images from this dataset for evaluation. As for the ExDark dataset, since it does not provide the ground-truth images, I propose to conduct non-reference evaluation on it, with 1200 selected images (100 images from each of the 12 classes defined by the dataset).

5.2 Quantitative and Qualitative Comparisons

Table 5.1, 5.2, and 5.3 report the numerical results of the PCPNet against state-of-the-art methods on LOL, MIT-Adobe FiveK and ExDark datasets, respectively. The PCPNet outper-

forms all 9 competitors in terms of PSNR, SSIM and BRISQUE. These results suggest that the nonlinear point processing techniques are considerable if we integrate them with current deep learning frameworks properly.

Method	PSNR $\uparrow$	SSIM $\uparrow$
Dong [42]	15.57	0.4465
KinD [26]	19.77	0.7884
LIME [15]	14.36	0.4883
MBLLEN [27]	17.32	0.6892
MSR [11]	12.73	0.4207
NPE [12]	16.56	0.4507
RetinexNet [25]	15.62	0.3964
Zero-DCE [31]	14.87	0.5200
BIMEF [16]	14.83	0.5717
PCPNet	22.17	0.819

Table 5.1: Quantitative comparison on the LOL dataset.

Method	PSNR $\uparrow$	SSIM $\uparrow$
Dong [42]	13.82	0.6667
KinD [26]	15.90	0.7483
LIME [15]	12.91	0.6982
MBLLEN [27]	18.08	0.7517
MSR [11]	12.15	0.6810
NPE [12]	16.37	0.7360
RetinexNet [25]	12.13	0.6487
Zero-DCE [31]	15.40	0.7268
BIMEF [16]	17.06	0.7526
PCPNet	19.16	0.7646

Table 5.2: Quantitative comparison on the MIT-Adobe FiveK dataset.

Fig. 5.1, 5.2, and 5.3 demonstrate the enhancement results of the PCPNet on the three datasets. We can see from these results that, some of these competing methods cause severe artifacts in the enhanced images, such as the ringing effect and noise. As shown in Fig. 5.1, Dong, NPE and Zero-DCE cause obvious noise; MBLLEN, MSR, and RetinexNet suffer from color distortion; and results of BIMEF and LIME are under-enhanced. Other issues like over-enhancement can be observed from Fig. 5.2 and Fig. 5.3 as well. In Fig. 5.3, we can observe the severe color distortion or noise in the results of Dong, KinD, LIME, MBLLEN, NPE and RetinexNet. Especially the two images with cat inside, although the PCPNet balance noise

Method	BRISQUE↓
Dong [42]	37.88
KinD [26]	30.14
LIME [15]	39.69
MBLLEN [27]	25.57
MSR [11]	34.53
NPE [12]	36.06
RetinexNet [25]	39.28
Zero-DCE [31]	37.10
BIMEF [16]	34.44
PCPNet	24.75

Table 5.3: Quantitative comparison on the ExDark dataset with BRISQUE as the metric (the lower the better).

removal, color fidelity and contrast enhancement well, almost all the methods fail in completely noise removal. The possible reason is that these images suffer from severe noise problem. As for our PCPNet, since the $G(I)$ and $C(I)$ complement each other and the low-light image enhancement task requires to handle noise amplification, color distortion, loss of details and contrast problems at the same time, the main goal is to improve image contrast while solve other problems as much as possible. Hence, for a dataset with numerous images and a trained neural network model, it is normal that we cannot get excellent outputs with the inputs of some extreme images. Generally speaking, results of our PCPNet on these three datasets not only recover the details and color, but also suppress the noise and other artifacts.

5.3 Adjustable Outputs

Actually, Fig. 5.2 shows that the PCPNet still have color distortion problem when using MIT-Adobe FiveK as the test dataset. The possible reason is that the ground-truth images of this dataset are images edited by Expert C, which means that they are not real-scene images. In other words, these ground-truth images are generated by computer software with one certain style, and we can see that ground-truth images have stronger color than others. Differently, we use LOL dataset as the training set, whose both low-light and ground-truth images are real-scene images taken by camera. In this case, parameters learned by PCPNet may not perform well on MIT-Adobe FiveK dataset, causing the outputs do not have such strong color as the ground-truth images, but our outputs are much natural than ground-truth images.

Hence, in order to improve image quality and make the model be user-friendly, the adjustable output is necessary, which means that the output of an image can be adjusted manually. For example, some people may pay attention to the quality of the overall image and be strict with the overall exposure, while some people may pay attention to the quality of some specific details and do not demand overall image quality. In this case, I try to use a manually adjustable parameter θ to make the model be user-friendly. $0 < \theta \leq 1$ is designed to control the grey value of the input image. Specifically, given an input image I with its luminance as I^L and its chrominance as I^C , we can get a new input image I_n with its luminance as $I_n^L = \theta \cdot I^L$ and its chrominance as $I_n^C = I^C$. This is because most of the images in the training set are in extremely low light environment. Hence, θ helps lower the grey value of some input images whose environments are not extremely low-light so that the input images are more similar with the training set images, which is beneficial to the model prediction. Besides, since θ helps lower I^L , the chrominance component I^C is magnified relatively. Therefore, we can see from Fig. 5.4 that the lower θ , the stronger color of enhanced images. However, we still need to be careful that too low θ may cause color distortion and obvious noise problems.

Figure 5.1: Visual comparison with state-of-the-art methods on the LOL dataset.

Figure 5.2: Visual comparison with state-of-the-art methods on the MIT-Adobe FiveK dataset.

Figure 5.4: Visual comparison of output with different θ .

Chapter 6

Conclusion and Recommendations

6.1 Conclusion

I propose a point and contextual processing network for low-light image enhancement. The proposed network realizes the self-adaptive nonlinear gamma correction within its point processing branch, and leverages the contextual information via the contextual processing branch with the encoder-decoder architecture. With these two branches combined into a unified learning framework, I am able to generate various nonlinear transformations, and augment them with the contextual information to alleviate the effects of noise. The proposed network is trained with the proposed luminance-guided loss. Preliminary experiments show the superiority of the proposed structure. Ablation study has verified the effectiveness of the proposed three components. Benchmark evaluations on three datasets demonstrates the superior performance of the proposed PCPNet to existing methods.

6.2 Recommendations for Further Research

6.2.1 Improvement of the existing work

About the future work, one possible improvement is the dataset. Although the proposed PCPNet has performed quite well on three real-scene image datasets, note that only LOL dataset has real-scene ground truth while MIT Adobe FiveK uses Expert retouched as ground truth and ExDark does not have ground truth images.

Actually, we need to know that sometimes real-scene low and normal image pairs are quite difficult to obtain so that people may use synthetic images to meet the requirements of computer vision tasks for large-scale datasets. For example, by using software such as PhotoShop, we can manually adjust the contrast or noise level of an image. Then, we just need a normal light level photo to create numerous image pairs. Hence, for further research, test on datasets with synthetic low and normal light image pairs may be a new challenge.

Another way is to create a new real-scene low-light and normal-light image pairs dataset. However, I think it will be a heavy work to create a large dataset. As for the indoor scenes, we can control the light switch to get images in different light environments. Or we can imitate the method of LOL dataset to set different ISO of camera to get image pairs. As for the outdoor scenes, it seems that we can only use the method of LOL dataset. Or we can manually use devices such as flash LED or photography fill light to create normal light environment. But these images may be unnatural.

Besides, the proposed PCPNet only tested on images shot by cameras. Hence, we also need to test low-light images shot by cell phones. But still, we lack the corresponding dataset to make quantitative comparisons.

6.2.2 Extension of the existing work

Histogram equalization techniques. Firstly, still target at the low-light image enhancement task, the fact is that some low-light images are just low-contrast. As we all know from the gamma curves, for gamma smaller than one, dark regions are expanded and bright regions are suppressed while for gamma bigger than one, it has the opposite effects. Hence, gamma correction cannot handle images with gray values of pixels centered in the middle region.

However, for low-contrast image, histogram equalization can spread the histogram of the input image so that the levels of the histogram-equalized image will span a fuller range of the gray value. And this technique is also effective on dark or bright images. Hence, we can know that histogram equalization has stronger generalization ability.

However, Histogram Equalization uses the statistical information of the whole image since it needs to spread the input histogram to the full range. Thus, it can't make fine adjustments to local regions. One possible solution is that split the whole image into many different blocks and do the histogram equalization, respectively. Based on this, we can let CNN learn the proper

Figure 6.1: Four basic image types and their corresponding histograms. It shows that histogram equalization is effective on processing varies types of under-enhanced images.

area size. But this solution may cause severe block artifacts.

Now, let us refer to the structure of Point Processing Branch, and we know that the hidden intensity image are fine-grained segmentation-like results, then, we can try histogram equalization in this structure.

Figure 6.2: Point Processing Branch with Histogram Equalization.

Or more specifically speaking, as we can see from Figure .6.2 , we can do the histogram equalization on different hidden intensity images X and get learnable parameter γ to adjust the histogram equalization. It is because we cannot let the histogram-equalized image Y to span a fuller range of the gray value. So, γ is designed to ensure the processed images Y have proper bounds. Finally, we can get weighted summation as the output. Note that the region and parameters are all learnt by CNN.

By taking this strategy, pixels adopted the same HE are discontinuous. Hence, it will not

cause block artifacts.

Extension on other missions Another try is to adopt the network structure on other low-level missions. For example, image dehazing. Firstly, different with low-light image enhancement task, image dehazing has a large-scale dataset called RESIDE¹ with about four hundred thousand images. Besides, as we can see from Figure . 6.3, hazy images also suffer from low contrast problem. Thus, there are similarities between image dehazing task and our enhancement task, since the key point is to adjust the contrast of the image.

Figure 6.3: Indoor training set (ITS) and outdoor training set (OTS) from RESIDE dataset for image dehazing task.

Hence, as I just discussed about the network with Histogram equalization as replacement, we may also get good performance on image dehazing task with such a large-scale dataset training. But we still need to do further research and numerous experiments to test if this model is applicable in image dehazing task, and if severe color distortion happens, we may take these strategies: Firstly, try to test different manually adjustable parameter θ to see if it will help get good results. Secondly, try to adjust parameters ω in Eq. 3.6. As for the image dehazing tasks, the weights of retaining color fidelity, contrast enhancement and noise removal may have great differences with low-light image enhancement task. Thirdly, we may also try to analyze the differences between results and ground-truth images in both luminance chrominance channels. If the luminance channels are roughly the same, then it proves the model need to pay more attention on chrominance part, and we need to design new structure or loss function to emphasize the importance of chrominance part. If both channels are quite different, then it means the model may not work well on image dehazing tasks.

¹<https://sites.google.com/view/reside-dehaze-datasets>

References

- [1] W. Wang, X. Wu, X. Yuan, and Z. Gao, "An experiment-based review of low-light image enhancement methods," *IEEE Access*, vol. 8, pp. 87 884–87 917, 2020.
- [2] J. Liu, D. Xu, W. Yang, M. Fan, and H. Huang, "Benchmarking low-light image enhancement and beyond," *IJCV*, pp. 1–32, 2021.
- [3] R. C. Gonzalez and R. E. Woods, *Digital Image Processing, Fourth Edition*, 2018.
- [4] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *Proceedings of the European Conference on Computer Vision*, 2016, pp. 630–645.
- [5] S. M. Pizer, "Contrast-limited adaptive histogram equalization: Speed and effectiveness stephen m. pizer, r. eugene johnston, james p. ericksen, bonnie c. yankaskas, keith e. muller medical image display research group," in *Proceedings of the First Conference on Visualization in Biomedical Computing, Atlanta, Georgia*, vol. 337, 1990.
- [6] H. Ibrahim and N. S. P. Kong, "Brightness preserving dynamic histogram equalization for image contrast enhancement," *IEEE Transactions on Consumer Electronics*, vol. 53, no. 4, pp. 1752–1758, 2007.
- [7] T. Celik and T. Tjahjadi, "Contextual and variational contrast enhancement," *IEEE Transactions on Image Processing*, vol. 20, no. 12, pp. 3431–3441, 2011.
- [8] C. Lee, C. Lee, and C.-S. Kim, "Contrast enhancement based on layered difference representation of 2d histograms," *IEEE Transactions on Image Processing*, vol. 22, no. 12, pp. 5372–5384, 2013.
- [9] E. H. Land, "The retinex theory of color vision," *Scientific american*, vol. 237, no. 6, pp. 108–129, 1977.
- [10] D. J. Jobson, Z.-u. Rahman, and G. A. Woodell, "Properties and performance of a center/surround retinex," *IEEE Transactions on Image Processing*, vol. 6, no. 3, pp. 451–462, 1997.
- [11] D. J. Jobson, Z.-u. Rahman, and G. A. Woodell, "A multiscale retinex for bridging the gap between color images and the human observation of scenes," *IEEE Transactions on Image Processing*, vol. 6, no. 7, pp. 965–976, 2002.
- [12] S. Wang, J. Zheng, H.-M. Hu, and B. Li, "Naturalness preserved enhancement algorithm for non-uniform illumination images," *IEEE Transactions on Image Processing*, vol. 22, no. 9, pp. 3538–3548, 2013.

REFERENCES

- [13] X. Fu, D. Zeng, Y. Huang, Y. Liao, X. Ding, and J. Paisley, "A fusion-based enhancing method for weakly illuminated images," *Signal Processing*, vol. 129, pp. 82–96, 2016.
- [14] X. Fu, D. Zeng, Y. Huang, X.-P. Zhang, and X. Ding, "A weighted variational model for simultaneous reflectance and illumination estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2782–2790.
- [15] X. Guo, Y. Li, and H. Ling, "Lime: Low-light image enhancement via illumination map estimation," *IEEE Transactions on Image Processing*, vol. 26, no. 2, pp. 982–993, 2016.
- [16] Z. Ying, G. Li, and W. Gao, "A bio-inspired multi-exposure fusion framework for low-light image enhancement," *arXiv preprint arXiv:1711.00591*, 2017.
- [17] Z. Ying, G. Li, Y. Ren, R. Wang, and W. Wang, "A new low-light image enhancement algorithm using camera response model," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2017, pp. 3015–3022.
- [18] A. C. Hurlbert and C. J. Wolf, "Contribution of local and global cone-contrasts to color appearance: a retinex-like model," in *Human Vision and Electronic Imaging VII*, vol. 4662. International Society for Optics and Photonics, 2002, pp. 286–297.
- [19] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising," *IEEE Transactions on Image Processing*, vol. 26, no. 7, pp. 3142–3155, 2017.
- [20] S. Guo, Z. Yan, K. Zhang, W. Zuo, and L. Zhang, "Toward convolutional blind denoising of real photographs," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 1712–1722.
- [21] J. Kim, J. Kwon Lee, and K. Mu Lee, "Accurate image super-resolution using very deep convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1646–1654.
- [22] X. Tao, H. Gao, X. Shen, J. Wang, and J. Jia, "Scale-recurrent network for deep image deblurring," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8174–8182.
- [23] K. G. Lore, A. Akintayo, and S. Sarkar, "Llnet: A deep autoencoder approach to natural low-light image enhancement," *Pattern Recognition*, vol. 61, pp. 650–662, 2017.
- [24] L. Shen, Z. Yue, F. Feng, Q. Chen, S. Liu, and J. Ma, "Msr-net: Low-light image enhancement using deep convolutional network," *arXiv preprint arXiv:1711.02488*, 2017.
- [25] C. Wei, W. Wang, W. Yang, and J. Liu, "Deep retinex decomposition for low-light enhancement," in *The British Machine Vision Conference*, 2018.
- [26] Y. Zhang, J. Zhang, and X. Guo, "Kindling the darkness: A practical low-light image enhancer," in *Proceedings of the ACM International Conference on Multimedia*, 2019, pp. 1632–1640.
- [27] F. Lv, F. Lu, J. Wu, and C. Lim, "Mblen: Low-light image/video enhancement using cnns." in *The British Machine Vision Conference*, 2018, p. 220.

REFERENCES

- [28] C. Chen, Q. Chen, J. Xu, and V. Koltun, "Learning to see in the dark," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 3291–3300.
- [29] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *Advances in neural information processing systems*, vol. 27, pp. 2672–2680, 2014.
- [30] Y. Jiang, X. Gong, D. Liu, Y. Cheng, C. Fang, X. Shen, J. Yang, P. Zhou, and Z. Wang, "Enlightengan: Deep light enhancement without paired supervision," *IEEE Transactions on Image Processing*, vol. 30, pp. 2340–2349, 2021.
- [31] C. Guo, C. Li, J. Guo, C. C. Loy, J. Hou, S. Kwong, and R. Cong, "Zero-reference deep curve estimation for low-light image enhancement," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 1780–1789.
- [32] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.
- [33] S. Woo, J. Park, J.-Y. Lee, and I. So Kweon, "Cbam: Convolutional block attention module," in *Proceedings of the European Conference on Computer Vision*, 2018, pp. 3–19.
- [34] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Physica D: nonlinear phenomena*, vol. 60, no. 1-4, pp. 259–268, 1992.
- [35] P. Getreuer, "Rudin-osher-fatemi total variation denoising using split bregman," *Image Processing On Line*, vol. 2, pp. 74–95, 2012.
- [36] Z. Wang, E. P. Simoncelli, and A. C. Bovik, "Multiscale structural similarity for image quality assessment," in *The Thirty-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, vol. 2. Ieee, 2003, pp. 1398–1402.
- [37] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014.
- [38] A. Mittal, A. K. Moorthy, and A. C. Bovik, "No-reference image quality assessment in the spatial domain," *IEEE Transactions on Image Processing*, vol. 21, no. 12, pp. 4695–4708, 2012.
- [39] V. Mnih, N. Heess, A. Graves *et al.*, "Recurrent models of visual attention," *Advances in neural information processing systems*, vol. 27, pp. 2204–2212, 2014.
- [40] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 7132–7141.
- [41] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proceedings of the European Conference on Computer Vision*. Springer, 2014, pp. 818–833.
- [42] X. Dong, G. Wang, Y. Pang, W. Li, J. Wen, W. Meng, and Y. Lu, "Fast efficient algorithm for enhancement of low lighting video," in *IEEE International Conference on Multimedia and Expo*. IEEE, 2011, pp. 1–6.

REFERENCES

- [43] V. Bychkovsky, S. Paris, E. Chan, and F. Durand, “Learning photographic global tonal adjustment with a database of input / output image pairs,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2011.
- [44] Y. P. Loh and C. S. Chan, “Getting to know low-light images with the exclusively dark dataset,” *Computer Vision and Image Understanding*, vol. 178, pp. 30–42, 2019.
- [45] M. Gharbi, J. Chen, J. T. Barron, S. W. Hasinoff, and F. Durand, “Deep bilateral learning for real-time image enhancement,” *ACM Transactions on Graphics*, vol. 36, no. 4, pp. 1–12, 2017.
- [46] R. Wang, Q. Zhang, C.-W. Fu, X. Shen, W.-S. Zheng, and J. Jia, “Underexposed photo enhancement using deep illumination estimation,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 6849–6857.