Representative Color Transform for Image Enhancement

Hanul Kim[1], Su-Min Choi[2], Chang-Su Kim[3], Yeong Jun Koh[2*]

[1]Seoul National University of Science and Technology
[2]Chungnam National University
[3]Korea University

hukim@seoultech.ac.kr, smchoi@o.cnu.ac.kr, changsukim@korea.ac.kr, yjkoh@cnu.ac.kr,

Abstract

Recently, the encoder-decoder and intensity transformation approaches lead to impressive progress in image enhancement. However, the encoder-decoder often loses details in input images during down-sampling and up-sampling processes. Also, the intensity transformation has a limited capacity to cover color transformation between low-quality and high-quality images. In this paper, we propose a novel approach, called representative color transform (RCT), to tackle these issues in existing methods. RCT determines different representative colors specialized in input images and estimates transformed colors for the representative colors. It then determines enhanced colors using these transformed colors based on the similarity between input and representative colors. Extensive experiments demonstrate that the proposed algorithm outperforms recent state-of-the-art algorithms on various image enhancement problems.

1. Introduction

Nowadays, more and more people take photographs to record and to share their valuable moments. Unfortunately, their photographs often have low dynamic ranges or distorted color tones due to inadequate lighting conditions. Therefore, image enhancement becomes popular to improve the visual aesthetics of these photos. For image enhancement, many attempts have been proposed, and considerable progress has been made.

In particular, some studies [6, 50, 52, 22] based on the encoder-decoder architecture [38] in Figure 1a provide promising results by learning a robust non-linear mapping from large amounts of paired data composed of low-quality and high-quality images. In these models, the encoder extracts features from the input image to exploit the high-level context information for image enhancement. The decoder conveys the high-level information to low-level pixel values while recovering the spatial information. Although these methods have led to the performance improvement, they have some limitations. First, details of the input image are not preserved in the up-sampling process of the decoder, even though they employ skip-connections. Second, these approaches train networks with fixed input size, which makes it difficult to enhance images of arbitrary spatial resolutions in the inference phase.

To overcome these issues, some methods [7, 21, 34, 16, 25, 13] estimate transformation functions to enhance images globally as in Figure 1b. Since these global enhancement methods do not require the down-sampling and up-sampling processes for image enhancement, images can be enhanced while preserving details. However, the existing global methods rely on intensity transformation functions on specific color space *e.g.* RGB [21, 13] or CIELab [7], pre-defined lookup tables [54], and pre-defined enhancement operations [34, 16, 25]. Also, they perform channel-wise color transformation and thus fail to consider all channels simultaneously. These pre-defined models have the limited capacity to cover color transformation between low-quality and high-quality images.

In this paper, we propose a novel enhancement approach, called representative color transform (RCT), which effectively achieves a large capacity for color transformation. First, we encode an input image to extract the high-level context information for image enhancement. Using the high-level context, we determine representative colors for the input image and estimate transformed colors for the representative colors, as in Figure 1c. Then, we compute the similarity between the input image and the representative colors in an embedding space. Finally, we develop a representative color transform to obtain the enhanced image by combining the similarity and the representative color transformation. Based on the proposed RCT, we propose a representative color transform network (RCTNet), which con-

*Corresponding author

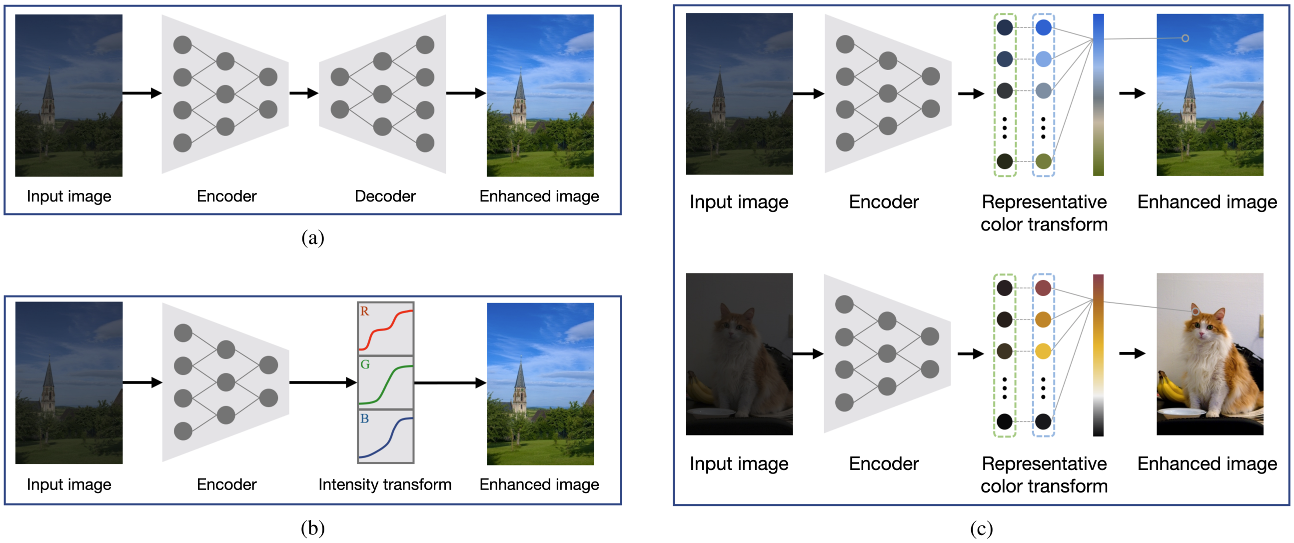


Figure 1: Outlines of image enhancement approaches: (a) encoder-decoder, (b) intensity transformation, and (c) representative color transform models.

sists of encoder, feature fusion, global RCT, and local RCT modules. The proposed RCTNet predicts different representative colors specialized in input images as in Figure 1c and enlarges the capacity for color transformation by combining several representative color transformations.

Experimental results demonstrate that the proposed RCTNet outperforms recent state-of-the-art algorithms on the MIT-Adobe 5K dataset [3]. Also, we validate the scalability of the proposed RCT on specific image enhancement problems: low-light image enhancement [49] and underwater image enhancement [28, 19].

The main contributions of this paper are three folds:

- The representative color transformation to enlarge the capacity for color transformation is developed for image enhancement.
- Development of RCTNet composed of encoder, feature fusion, global RCT, and local RCT modules.
- We demonstrate excellent scalability of RCTNet for various image enhancement problems.

2. Related Work

Early studies on image enhancement improve the global contrast of an input image. For instance, power-law (gamma) and logarithmic transformation [12], which map input pixel values to output pixel values using pre-defined transformations, are well-known enhancement methods. Histogram equalization [12] improves the limited dynamic range of an image by modifying its histogram. Many attempts [23, 48, 41, 2, 26] have been developed based on these approaches to enhance the visual quality.

Recent image enhancement methods mainly focus on learning mapping functions between low-quality and high-quality images based on data-driven approaches. Bychkovsky *et al*. [3] provided the MIT-Adobe 5K dataset that includes 5,000 input images, where 5 different photographers manually enhance each image. This dataset is widely adopted to train image enhancement models based on deep learning. Yan *et al*. [51] proposed the first deep learning model for image enhancement, where the network predicts a pixel-wise color mapping from hand-crafted feature descriptors. Lore *et al*. [32] employed a stacked sparse denoising autoencoder to enhance a low-light image. However, these methods [3, 32] employed neural networks with small receptive fields. As a result, their models may be insufficient to exploit high-level contexts for image enhancement.

The encoder-decoder structure [38] in Figure 1a has drawn much attention to image enhancement. The encoder incrementally increases the size of receptive fields by reducing the input's resolution to extract a deep feature containing the useful high-level information. From the deep feature, the decoder recovers the original resolution while enhancing images. Based on the encoder-decoder approach, Chen *et al*. [6] introduced the U-Net structure, which yields a residual image to enhance the input image. Gharbi *et al*. [11] predicted affine coefficients for each pixel in a low-resolution image and developed the bilateral interpolation method that effectively restores the image's original resolution. Wang *et al*. [46] decomposed an input image into the reflectance and illumination and estimated the illumination to enhance the input image. Xu *et al*. [50] developed the frequency-based decomposition for enhancement of low-light images. Yang *et al*. [53] constructed two encoder-decoder structures for image correction of underexposed inputs. Yang *et al*. [52] proposed the deep recursive band network and trained it in the semi-supervised framework. Kim *et al*. [22] designed the encoder-decoder network to produce a personalized image according to the user's preference. However, these encoder-decoder architectures [6, 11, 46, 50, 52, 22] have the problem that details of input images are not preserved in down-sampling and up-sampling processes. Also, they train networks with fixed input sizes, which makes it difficult to enhance images of arbitrary spatial resolutions.

As in Figure 1b, some methods [7, 13, 21, 34, 16, 25, 54] perform global enhancement through transformation functions or pre-defined enhancement operations. Deng *et al*. [7] estimated piece-wise intensity transform functions on the CIELab color space. Guo *et al*. [13] developed pixel-wise and high-order curves for dynamic range adjustment of an input image. Kim *et al*. [21] proposed the non-monotonic and channel-wise intensity transformation for both paired and unpaired image enhancement. In [34, 16, 25], neural networks are trained to select the best operation among some pre-defined enhancement operations based on deep reinforcement learning. Zeng *et al*. [54] learned image-adaptive 3-dimension lookup tables for global image enhancement. These global-based methods [7, 13, 21, 34, 16, 25, 54] can enhance low-quality images without image resize unlike the encoder-decoder models. However, they have the limitation in that transformation functions on the pre-defined color space, pre-defined lookup tables, or pre-defined operations may not be sufficient to estimate highly non-linear mapping between low-quality and high-quality images. In contrast, the proposed method estimates adaptive representative colors according to the input image, and predicts color transformation for each representative color based on the attention mechanism.

Finally, we review palette-based image enhancement methods [5, 42], which interpolate colors based on the palette colors. Chang *et al*. [5] set initial palettes using the K-mean clustering, and then users manually change pallet colors for image enhancement. Tan *et al*. [42] determined initial palettes using vertices of the convex hull wrapped around input colors. The proposed algorithm is related to them in that representative colors are similar to initial palettes. However, the proposed algorithm automatically determines representative colors and their transformed colors, while they require user interaction to update palettes.

3. Method

In this section, we propose the representative color transform (RCT), which is a simple and effective approach to improve the visual quality of input images. Based on RCT, we develop the representative color transform network (RCTNet) that contains global and local enhancement modules, which are trained in an end-to-end manner. Figure 2 summarizes the proposed RCTNet architecture.

3.1. Representative Color Transform

Let $\mathbf{X} \in \mathbb{R}^{H\times W\times 3}$ denote an input low-quality image, where $H \times W$ is the spatial resolution of the image. We encode it to a feature representation $\mathbf{Z}$ to embed the high-level context for image enhancement. Given $\mathbf{Z}$, we extract features and transformed colors for N representative colors. Let $\mathbf{R}$ denote the set of representative features, which is given by

$$\mathbf{R} = [\mathbf{r}_1, \mathbf{r}_2, \cdots, \mathbf{r}_N] \in \mathbb{R}^{C\times N} \tag{1}$$

where $\mathbf{r}_i$ denotes a feature vector of ith representative color and C is a feature dimension. Also, the set of transformed colors is defined as

$$\mathbf{T} = [\mathbf{t}_1, \mathbf{t}_2, \cdots, \mathbf{t}_N] \in \mathbb{R}^{3\times N} \tag{2}$$

where $\mathbf{t}_i$ denotes the transformed RGB values of ith representative color. In other words, the ith representative color should be transformed to $\mathbf{t}_i$ for image enhancement.

Note that $\mathbf{T}$ contains the transformation for only N representative colors, not all colors. Thus, we should map each pixel color in the input image to the representative colors. For this purpose, we compute the similarity between the input color and the representative features in the embedding space and perform the color transformation based on the similarity. We extract a image feature $\mathbf{F} \in \mathbb{R}^{H\times W\times C}$ from the input $\mathbf{X}$ using a stack of convolution layers. Then, we perform the matrix multiplication to obtain the attention matrix $\mathbf{A}$ via the scaled-dot product [45],

$$\mathbf{A} = \mathrm{softmax}(\frac{\mathbf{F}_\mathrm{r}\mathbf{R}}{\sqrt{C}}) \in \mathbb{R}^{HW\times N} \tag{3}$$

where $\mathbf{F}_\mathrm{r} \in \mathbb{R}^{HW\times C}$ a is reshaped tensor of $\mathbf{F}$. The element a_{ij} of $\mathbf{A}$ is the attention weight that represents the similarity between ith pixel in the input image and the jth representative color. Thus, the attention matrix determines all similarities on every transformed color in $\mathbf{T}$ for each pixel in the input image.

Let us consider enhancement of ith pixel in the input image. Then, enhanced RGB values of ith pixel is determined by the combination of N transformed colors for the representative colors with attention weights as $\sum_{j=1}^{N} a_{ij}\mathbf{t}_j$. To this end, the enhanced image $\mathbf{Y}$ is obtained by

$$\mathbf{Y} = \mathbf{A}\mathbf{T}^T. \tag{4}$$

Compared to existing image enhancement models, the proposed RCT has several advantages. First, the proposed method has a large capacity to cover the color transformation between input and enhanced images. This is because the proposed algorithm yields adaptive representative colors specialized in the input image and enhances input images based on the combination of N transformed colors.

Table 1: Specification of the encoder architecture.

Stage	Operations	Outputs
0	resize	$256 \times 256 \times 3$
1	conv-bn-swish, k3x3	$128 \times 128 \times 16$
2	conv-bn-swish, k3x3	$64 \times 64 \times 32$
3	conv-bn-swish, k3x3	$32 \times 32 \times 64$
4	conv-bn-swish, k3x3	$16 \times 16 \times 128$
5	conv-bn-swish, k3x3	$8 \times 8 \times 256$
6	conv-bn-swish-pool, k1x1	$1 \times 1 \times 1024$

Second, the proposed method is independently performed on each pixel. In other words, it can enhance an input image of the arbitrary size without any image resizing. This prevents performance degradation due to artifacts, such as blur, that occurs in the resize of enhanced images.

3.2. Representative Color Transform Network

As in Figure 2, the proposed RCTNet consists of four modules: encoder, feature fusion, global RCT, and local RCT. Given an input low-quality image $\mathbf{X}$, RCTNet produces a high-quality image:

$$\tilde{\mathbf{Y}} = \alpha \mathbf{Y}_{\mathrm{G}} + \beta \mathbf{Y}_{\mathrm{L}} \tag{5}$$

where $\mathbf{Y}_{\mathrm{G}}$ and $\mathbf{Y}_{\mathrm{L}}$ are enhanced images obtained from the global and local RCT modules, respectively. Also, α and β are non-negative learnable weights to combine two images effectively. Let us describe each module subsequently.

Encoder: Encoder is a convolutional neural network to encode an input image to extract the high-level context information for image enhancement. Table 1 describes the detailed architecture of the encoder. The input image is resized to 256×256 and fed into the encoder, composed of a stack of 6 'conv-bn-swish' blocks. Each 'conv-bn-swish' block contains a convolution, a batch normalization [17], and a swish activation [37] layers. All convolution layers except the last block have 3×3 filters. Unlike the others, the last block uses a convolution layer with 1×1 filter and employs a global average pooling layer to extract a global feature vector. In the encoder, we extract multi-scale feature maps from the last four blocks to combine them in the feature fusion module.

Feature Fusion: Feature maps provide different context information according to resolutions. In general, coarse-scale feature maps contain the global context due to large receptive fields. In contrast, fine-scale feature maps preserve the detailed local context. Since both global and local contexts are essential for image enhancement, we aggregate multi-scale feature maps through the feature fusion module.

To construct the feature fusion module, we employ the bidirectional cross-scale connections [43]. In the feature fusion in Figure 2, nodes that have a single input represents the 'conv-bn-swish' block. On the other hand, nodes, which have multiple inputs, contains a feature fusion layer before the 'conv-bn-swish' block to mix multiple inputs effectively. When M inputs are provided to the feature fusion layer, the output of the feature fusion layer is defined as

$$\mathbf{O} = \sum_{i=1}^{M} \frac{w_i}{\epsilon + \sum_j w_j} \cdot \mathbf{I}_i \tag{6}$$

where w_i is a non-negative learnable weight for the ith input $\mathbf{I}_i$ and $\epsilon = 0.0001$. All nodes have 128 convolution filters of size 3×3 except nodes at the coarsest-scale (red nodes in Figure 2). Since the spatial resolution of the coarsest feature map is 1×1, these node have convolution filters of size 1×1.

Global RCT: Let $\mathbf{Z}_{\mathrm{G}} \in \mathbb{R}^{C'}$ denote an output feature at the coarsest-scale in the feature fusion module, where C' is set to 128. By analyzing the feature vector $\mathbf{Z}_{\mathrm{G}}$, which includes the global context for image enhancement, the global RCT module determines representative features $\mathbf{R}_{\mathrm{G}} \in \mathbb{R}^{C \times N_{\mathrm{G}}}$ and transformed colors $\mathbf{T}_{\mathrm{G}} \in \mathbb{R}^{3 \times N_{\mathrm{G}}}$ through two different 'conv-bn-swish-conv' blocks. One 'conv-bn-swish-conv' block yields a vector with CN_{G} dimension, while another block produces a vector with $3N_{\mathrm{G}}$ dimension. These output vectors are reshaped to the 2D structures, $\mathbf{R}_{\mathrm{G}}$ and $\mathbf{T}_{\mathrm{G}}$, respectively. In this work, we set C and N_{G} to 16 and 64, respectively. Also, the input image is transformed to the image feature $\mathbf{F} \in \mathbb{R}^{H \times W \times C}$ through one 'conv-bn-swish-conv' block and $\mathbf{F}$ is reshaped to $\mathbf{F}_r \in \mathbb{R}^{HW \times C}$. Finally, the global enhanced image $\mathbf{Y}_{\mathrm{G}}$ is obtained by applying $\mathbf{R}_G$, $\mathbf{T}_G$, and $\mathbf{F}_r$ to (3) and (4).

Local RCT: The local RCT module determines region-wise representative colors to consider local region characteristics for image enhancement. For this purpose, the local RCT module takes a feature map $\mathbf{Z}_{\mathrm{L}} \in \mathbb{R}^{32 \times 32 \times C'}$, extracted from the finest-scale in the feature fusion module, whose the spatial resolution is 32×32. Then, given $\mathbf{Z}_{\mathrm{L}}$, the sets of representative features and transformed colors are generated for each spatial position. Specifically, $\mathbf{Z}_{\mathrm{L}}$ are fed into two different 'conv-bn-swish-conv' blocks, where the first convolution layers have 128 convolution filters of size 3×3 and each second convolution layer has CN_{L} and $3N_{\mathrm{L}}$ filters of size 3×3, respectively. Then, the local RCT produces the representative feature sets $\mathbf{R}_{\mathrm{L}} \in \mathbb{R}^{32 \times 32 \times C \times N_{\mathrm{L}}}$ and the transformed color sets $\mathbf{T}_{\mathrm{L}} \in \mathbb{R}^{32 \times 32 \times 3 \times N_{\mathrm{L}}}$, where N_{L} is 16. To this end, the set of representative features $\mathbf{R}_{\mathrm{L}}(u, v)$ and the set of transformed colors $\mathbf{T}_{\mathrm{L}}(u, v)$ are obtained for each spatial position (u, v).

Given $\mathbf{R}_{\mathrm{L}}$ and $\mathbf{T}_{\mathrm{L}}$, the local RCT module assigns different sets of representative features and transformed colors to

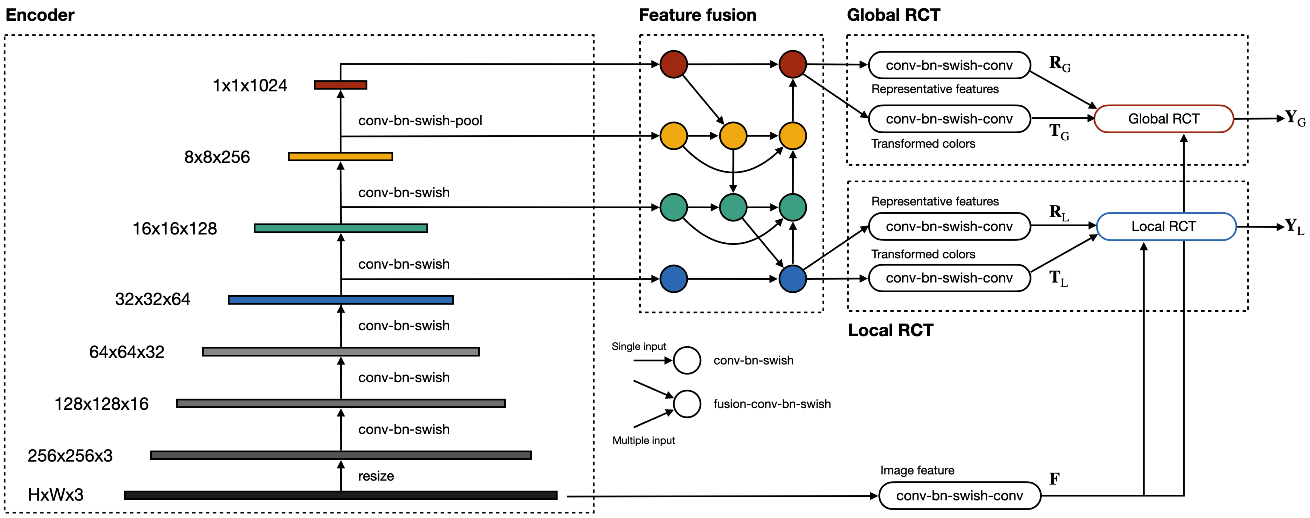


Figure 2: An overview of the proposed RCTNet.

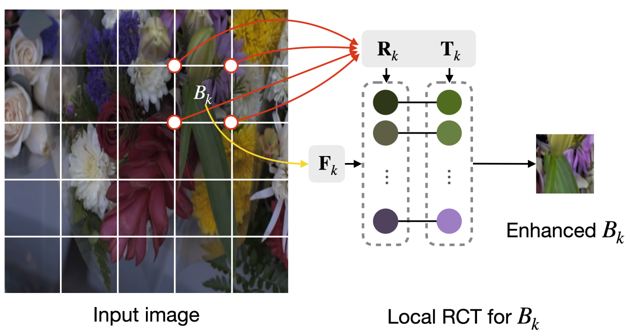


Figure 3: An illustration of Local RCT.

an input color according to its pixel coordinates. First, we set a 31×31 uniform mesh grid on the input image, and thus there are 32×32 corner points. We regard representative features and transformed colors at corner points on spatial position (u, v) as $\mathbf{R}_{\mathrm{L}}(u, v)$ and $\mathbf{T}_{\mathrm{L}}(u, v)$, respectively. Then, grid-wise RCT is performed in the local RCT module. Specifically, the kth grid B_k has four corner points, which means that B_k are related to the four sets of representative features and transformed colors. We then determine the representative features $\mathbf{R}_k$ for the grid B_k by concatenating the four sets of representative features at corner points. Also, the transformed colors $\mathbf{T}_k$ for B_k is obtained similarly. From the image feature $\mathbf{F}$, a grid feature $\mathbf{F}_k$ is extracted by cropping on the grid region. Finally, given $\mathbf{R}_k$, $\mathbf{T}_k$, and $\mathbf{F}_k$, enhanced colors for B_k is computed by (3) and (4). The local RCT module repeat this process for all grids to yield the local enhanced image $\mathbf{Y}_{\mathrm{L}}$. Figure 3 shows an example of how the local RCT is performed for the grid B_k. For the simplicity, we set a 5×5 mesh grid in this example.

Though the encoder requires the fixed-size input to extract multi-scale feature maps, both global and local RCT modules enhance input images without any image resize by extracting the image feature $\mathbf{F}$ without any down-sampling.

3.3. Loss Functions

Let us consider a pair $(\mathbf{X}, \mathbf{Y})$, where $\mathbf{X}$ and $\mathbf{Y}$ are an input low-quality image and its high-quality image, respectively. Given $\mathbf{X}$, the proposed RCTNet produces an enhanced image $\tilde{\mathbf{Y}}$. We then define the loss function between $\tilde{\mathbf{Y}}$ and $\mathbf{Y}$ by

$$\mathcal{L} = \|\tilde{\mathbf{Y}} - \mathbf{Y}\|_1 + \lambda \sum_{k=2,4,6} \|\phi^k(\tilde{\mathbf{Y}}) - \phi^k(\mathbf{Y})\|_1. \quad (7)$$

Here, the first term is the mean absolute error between the predicted and ground-truth enhanced images. And the second term penalizes the difference between them in the well-defined embedding space. Specifically, the embedding function $\phi^k(\cdot)$ is the output of kth layer in VGG-16 [40], which is pre-trained on the ImageNet [39] dataset. The hyper parameter λ is fixed to 0.04 to balance two terms.

4. Experiments

In this section, we verify the effectiveness of the proposed method through extensive experiments:

- We compare the proposed algorithm with recent state-of-the-arts in standard image enhancement.
- We evaluate the scalability of the proposed RCTNet on specific image enhancement problems: low-light image enhancement and underwater image enhancement.
- We analyze parameters and components of RCTNet through ablation studies on the MIT-Adobe 5K dataset [3].

Table 2: Quantitative comparison on the MIT-Adobe 5K dataset [3]. The best results are boldfaced and the second best ones are underlined.

Method.	PSNR	SSIM
HDRNet [11]	23.44	0.882
DPE [6]	23.34	0.873
DUPE [46]	23.61	0.887
DLPF [33]	24.48	0.887
3D LUT [54]	25.21	0.922
GEN-LEN [21]	25.88	**0.925**
RCTNet	<u>26.02</u>	0.915
RCTNet + BF	**26.07**	<u>0.923</u>

Table 3: Quantitative comparison on the LoL dataset [49]. The best results are boldfaced and the second best ones are underlined.

Method	PSNR	SSIM
NPE [47]	16.97	0.589
LIME [14]	15.24	0.470
SRIE [9]	17.34	0.686
RRM [31]	17.34	0.686
SICE [4]	19.40	0.690
DRD [49]	16.77	0.559
KinD [55]	20.87	0.802
DRBN [52]	20.13	**0.830**
ZeroDCE [13]	14.86	0.559
EnlightenGAN [20]	15.34	0.528
RCTNet	<u>22.67</u>	0.788
RCTNet + BF	**22.81**	<u>0.827</u>

We adopt PSNR and SSIM metrics for the quantitative evaluation in all experiments, which measure color and structural similarity between predicted and ground-truth images. More results are available in the supplementary material.

4.1. Datasets

MIT-Adobe 5K: The MIT-Adobe 5K dataset [3] consists of 5,000 images, each of which was manually enhanced by five different photographers (A/B/C/D/E). There are five sets (one set per photographer) consisting of 5,000 pairs of input and retouched images. Among these sets, we only use images retouched by photographer C as done in most existing image enhancement methods [6, 46, 21]. And we decompose it to 4,500 and 500 images for training and test set, respectively.

Low Light (LoL): LoL [49] is a dataset for low-light image enhancement. The LoL dataset contains 500 pairs of low-light and normal-light images in which 500 pairs are separated into 485 training images and 15 testing images. We use the training images for training RCTNet and the test images for experiments.

Enhancing Underwater Visual Perception (EUVP): The EUVP dataset [19] provides subsets of the paired and unpaired collections for underwater images. The paired dataset separates pairs of low-quality and high-quality images into 11435, 570, and 515 pairs for the training, validation, and test sets. The pairs in the training and test set are used for the training and evaluation, respectively.

Underwater Image Enhancement Benchmark (UIEB): The UIEB dataset [28] includes 890 pairs of underwater image and its enhanced image. These pairs are divided into 800 and 90 for training and test, respectively. We train RCTNet using 800 training images and evaluate the proposed algorithm on the test set.

4.2. Implementation Details

We train the proposed model for 100, 500, 500, and 100 epochs with batch size of 8 for the MIT-Adobe-5K, LoL, EUVP, and UIEB datasets, respectively. We use Adam optimizer [24] to minimize the loss function, with an initial learning rate of 5.0×10^{-4} and a weight decay of 1.0×10^{-5}. We decrease learning rate according to the cosine learning rate scheduling. Following the literature [21], we randomly crop image and then rotate them by multiples of 90 degrees for data augmentation. We fix the hyperparameter λ to 0.04.

4.3. Comparison with state-of-the-arts

MIT-Adobe 5K: We compare the performance of the proposed method with recent state-of-the-art methods [11, 6, 46, 33, 54, 21]. For comparison, we obtain the results of existing algorithms using their published source codes and default settings. Table 2 lists the PSNR and SSIM performances on the MIT-Adobe 5K dataset. We resize each test image to have 512 pixels in the long side of each test image as done in existing algorithms [33, 21] for the comparison. In Table 2, the proposed RCTNet achieves the best on PSNR, which indicates that the proposed RCTNet is effective for color enhancement. In contrast, RCTNet yields the second best performance on SSIM, since it does not perform the spatial filtering that suppresses noises. To address denoising problem, we can employ the simple filtering method, such as the bilateral filter (BF) [44], as the post-processing. As in Table 2, the bilateral filter improves the SSIM score to 0.923.

Low-light Image Enhancement: Next, we evaluate the proposed RCTNet on the low-light image enhancement problem. Table 3 compares the proposed RCTNet with state-of-the-art low-light image enhancement algorithms [47, 14, 9, 31, 4, 49, 55, 52, 13, 20] on the LoL dataset [49]. ZeroDCE [13] and EnlightenGAN [20] provides relatively lower performance because they train their network with unpaired images. The proposed RCTNet achieves the best PSNR score by enhancing input colors ef-

(a) Input (b) KinD [55] (c) RCTNet (d) RCTNet + BF (e) Ground-truth

Figure 4: Qualitative comparison on the LoL dataset [49].

Table 4: Quantitative comparison on the UIEB dataset [28]. The best results are boldfaced and the second best ones are underlined.

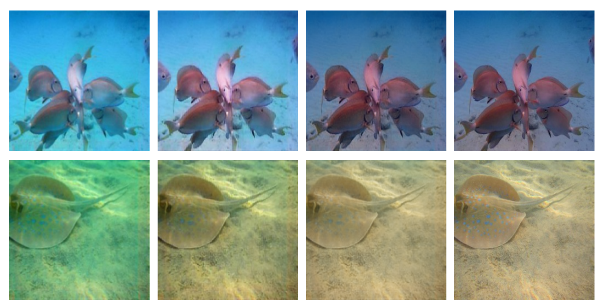
Method	PSNR	SSIM
Fusion [1]	17.60	0.772
Retinex [10]	17.02	0.607
GDCP [35]	12.09	0.512
Histogram [30]	15.82	0.539
Blurriness [36]	15.32	0.603
Water CycleGAN [29]	15.75	0.521
Dense GAN [15]	17.28	0.443
WaterNet [28]	19.11	0.797
Ucolor [27]	<u>20.63</u>	<u>0.770</u>
RCTNet	**22.45**	**0.891**

fectively. In contrast, the proposed algorithm provides the relatively low SSIM score, since low-light images include a lot of sensor noises due to their shooting conditions. This indicates that the representative color transform is less effective in denoising. Figure 4 shows enhancement results of the proposed RCTNet and KinD [55], which is the best existing method in Table 3. In these examples, RCTNet enhances low-light images to have more similar color-tones to their ground-truths than KinD, while KinD suppresses noises more effectively. However, this weakness of the proposed method can be alleviated by employing simple denoising techniques as the post-processing. For instance, we observe that simple bilateral filter improves the PSNR and SSIM scores to 22.81dB and 0.827 as in Table 3. Also, the simple post processing successfully suppress noise as in Figure 4d.

Underwater Image Enhancement: Finally, we assess the performance of the proposed RCTNet on the underwater image enhancement problem. Tables 4 and 5 compare the proposed algorithm with the existing underwater image enhancement algorithms [1, 10, 35, 30, 36, 29, 8, 28, 19, 15, 18, 27] on the UIEB [28] and EUVP [19]

Table 5: Quantitative comparison on the EUVP dataset [19]. The best results are boldfaced and the second best ones are underlined.

Method	PSNR	SSIM
U-GAN [8]	23.49	<u>0.842</u>
Funie-GAN [19]	23.40	0.827
Deep SESR [18]	<u>24.21</u>	0.840
RCTNet	**26.43**	**0.891**



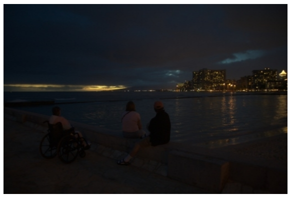
(a) Input (b) Deep SESR [18] (c) RCTNet (d) GT

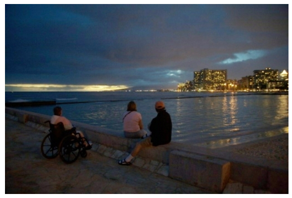
Figure 5: Qualitative comparison on the EUVP dataset [19].

datasets, respectively. Underwater images tend to be degraded by wavelength-dependent absorption and scattering due to shooting environments. Nevertheless, RCTNet faithfully enhances underwater images and significantly outperforms the existing state-of-the-art methods in PSNR and SSIM scores on both datasets. Figures 5 qualitatively compare the proposed algorithm with the second-best methods in EUVP dataset. Remarkably, we see that RCTNet produces visually pleasing results.

4.4. Ablation Studies

Component Analysis: We analyze the efficacy of the three components of feature fusion, global RCT, and local RCT modules in RCTNet. In this test, we measure the three


(a) Input


(b) Global IT


(c) Global RCT


(d) Ground-truth

Figure 6: Qualitative comparison between IT and RCT on the Adobe5k dataset [3]

performances of RCTNet: 1) without the feature fusion, 2) without the global RCT, and 3) without the local RCT. Let us refer to these settings as 'w/o global RCT,' 'w/o local RCT,' and 'w/o feature fusion.' Table 6 summarizes the average PSNR scores of theses settings on four datasets. Without the global RCT, the local RCT, or the future fusion, the PSNR scores are degraded severely. This indicates that the proposed components are essential for image enhancement.

Representative Color Transform: We verify the efficacy of the proposed RCT by replacing it with different enhancement models. First, we employ the decoder in the U-Net architecture [6], which includes 6 up-sample blocks to perform bilinear interpolation, concatenation, and convolution filtering. Second, we substitute the proposed RCT with the channel-wise intensity transform model in [21]. Third, we exclude the local RCT module from RCTNet for the fair comparison. All enhancement models include the same encoder and the feature fusion module in Figure 2.

Tables 7 summarizes the PSNR scores of three image enhancement approaches. Here, 'Decoder,' 'Global IT,' and 'Global RCT' denote the encoder-decoder model, the global intensity transform, and the proposed global RCT, respectively. 'Decoder' provides the worst performance on all datasets except LoL dataset. This indicates that approaches based on the color transformation are more effective for image enhancement than encoder-decoder structures. Also, 'Global RCT' outperforms 'Global IT' by providing larger color transformation capacities.

Figure 6 illustrates enhancement results of 'Global RCT' and 'Global IT' on the MIT-Adobe5k dataset. The input image in Figure 6a has low intensity values on the blue channel in most regions including sky, sea, and ground. To enhance the input image effectively, intensity values on the

Table 6: PSNR scores according to different components. The best results are boldfaced.

Method.	Adobe 5K	LoL	UIEB	EUVP
RCTNet	**26.02**	**22.67**	**22.81**	**26.43**
w/o global RCT	25.43	22.12	21.99	24.30
w/o local RCT	25.57	22.35	22.41	24.46
w/o feature fusion	25.68	22.30	22.34	25.19

Table 7: PSNR scores according to image enhancement approaches. The best results are boldfaced.

Method.	Adobe 5K	LoL	UIEB	EUVP
Decoder [6]	24.26	21.25	21.70	23.86
Global IT [21]	24.97	21.20	22.23	24.05
Global RCT	**25.57**	**22.35**	**22.41**	**24.46**

blue channel in sky and sea should increase more than in ground. However, 'Global IT' increases similar blue intensities in most regions, as in Figure 6b, since pixels in sky, sea, and ground have similar blue intensities in the input image. Thus, blue intensities in the sky and sea regions are not sufficiently enhanced. As pointed out in [21], this indicates that 'Global IT' is vulnerable to one-to-many mapping. On the contrary, as in Figure 6c, 'Global RCT' can perform different modifications to sky, sea, and ground. So, it provides the faithful enhanced result as in the manually retouched result in Figure 6d.

5. Conclusion

We proposed the novel image enhancement algorithm based on the representative color transform. The proposed RCT determines different representative colors specialized in input images and enhances input images using representative features and transformed colors. Then, with the proposed RCT, we developed RCTNet, composed of encoder, feature fusion, global RCT, and local RCT modules. The global RCT predicts representative colors for an input image, while the local RCT determines region-wise representative colors to consider local region characteristics for image enhancement. Extensive experiments demonstrated that the proposed RCTNet outperforms recent state-of-the-art algorithms on various datasets with standard image enhancement, low-light image enhancement, and underwater image enhancement.

Acknowledgement

This work was supported by the National Research Foundation of Korea (NRF) grants funded by the Korea government (MSIT) (No.NRF-2018R1A2B3003896, No. NRF-2019R1F1A1062907, and No. NRF-2021R1A4A1031864)

References

[1] Cosmin Ancuti, Codruta Orniana Ancuti, Tom Haber, and Philippe Bekaert. Enhancing underwater images and videos by fusion. In *Proc. CVPR*, pages 81–88, 2012. 7

[2] Tarik Arici, Salih Dikbas, and Yucel Altunbasak. A histogram modification framework and its application for image contrast enhancement. *IEEE Trans. Image Process.*, 18(9):1921–1935, 2009. 2

[3] Vladimir Bychkovsky, Sylvain Paris, Eric Chan, and Frédo Durand. Learning photographic global tonal adjustment with a database of input/output image pairs. In *Proc. CVPR*, pages 97–104, 2011. 2, 5, 6, 8

[4] Jianrui Cai, Shuhang Gu, and Lei Zhang. Learning a deep single image contrast enhancer from multi-exposure images. *IEEE Trans. Image Process.*, 27(4):2049–2062, 2018. 6

[5] Huiwen Chang, Ohad Fried, Yiming Liu, Stephen DiVerdi, and Adam Finkelstein. Palette-based photo recoloring. *ACM Trans. Graph.*, 34(4):139:1–139:11, 2015. 3

[6] Yu-Sheng Chen, Yu-Ching Wang, Man-Hsin Kao, and Yung-Yu Chuang. Deep photo enhancer: Unpaired learning for image enhancement from photographs with GANs. In *Proc. CVPR*, pages 6306–6314, 2018. 1, 2, 3, 6, 8

[7] Yubin Deng, Chen Change Loy, and Xiaoou Tang. Aesthetic-driven image enhancement by adversarial learning. In *Proc. ACM MM*, pages 870–878, 2018. 1, 3

[8] Cameron Fabbri, Md Jahidul Islam, and Junaed Sattar. Enhancing underwater imagery using generative adversarial networks. In *Proc. ICRA*, pages 7159–7165, 2018. 7

[9] Xueyang Fu, Delu Zeng, Yue Huang, Xiao-Ping Zhang, and Xinghao Ding. A weighted variational model for simultaneous reflectance and illumination estimation. In *Proc. CVPR*, pages 2782–2790, 2016. 6

[10] Xueyang Fu, Peixian Zhuang, Yue Huang, Yinghao Liao, Xiao-Ping Zhang, and Xinghao Ding. A retinex-based enhancing approach for single underwater image. In *Proc. ICIP*, pages 4572–4576, 2014. 7

[11] Michaël Gharbi, Jiawen Chen, Jonathan T Barron, Samuel W Hasinoff, and Frédo Durand. Deep bilateral learning for real-time image enhancement. *ACM Trans. Graph.*, 36(4):1–12, 2017. 2, 3, 6

[12] Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing (4th Edition)*. Pearson, 2018. 2

[13] Chunle Guo, Chongyi Li, Jichang Guo, Chen Change Loy, Junhui Hou, Sam Kwong, and Runmin Cong. Zero-reference deep curve estimation for low-light image enhancement. In *Proc. CVPR*, pages 1780–1789, 2020. 1, 3, 6

[14] Xiaojie Guo, Yu Li, and Haibin Ling. LIME: Low-light image enhancement via illumination map estimation. *IEEE Trans. Image Process.*, 26(2):982–993, 2016. 6

[15] Y. Guo, H. Li, and P. Zhuang. Underwater image enhancement using a multiscale dense generative adversarial network. *IEEE J. Ocean. Eng.*, 45(3):862–870, 2020. 7

[16] Yuanming Hu, Hao He, Chenxi Xu, Baoyuan Wang, and Stephen Lin. Exposure: A white-box photo post-processing framework. *ACM Trans. Graph.*, 37(2):1–17, 2018. 1, 3

[17] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proc. ICML*, pages 448–456, 2015. 4

[18] Md Jahidul Islam, Peigen Luo, and Junaed Sattar. Simultaneous enhancement and super-resolution of underwater imagery for improved visual perception. In *Proc. RSS*, 2020. 7

[19] Md Jahidul Islam, Youya Xia, and Junaed Sattar. Fast underwater image enhancement for improved visual perception. *IEEE Robot. Autom. Lett.*, 5(2):3227–3234, 2020. 2, 6, 7

[20] Yifan Jiang, Xinyu Gong, Ding Liu, Yu Cheng, Chen Fang, Xiaohui Shen, Jianchao Yang, Pan Zhou, and Zhangyang Wang. Enlightengan: Deep light enhancement without paired supervision. *IEEE Trans. Image Process.*, 30:2340–2349, 2021. 6

[21] Han-Ul Kim, Yeong Jun Koh, and Chang-Su Kim. Global and local enhancement networks for paired and unpaired image enhancement. In *Proc. ECCV*, 2020. 1, 3, 6, 8

[22] Han-Ul Kim, Yeong Jun Koh, and Chang-Su Kim. PieNet: Personalized image enhancement network. In *Proc. ECCV*, pages 374–390, 2020. 1, 3

[23] Yeong-Taeg Kim. Contrast enhancement using brightness preserving bi-histogram equalization. *IEEE Trans. Consum. Electro.*, 43(1):1–8, 1997. 2

[24] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proc. ICLR*, 2015. 6

[25] Satoshi Kosugi and Toshihiko Yamasaki. Unpaired image enhancement featuring reinforcement-learning-controlled image editing software. In *Proc. AAAI*, 2020. 1, 3

[26] Chulwoo Lee, Chul Lee, and Chang-Su Kim. Contrast enhancement based on layered difference representation of 2d histograms. *IEEE Trans. Image Process.*, 22(12):5372–5384, 2013. 2

[27] Chongyi Li, Saeed Anwar, Junhui Hou, Runmin Cong, Chunle Guo, and Wenqi Ren. Underwater image enhancement via medium transmission-guided multi-color space embedding. *IEEE Trans. Image Process.*, 30:4985–5000, 2021. 7

[28] Chongyi Li, Chunle Guo, Wenqi Ren, Runmin Cong, Junhui Hou, Sam Kwong, and Dacheng Tao. An underwater image enhancement benchmark dataset and beyond. *IEEE Trans. Image Process.*, 29:4376–4389, 2019. 2, 6, 7

[29] Chongyi Li, Jichang Guo, and Chunle Guo. Emerging from water: Underwater image color correction based on weakly supervised color transfer. *IEEE Sign. Process. Letters*, 25(3):323–327, 2018. 7

[30] Chong-Yi Li, Ji-Chang Guo, Run-Min Cong, Yan-Wei Pang, and Bo Wang. Underwater image enhancement by dehazing with minimum information loss and histogram distribution prior. *IEEE Trans. Image Process.*, 25(12):5664–5677, 2016. 7

[31] Mading Li, Jiaying Liu, Wenhan Yang, Xiaoyan Sun, and Zongming Guo. Structure-revealing low-light image enhancement via robust retinex model. *IEEE Trans. Image Process.*, 27(6):2828–2841, 2018. 6

[32] Kin Gwn Lore, Adedotun Akintayo, and Soumik Sarkar. LLNet: A deep autoencoder approach to natural low-light image enhancement. *Pattern Recognit.*, 61:650–662, 2017. 2

[33] Sean Moran, Pierre Marza, Steven McDonagh, Sarah Parisot, and Gregory Slabaugh. DeepLPF: Deep local parametric filters for image enhancement. In *Proc. CVPR*, pages 12826–12835, 2020. 6

[34] Jongchan Park, Joon-Young Lee, Donggeun Yoo, and In So Kweon. Distort-and-recover: Color enhancement using deep reinforcement learning. In *Proc. CVPR*, pages 5928–5936, 2018. 1, 3

[35] Yan-Tsung Peng, Keming Cao, and Pamela C Cosman. Generalization of the dark channel prior for single image restoration. *IEEE Trans. Image Process.*, 27(6):2856–2868, 2018. 7

[36] Yan-Tsung Peng and Pamela C Cosman. Underwater image restoration based on image blurriness and light absorption. *IEEE Trans. Image Process.*, 26(4):1579–1594, 2017. 7

[37] Prajit Ramachandran, Barret Zoph, and Quoc V Le. Searching for activation functions. In *Proc. ICLRW*, 2018. 4

[38] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Proc. MICCAI*, pages 234–241, 2015. 1, 2

[39] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *Int. J. Comput. Vis.*, 115(3):211–252, 2015. 5

[40] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *Proc. ICLR*, 2015. 5

[41] J Alex Stark. Adaptive image contrast enhancement using generalizations of histogram equalization. *IEEE Trans. Image Process.*, 9(5):889–896, 2000. 2

[42] Jianchao Tan, Jose Echevarria, and Yotam Gingold. Efficient palette-based decomposition and recoloring of images via rgbxy-space geometry. *ACM Trans. Graph.*, 37(6):1–10, 2018. 3

[43] Mingxing Tan, Ruoming Pang, and Quoc V Le. Efficientdet: Scalable and efficient object detection. In *Proc. CVPR*, pages 10781–10790, 2020. 4

[44] Carlo Tomasi and Roberto Manduchi. Bilateral filtering for gray and color images. In *Proc. ICCV*, pages 839–846, 1998. 6

[45] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proc. NeurIPS*, pages 5998–6008, 2017. 3

[46] Ruixing Wang, Qing Zhang, Chi-Wing Fu, Xiaoyong Shen, Wei-Shi Zheng, and Jiaya Jia. Underexposed photo enhancement using deep illumination estimation. In *Proc. CVPR*, pages 6849–6857, 2019. 2, 3, 6

[47] Shuhang Wang, Jin Zheng, Hai-Miao Hu, and Bo Li. Naturalness preserved enhancement algorithm for non-uniform illumination images. *IEEE Trans. Image Process.*, 22(9):3538–3548, 2013. 6

[48] Yu Wang, Qian Chen, and Baeomin Zhang. Image enhancement based on equal area dualistic sub-image histogram equalization method. *IEEE Trans. Consum. Electro.*, 45(1):68–75, 1999. 2

[49] Chen Wei, Wenjing Wang, Wenhan Yang, and Jiaying Liu. Deep retinex decomposition for low-light enhancement. In *Proc. BMVC*, 2018. 2, 6, 7

[50] Ke Xu, Xin Yang, Baocai Yin, and Rynson WH Lau. Learning to restore low-light images via decomposition-and-enhancement. In *Proc. CVPR*, pages 2281–2290, 2020. 1, 2, 3

[51] Zhicheng Yan, Hao Zhang, Baoyuan Wang, Sylvain Paris, and Yizhou Yu. Automatic photo adjustment using deep neural networks. *ACM Trans. Graph.*, 35(2):1–15, 2016. 2

[52] Wenhan Yang, Shiqi Wang, Yuming Fang, Yue Wang, and Jiaying Liu. From fidelity to perceptual quality: A semi-supervised approach for low-light image enhancement. In *Proc. CVPR*, pages 3063–3072, 2020. 1, 3, 6

[53] Xin Yang, Ke Xu, Yibing Song, Qiang Zhang, Xiaopeng Wei, and Rynson WH Lau. Image correction via deep reciprocating hdr transformation. In *Proc. CVPR*, pages 1798–1807, 2018. 2

[54] Hui Zeng, Jianrui Cai, Lida Li, Zisheng Cao, and Lei Zhang. Learning image-adaptive 3D lookup tables for high performance photo enhancement in real-time. *IEEE Trans. Pattern Anal. Mach. Intell.*, 2020. 1, 3, 6

[55] Yonghua Zhang, Jiawan Zhang, and Xiaojie Guo. Kindling the darkness: A practical low-light image enhancer. In *Proc. ACM MM*, pages 1632–1640, 2019. 6, 7