Representative Color Transform for Image Enhancement

Hanul Kim, Su-Min Choi, Chang-Su Kim, Yeong Jun Koh

Seoul National University of Science and Technology

Chungnam National University

Korea University

Partners: 110354016 統計碩二 蔡耀德、110354025 統計碩二 林映孝、110354032 統計碩二 徐偉鑫

1. Introduction

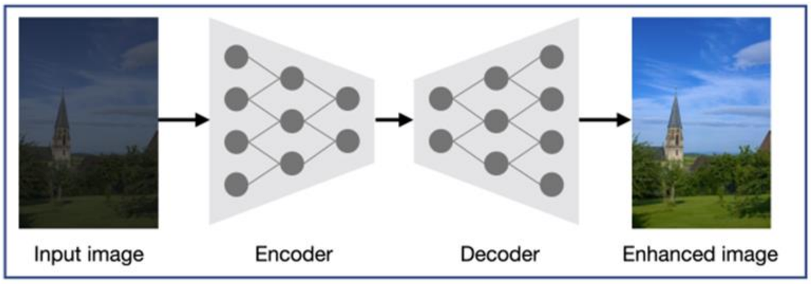
- Background

Nowadays, many people take photographs to record and 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.

- Encoder-decoder architecture

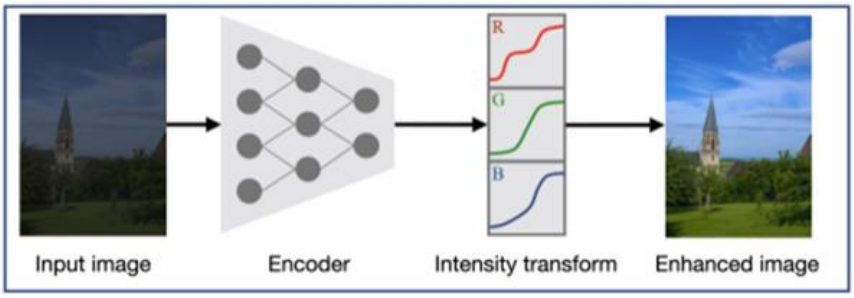
Some studies based on the encoder-decoder architecture provide promising results by learning robust non-linear mapping from large amounts of images. But encoder-decoder-based methods are not preserved in the up-sampling process of the decoder and these approaches train networks with fixed input size, which makes it difficult to enhance images of arbitrary spatial resolutions in the inference phase.

To overcome these issues, some methods estimate transformation functions to enhance images globally.



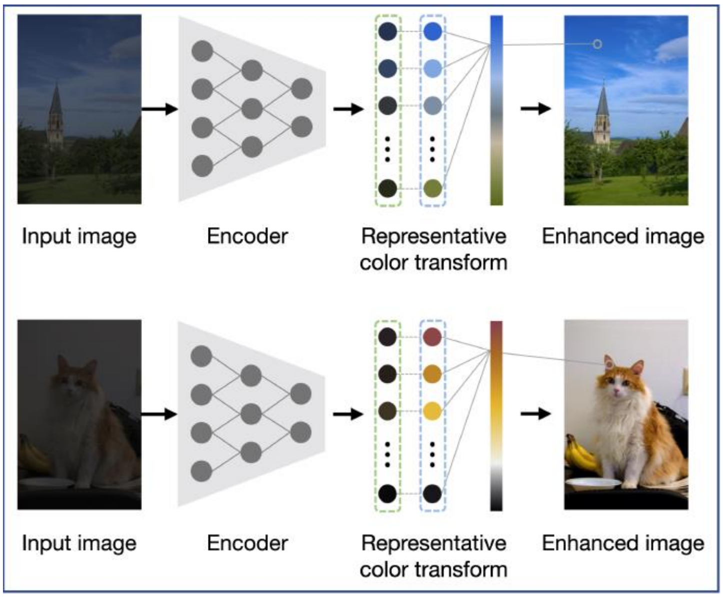
- Global-based method

Images can be enhanced while preserving details since these global enhancement methods do not require the down-sampling and up-sampling processes. However, the existing global methods rely on intensity transformation functions on specific color space e.g. RGB or CIELab, pre-defined lookup tables, and enhancement operations. Also, they perform channelwise 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.



- Proposed method - Representative Color Transform (RCT)

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

The main contributions of this paper are three folds:

(a) The representative color transformation to enlarge the capacity for color transformation is developed for image enhancement.

(b) Development of RCTNet composed of encoder, feature fusion, global RCT, and local RCT modules.

(c) 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, histogram equalization. Recent image enhancement methods mainly focus on learning mapping functions between low-quality and high-quality images based on data-driven approaches.

- Neural networks

Yan et al. and Lore et al. propose methods which are based on neural networks with small receptive fields, and their models may be insufficient to exploit high-level contexts for image enhancement.

- Encoder-decoder approach

Chen et al., Gharbi et al., Wang et al., Xu et al., Yang et al. propose methods which are based on the encoder-decoder approach, the encoder incrementally increases the size of receptive fields. But these encoder-decoder architectures do not preserved details of input images in down-sampling and upsampling processes.

- Global enhancement approcah

Some methods (Deng et al., Guo et al., Kim et al. Zeng et al.) perform global enhancement through transformation functions or pre-defined enhancement operations, these global-based methods enhance low-quality images without resize image. But they have limitations in transformation functions on the pre-defined color space, lookup tables, or operations. It may not be sufficient to estimate highly non-linear mapping between low-quality and high-quality images.

- RCT method

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.

3. Presented method:

- Representative Color Transform (RCT) structure

The proposed method, called Representative Color Transformation (RCT), aims to enhance low-quality images by embedding high-level context and performing color transformations.

1) The input image, represented as an H×W×3 matrix, is encoded into a feature representation Z as following:

   Input: $\mathbf{X} \in \mathbb{R}^{H\times W\times 3} \xrightarrow{\text{encode}}$ Feature representation: $\mathbf{Z}$

2) Extract features R and transformed colors T for N representative color

   Let $\mathbf{R} = [\mathbf{r}_1, \mathbf{r}_2, \cdots, \mathbf{r}_N] \in \mathbb{R}^{C\times N}$ as the set of representative features,

   where $\mathbf{r}_i$ denotes a feature vector of ith representative color and C is a feature dimension

   Let $\mathbf{T} = [\mathbf{t}_1, \mathbf{t}_2, \cdots, \mathbf{t}_N] \in \mathbb{R}^{3\times N}$ as the set of transformed colors,

   where $\mathbf{t}_i$ denotes a feature vector of ith representative color and it mapped by $\mathbf{R}$

3) The attention matrix A as following is computed by measuring the similarity between the input colors and the representative features.

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

   where $\mathbf{F}_{\mathrm{r}} \in \mathbb{R}^{HW\times C}$ is from the input $\mathbf{X}$ using a stack of convolution layers

4) Therefore, the enhanced RGB values for each pixel in the input image are determined by combining the N transformed colors with attention weights.

   The enhanced image $\mathbf{Y} = \mathbf{A}\mathbf{T}^T$

The proposed RCT method offers advantages in terms of its capacity to cover color transformations and its ability to independently enhance each pixel, allowing for arbitrary input image sizes without resizing and preventing performance degradation.

- Representative Color Transform Network (RCTNet):

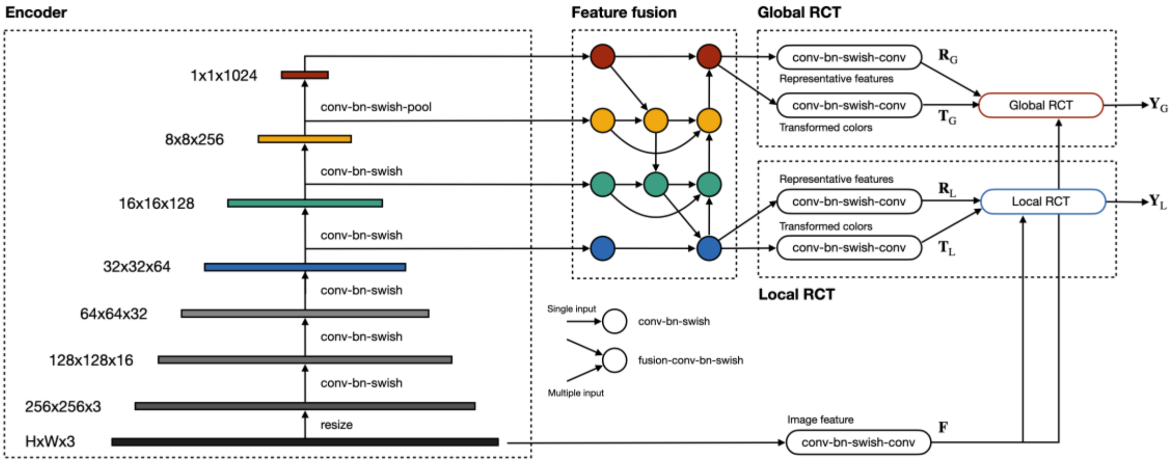
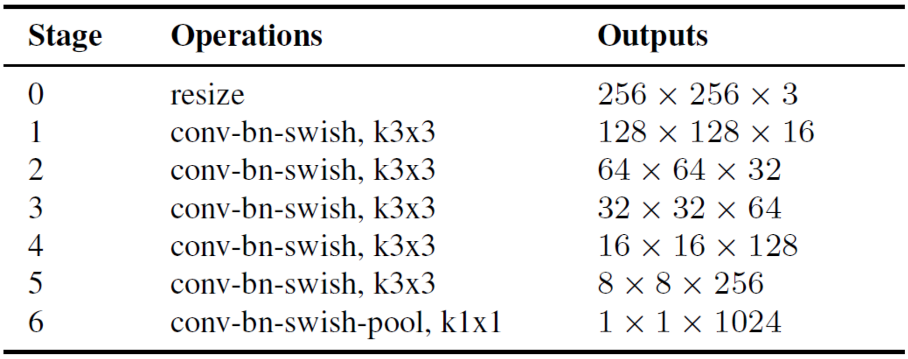


Figure 2: An overview of the proposed RCTNet.

The above image shows that the proposed RCTNet consists of four modules: encoder, feature fusion, global RCT, and local RCT. Given an input low-quality image X, RCTNet produces a high-quality image. Through 4 modules, we will obtain the outputs from global RCT and local RCT: Y_G and Y_L. As a result, our final output $\tilde{Y}$ combines the general and detailed information from the equation $\tilde{Y} = \alpha Y_G + \beta Y_L$ where α, β are the non-negative learnable weights to tune the suitable ratio between global and local. Next, we explain 4 modules in detail.

1) Encoder

The encoder in the proposed method is a convolutional neural network designed to extract high-level context information from the input image for image enhancement. The following table provides a detailed description of the encoder's 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	$1 \times 1 \times 1024$

The input image is resized to 256×256 and passed through the encoder, which consists of six 'conv-bn-swish' blocks. Each 'conv-bn-swish' block comprises a convolution layer, a batch normalization layer, and a swish activation layer. The convolution layers, except for the last block, have 3×3 filters. The last block differs in that it uses a convolution layer with a 1×1 filter and incorporates a global average pooling layer to extract a global feature vector.

The encoder extracts multi-scale feature maps from the last four blocks, which are then combined in the feature fusion module.

2) Feature fusion

The feature maps obtained from different scales provide distinct context information, with coarser-scale maps capturing global context and finer-scale maps preserving local details. To incorporate both global and local contexts for image enhancement, the feature fusion module employs bidirectional cross-scale connections. According to the RCT architecture figure, nodes with a single input represent the 'conv-bn-swish' blocks, while nodes with multiple inputs include a feature fusion layer before the 'conv-bn-swish' block to effectively combine multiple inputs.

When M inputs are provided to the feature fusion layer, the output of the feature fusion layer is defined as

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

where w_i is a non-negative learnable weight for the i-th input I_i and a small constant $\varepsilon = 0.0001$.

The nodes have 128 convolution filters of size 3×3, except for the coarsest-scale nodes (highlighted in red in Figure 2), which have 1×1 filters due to the spatial resolution being 1×1.

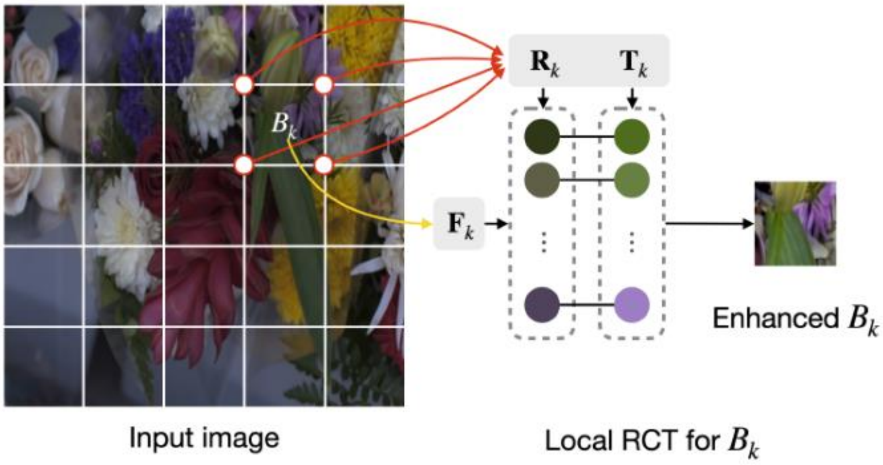
3) Global RCT

In the global RCT module, the coarsest-scale output feature $\mathbf{Z}_{\mathrm{G}} \in \mathbb{R}^{C'}$ is analyzed to extract the global context for image enhancement where $C'$is set to 128. This is done through two different 'conv-bn-swish-conv' blocks, resulting in representative features $\mathbf{R}_{\mathrm{G}} \in \mathbb{R}^{C\times N_{\mathrm{G}}}$ and transformed colors$\mathbf{T}_{\mathrm{G}} \in \mathbb{R}^{3\times N_{\mathrm{G}}}$. R_G and T_G are reshaped into 2D structures. The values of C and N_G are set to 16 and 64, respectively. The input image is transformed into the image feature F using another 'conv-bn-swish-conv' block, and F is reshaped into F_r. The global enhanced image Y_G is obtained by applying R_G, T_G, and F_r to the attention matrix $\mathbf{A} = \mathrm{softmax}(\frac{\mathbf{F}_{\mathrm{r}}\mathbf{R}}{\sqrt{C}}) \in \mathbb{R}^{HW\times N}$, and calculating $\mathbf{Y} = \mathbf{A}\mathbf{T}^T$.

4) Local RCT

The local RCT module aims to determine region-wise representative colors to capture local characteristics for image enhancement. It takes a feature map $\mathbf{Z}_\mathrm{L} \in \mathbb{R}^{32\times32\times C'}$, obtained from the finest-scale in the feature fusion module. Z_L is fed into two 'conv-bn-swish-conv' blocks, with the first convolution layers having 128 filters of size 3×3, and each second convolution layer having CN_L and $3N_L$ filters of size 3×3, respectively. The local RCT module generates representative feature sets $\mathbf{R}_\mathrm{L} \in \mathbb{R}^{32\times32\times C\times N_\mathrm{L}}$ and transformed color sets $\mathbf{T}_\mathrm{L} \in \mathbb{R}^{32\times32\times3\times N_\mathrm{L}}$, with N_L set to 16. $R_L(\mathrm{u},\mathrm{v})$ and $T_L(\mathrm{u},\mathrm{v})$ represent the sets of representative features and transformed colors, respectively, for each spatial position (u,v).

To assign representative features and transformed colors based on pixel coordinates, a uniform mesh grid of size 31×31 is placed on the input image, resulting in 32×32 corner points. The representative features and transformed colors at each corner point (u,v) are denoted as $R_L(\mathrm{u},\mathrm{v})$ and $T_L(\mathrm{u},\mathrm{v})$, respectively. The local RCT module performs grid-wise RCT, where each grid B_K is associated with four corner points and, consequently, four sets of representative features and transformed colors. The representative features R_K and transformed colors T_K for the grid B_K are obtained by concatenating the four sets of features at the corner points. From the image feature F, a grid feature F_K is extracted by cropping the corresponding grid region. Finally, using R_K, T_K, and F_K, enhanced colors are computed for B_K equations $\mathbf{A} = \mathrm{softmax}(\frac{\mathbf{F}_\mathrm{r}\mathbf{R}}{\sqrt{C}}) \in \mathbb{R}^{HW\times N}$ and $\mathbf{Y} = \mathbf{A}\mathbf{T}^T$. This process is repeated for all grids to obtain the locally enhanced image Y_L.



While the encoder requires fixed-size input for extracting multi-scale feature maps, both the global and local RCT modules enhance input images without resizing by extracting the image feature F without down-sampling.

- Loss Functions:

Consider a pair (X, Y), where X and Y are an input low-quality image and its high-quality image, the loss function between an enhanced image $\tilde{Y}$ and the ground-truth image Y is given by:

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

where the embedding function $\phi^{(k)}(.)$ is the output of kth layer in VGG-16, which is pre-trained on the ImageNet dataset.

4. Experimental results

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

(a) Compare the proposed algorithm with recent state-of-the-art in standard image enhancement.

(b) Evaluate the scalability of the proposed RCTNet on specific image enhancement problems: low-light image enhancement and underwater image enhancement.

(c) Analyze parameters and components of RCTNet through ablation studies on the MIT-Adobe 5K dataset.

- Datasets

(a) MIT-Adobe 5K: Consists of 5,000 images, decompose it to 4,500 and 500 images for training and testing set.

(b) Low Light (LoL): 500 pairs of low-light and normal-light images, separated into 485 training images and 15 testing images.

(c) Enhancing Underwater Visual Perception (EUVP) : 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.

(d) Underwater Image Enhancement Benchmark (UIEB): 890 pairs of underwater image and its enhanced image. 800 for training and 90 for testing image.

- Implementation Details

The authors trained 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 to minimize the loss function, with an initial learning rate of 5.0×10^{-4} and a weight decay of 1.0×10^{-5}. The authors decrease learning rate according to the cosine learning rate scheduling. Following the literature, we randomly crop images and then rotate them by multiples of 90 degrees for data augmentation. We fix the hyperparameter λ to 0.04.

- Comparison with state-of-the-arts

The following table shows best results are **boldfaced** and the second-best ones are underlined.

(a) MIT-Adobe 5K

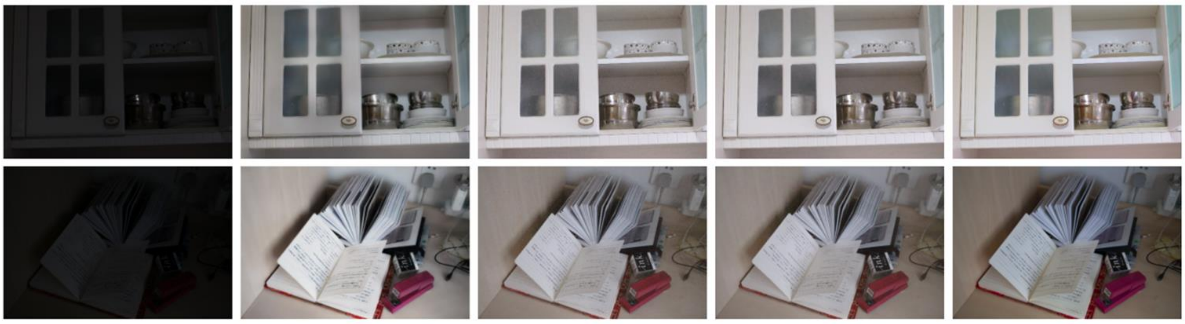
Method	**PSNR**	**SSIM**
HDRNET	23.44	0.882
DPE	23.34	0.873
DUPE	23.61	0.887
DLPF	24.48	0.887
3DLUT	25.21	0.922
GEN-LEN	25.88	**0.925**
RCTNet	26.02	0.915
RCTNet + BF*	**26.07**	0.923

*__BF__ is Bilateral Filter.

(b) Low Light (LoL)

Method	**PSNR**	**SSIM**
NPE	16.97	0.589
LIME	15.24	0.470
SRIE	17.34	0.686
RRM	17.34	0.686
SICE	19.40	0.690
DRD	16.77	0.559
KinD	20.87	0.802
DRBN	20.13	**0.830**
ZeroDCE*	14.86	0.559
EnlightenGAN*	15.34	0.528
RCTNet	22.67	0.788
RCTNet + BF	**22.81**	0.827

* The methods are trained with unpaired images.

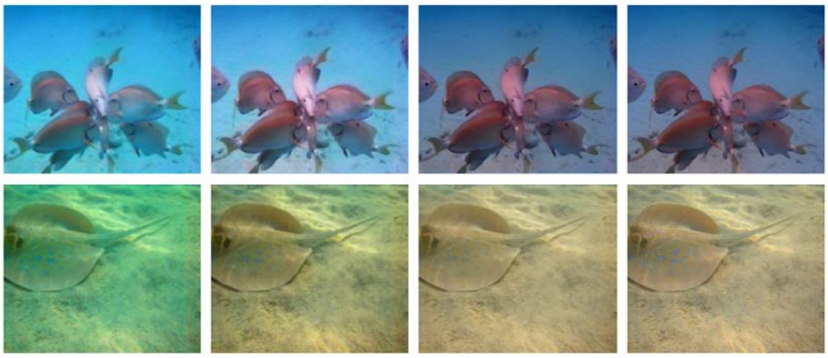


(a) Input (b) KinD [55] (c) RCTNet (d) RCTNet + BF (e) Ground-truth

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

(c) Enhancing Underwater Visual Perception (EUVP)

Method	**PSNR**	**SSIM**
U - GAN	23.49	0.842
Funie - GAN	23.40	0.827
Deep SESR	24.21	0.840
RCTNet	**26.43**	**0.915**



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

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

(d) Underwater Image Enhancement Benchmark (UIEB)

Method	**PSNR**	**SSIM**
Fushion	17.60	0.772
Retinex	17.02	0.607
GDCP	12.09	0.512
Histogram	15.82	0.539
Bluriness	15.32	0.603
Water CycleGAN	15.75	0.521
Dense GAN	17.28	0.443
WaterNet	19.11	0.797
Ucolor	20.63	0.770
RCTNet	**22.45**	**0.891**

- Ablation Studies

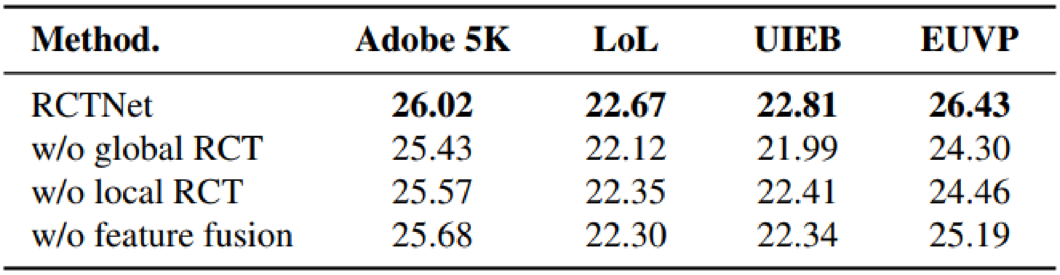
Component Analysis: The authors analyze the efficacy of the three components of feature fusion, global RCT, and local RCT modules in RCTNet.

In this test, there are three performances of RCTNet:

(a) without the global RCT

(b) without the local RCT

(c) without the feature of fusion



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

The performance of PSNR indicates that the proposed components are essential for image enhancement.

5. Conclusion:

The RCT determines different representative colors specialized in input images and enhances input images using representative features and transformed colors. Composed of encoder, feature fusion, global RCT, and local RCT modules.

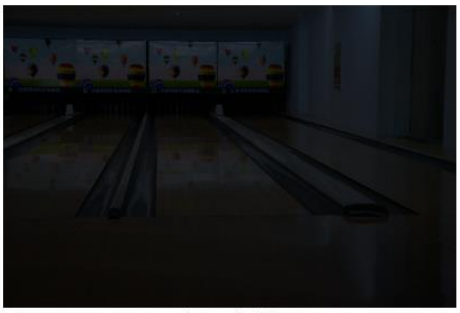
The global RCT predicts representative colors for an input image, while the local RCT considers local region characteristics for image enhancement.

RCTNet outperforms recent state-of-the-art algorithms on various datasets with standard image enhancement, low-light image enhancement, and underwater image enhancement.

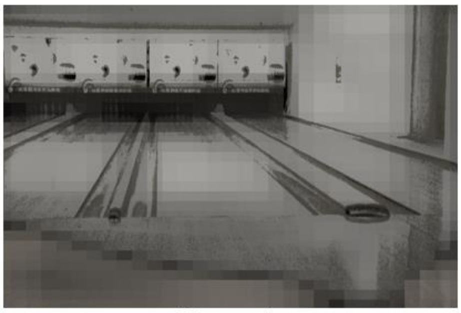
6. Our implementation:

We trained the model for only 20 epochs, which took a lot of time (1 hour per epoch), while the source code provide a pre-train model trained for 500 epochs.

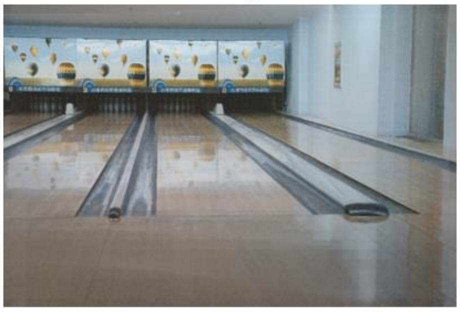
Experiments - LoL testing image (No.669)



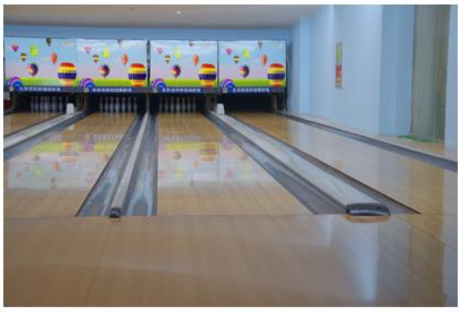
low-light



20 epochs

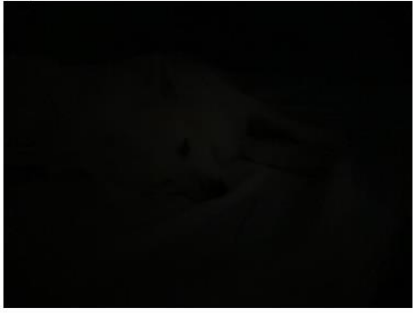


500 epochs

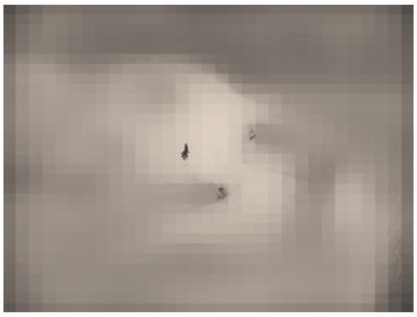


normal-light

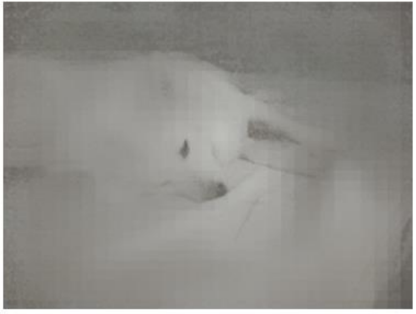
Experiments - 毛毛 image



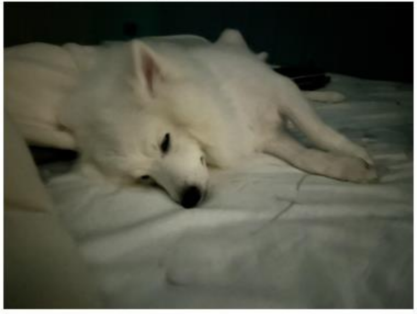
low-light



20 epochs



500 epochs



normal-light

*毛毛 is our partner 林映孝's pet.

The following table is the performance of we trained model and pre-trained model in the Low Light (LoL) data set.

Method	**PSNR**	**SSIM**
RCTNet - 20 epochs	16.05	0.599
RCTNet - 500 epochs	19.96	0.768
RCTNet – In the paper	22.67	0.788

7. Resource:

Paper resource (ICCV 2021)

Code resource (Github)