CLUT-Net: Learning Adaptively Compressed Representations of 3DLUTs for Lightweight Image Enhancement

Fengyi Zhang
School of Software Engineering, Tongji University
Shanghai, China
2131507@tongji.edu.cn

Hui Zeng
cshzeng@gmail.com
OPPO Research
Shenzhen, China

Tianjun Zhang
School of Software Engineering, Tongji University
Shanghai, China
1911036@tongji.edu.cn

Lin Zhang*
School of Software Engineering, Tongji University
Shanghai, China
cslinzhang@tongji.edu.cn

ABSTRACT

Learning-based image enhancement has made great progress recently, among which the 3-Dimensional LookUp Table (3DLUT) based methods achieve a good balance between enhancement performance and time-efficiency. Generally, the more basis 3DLUTs are used in such methods, the more application scenarios could be covered, and thus the stronger enhancement capability could be achieved. However, more 3DLUTs would also lead to the rapid growth of the parameter amount, since a single 3DLUT has as many as D^3 parameters where D is the table length. A large parameter amount not only hinders the practical application of the 3DLUT-based schemes but also gives rise to the training difficulty and does harm to the effectiveness of the basis 3DLUTs, leading to even worse performances with more utilized 3DLUTs. Through in-depth analysis of the inherent compressibility of 3DLUT, we propose an effective Compressed representation of 3-dimensional LookUp Table (CLUT) which maintains the powerful mapping capability of 3DLUT but with a significantly reduced parameter amount. Based on CLUT, we further construct a lightweight image enhancement network, namely CLUT-Net, in which image-adaptive and compression-adaptive CLUTs are learned in an end-to-end manner. Extensive experimental results on three benchmark datasets demonstrate that our proposed CLUT-Net outperforms the existing state-of-the-art image enhancement methods with orders of magnitude smaller parameter amounts. The source codes are available at https://github.com/Xian-Bei/CLUT-Net.

CCS CONCEPTS

• **Computing methodologies → Computational photography**.

KEYWORDS

Image enhancement, 3-dimensional lookup table (3DLUT), compressed representation

ACM Reference Format:
Fengyi Zhang, Hui Zeng, Tianjun Zhang, and Lin Zhang. 2022. CLUT-Net: Learning Adaptively Compressed Representations of 3DLUTs for Lightweight Image Enhancement . In *Proceedings of the 30th ACM International Conference on Multimedia (MM '22), October 10–14, 2022, Lisboa, Portugal.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3503161.3547879

1 INTRODUCTION

The digital imaging process suffers from various types of degradation such as adverse shooting environments and limited hardware conditions. Even without these distortions, the captured photos may still have unsatisfied perceptual quality. The traditional enhancement process adjusts the input image to meet human aesthetic requirements by photographers or expert-designed cascade modules, which are not only tedious and inefficient but also inflexible. Therefore, automatic and adaptive image enhancement methods are highly desired.

Benefiting from the rapid development of artificial intelligence technology, recent years have witnessed an increasing interest and a significant progress in the learning-based automatic enhancement methods [1, 2, 4, 10, 12, 13, 15, 19–21, 27]. Some of them [1, 4, 13, 27] directly transform the low-quality input image to the corresponding enhanced one using image-to-image networks, while others [2, 10, 12, 15, 19–21] construct the enhancement pipeline by combining powerful neural networks with manual-designed models to make full use of domain prior knowledge.

Among the latter category, Zeng *et al.* [31] proposed an adaptive 3-Dimensional LookUp Table (3DLUT) based approach with state-of-the-art (SOTA) overall performance. Specifically, it learns N 3DLUTs as a set of basis vectors of $\mathbb{R}^{3D^3}$ (D indicates the table length) and a lightweight convolutional neural network (CNN) simultaneously. The former aims to span a subspace that contains as many 3DLUTs as possible to cover various adjustment effects required in different scenes. The latter is intended to linearly combine the basis 3DLUTs to enhance the input image according to its characteristics. Since a larger set of effective basis vectors have the potential to span a larger subspace that could cover more types of enhancement effects, a larger N could naturally improve the enhancement capability.

[*] Corresponding author: Lin Zhang

MM '22, October 10–14, 2022, Lisboa, Portugal

ACM ISBN 978-1-4503-9203-7/22/10…$15.00
https://doi.org/10.1145/3503161.3547879

However, the 3DLUT-based methods enjoy powerful enhancement ability at the cost of a huge space complexity of $O(ND^3)$, which significantly hinders the practical application, especially for mobile devices with limited resources. Moreover, a large number of parameters gives rise to the training difficulty and does harm to the effectiveness of the learned basis 3DLUTs, making the performance even worse with the increase of N. Fig. 1 shows how the enhancement performance and the parameter amount (3DLUT part) of the 3DLUT-based methods change with N on MIT-Adobe FiveK [3] dataset. It can be seen that the PSNR (Peak Signal to Noise Ratio) increases with N when N is relatively small but quickly stagnates and even starts to decrease when N continues to grow, while the parameter amount keeps a steep growth rate.

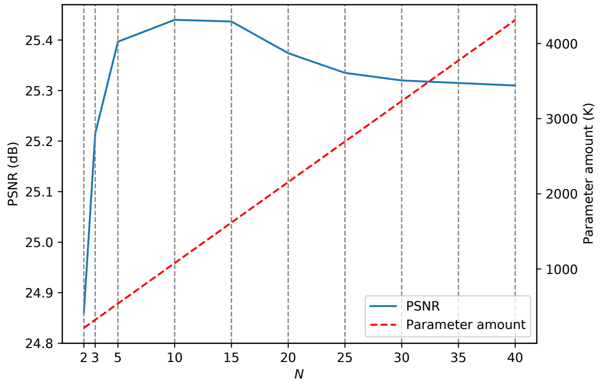

Figure 1: The trends of the PSNR and the parameter amount with the increase of N in the 3DLUT-based methods.

As an attempt to fill in the research gap to some extent, we conduct thorough analyses on the inherent property of 3DLUT and find that there are two kinds of correlations between the output of 3DLUT and the input of its three dimensions, according to which each dimension should be compressed in different manner and degree. Based on this observation, we propose an effective Compressed representation of 3-dimensional LookUp Table (CLUT) and build an efficient image enhancement network, namely CLUT-Net. The main contributions of this paper are summarized as follows:

- An effective compressed representation of 3DLUT, namely CLUT, is proposed based on in-depth analysis and exploration of its inherent compressibility. Compared with the standard 3DLUT, CLUT has orders of magnitude fewer parameters while maintaining the powerful mapping ability.
- A lightweight image enhancement network is built based on CLUT, namely CLUT-Net, in which adaptive CLUTs are learned end-to-end to strike an optimal balance between the enhancement performance and the parameter amount.
- Extensive qualitative and quantitative experiments are conducted on three benchmark datasets and the results show that our CLUT-Net outperforms SOTA image enhancement methods with a significantly reduced parameter amount.

2 RELATED WORK

In this section, we take a brief review of the learning-based image enhancement methods and then focus on a more specific field that our work belongs to, namely the 3DLUT-based enhancement methods. In addition, we introduce the relationship between our work and the general network compression.

2.1 Learning-based enhancement methods

Learning-based image enhancement methods could be traced back to 2011 when Bychkovsky *et al.* [3] collected the first large benchmark in this domain, namely MIT-Adobe FiveK dataset, and utilized handcraft features and shallow regression models to predict the adjustment strategies. The recent learning-based enhancement methods could be divided into two categories.

The methods in the first category [1, 4, 13, 27] treat image enhancement as an image translation task and utilize image-to-image networks to predict the enhancement results directly. Afifi *et al.* [1] and Kim *et al.* [13] both utilized UNet-like [24] networks to generate the enhanced images, while Chen *et al.* [4] utilized two-way GANs [8] that could be trained with unpaired retouched data. DeepUPE [27] learned to generate illumination maps with an encoder-decoder architecture to guide the enhancement process.

The methods in the other category [2, 7, 10, 12, 15, 19–21] combine neural networks with domain prior knowledge contained in various kinds of manual-designed models. Afifi *et al.* [2] and Liu *et al.* [19] trained networks to predict polynomial mapping functions, while Kim *et al.* [12], Moran *et al.* [21], and Li *et al.* [15] chose to predict one-dimensional rgb curves. HDRNet [7] learned pixel-wise transformation coefficients in the bilateral space. Moran *et al.* [20] learned three types of spatial filters to enhance the input image locally. He *et al.* [10] simulated some commonly used pixel-independent adjustments by multi-layer perceptrons (MLPs).

2.2 3DLUT-based enhancement methods

Zeng *et al.* [31] firstly proposed to combine 3DLUT with deep-learning techniques and constructed an adaptive 3DLUT-based image enhancement network that possesses powerful enhancement capability and high time efficiency. Following their strategy, some studies were conducted to utilize 3DLUT for other tasks and scenes. Liang *et al.* [18] applied 3DLUT to the portrait photo retouching (PPR) task by adjusting the learning strategy. Wang *et al.* [28] extended the 3DLUT-based methods to a spatial-aware version by learning pixel-level fusions of the basis 3DLUTs. Cong *et al.* [5] embedded a 3DLUT-based sub-module in their network for the high-resolution image harmonization task. To the best of our knowledge, all the follow-up researches extended the network that 3DLUT is embedded into, but none of them focus on improving the 3DLUT itself. In comparison, we further analyzed and utilized the inherent properties of 3DLUT and made fundamental improvements that could be simply integrated into any 3DLUT-based methods to significantly reduce the space complexity while maintaining or further improving the enhancement performance.

2.3 General network compression

There exists a wide range of studies that aim at the compression of general network architectures such as CNN and MLP. Among them [16, 17, 22] utilized the decomposition technique to represent high-dimension tensors as matrix multiplications to reduce parameter amount, [6] enforced sparsity constraints upon networks to reduce computation cost and improve accuracy. Differently, our work focuses on the compression of 3DLUT, a specific high-dimensional data structure with abundant domain prior knowledge and special intrinsic properties that could be leveraged to reduce the parameter amount while maintaining the mapping capability. It is worth mentioning that our work could be complementary and cooperate with the general network compression approaches, since the 3DLUT-based models consist of a feature extraction network and N basis 3DLUTs, which could be compressed by the general network compression technique and our scheme, respectively.

3 METHOD

In this section, we analyze the compressibility of 3DLUT in Sect. 3.1 and present in detail the proposed CLUT and CLUT-Net in Sect. 3.2. Then we give a thorough analysis of CLUT-Net in Sect. 3.3.

3.1 Compressibility of 3DLUT

Mathematically, standard 3DLUT is commonly represented by a three-dimensional three-channel array of $\mathbb{R}^{3\times D\times D\times D}$ denoted by $\boldsymbol{\phi}$. We split 3DLUT into three sub-tables corresponding to different channels and denote them by $[\boldsymbol{\phi}^c]_{c\in\{r,g,b\}}$ where $\boldsymbol{\phi}^c \in \mathbb{R}^{D\times D\times D}$. 3DLUT discretizes each dimension of the RGB color space into D bins, resulting in D^3 discrete colors denoted by $\{(i,j,k)\}_{i,j,k=1,...,D}$, and stores the corresponding mapped color of each of them as $(\boldsymbol{\phi}^r_{[i,j,k]}, \boldsymbol{\phi}^g_{[i,j,k]}, \boldsymbol{\phi}^b_{[i,j,k]})$, where $\boldsymbol{\phi}^c_{[i,j,k]}$ represents the element with $[i,j,k]$ coordinate in $\boldsymbol{\phi}^c$. Basically, the mapping process of 3DLUT consists of a lookup operation and a trilinear interpolation operation, where the former finds the surrounding elements of the input color in the table and the latter fuses them into an output color.

Although 3DLUT enjoys a high time efficiency as it could process all the pixels in parallel, it suffers a large space complexity since its parameter amount grows cubically with D, resulting in the rapid growth of the parameter amount of the 3DLUT-based methods. A natural way to compress 3DLUT is to utilize a smaller value for the hyper-parameter D, which is equivalent to directly reducing the number of discrete bins of r, g, and b dimensions by the same proportion simultaneously. However, the number of discrete bins determines the color mapping precision of 3DLUT on the corresponding channel. Therefore, such a naive compression approach would lose the mapping precision of 3DLUT on all three channels to some degree and even lead to an unacceptable degradation of the enhancement performance, which will be verified through ablation studies in Sect. 4.2.

Actually, we find that in each $\boldsymbol{\phi}^c$, the input values of different dimensions have different impacts on the output values. Therefore, each dimension of each $\boldsymbol{\phi}^c$ has its most suitable compression approach and degree depending on its corresponding impact, and it is not an effective compression scheme to directly reduce the bin number of each dimension to the same degree. Specifically, we use D_r, D_g, D_b to denote the bin numbers of different dimensions of 3DLUT corresponding to channel r, g, and b, respectively, instead of using D for all of them. In consequence, 3DLUT could be represented as $[\boldsymbol{\phi}^c]_{c\in\{r,g,b\}}$ where $\boldsymbol{\phi}^c \in \mathbb{R}^{D_r\times D_g\times D_b}$. Given a specific color channel c where $c \in \{r,g,b\}$ and the other two channels denoted by x and y, we find that the output value c_{out} of $\boldsymbol{\phi}^c$ is strongly correlated to the input value c_{in} of channel c while weakly correlated to the input values x_{in}, y_{in} of channel x and y, respectively. To demonstrate our observation, we visualize the strong correlations between c_{out} and c_{in} and weak correlations between c_{out} and (x_{in}, y_{in}) of five 3DLUTs in Fig. 2 and Fig. 3, respectively, where these 3DLUTs are learned on MIT-Adobe FiveK dataset [3] under the setting of [31] with $D = 33$ and $N = 5$. Each column in these two figures corresponds to one of the five 3DLUTs and each row visualizes $\boldsymbol{\phi}^r$, $\boldsymbol{\phi}^g$, and $\boldsymbol{\phi}^b$ from top to bottom, respectively. Obviously, no matter which color channel c indicates, c_{out} is mainly determined by c_{in} but only fluctuates slightly with x_{in} and y_{in}. Therefore, dimensions corresponding to channel c, x, and y of each $\boldsymbol{\phi}^c$ should be precisely compressed in their most suitable manner and degree.

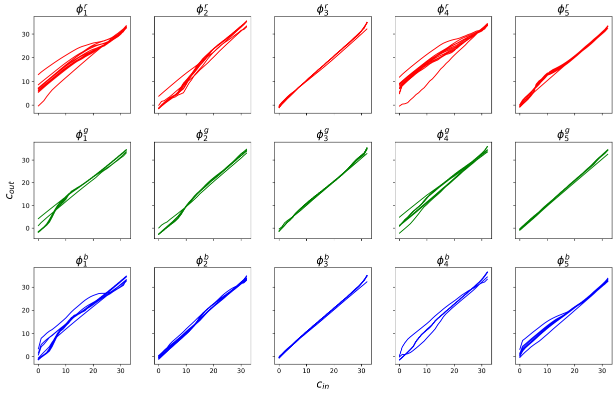

Figure 2: Visualization of the strong correlations between c_{out} and c_{in} given (x_{in}, y_{in}) in five 3DLUTs. The horizontal axis represents c_{in} and the longitudinal axis represents c_{out}. Each $\boldsymbol{\phi}^c \in \mathbb{R}^{D\times D\times D}$ consists of D^2 strong correlations (i.e., $(c_{in}, c_{out}) = \{(0,0), (0,1), ..., (D-1, D-1)\}$), and we only visualize ten of them for clear and intuitive observation.

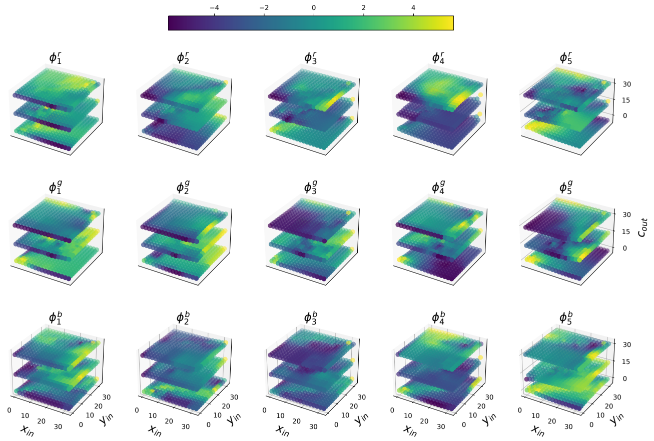

Figure 3: Visualization of the weak correlations between c_{out} and (x_{in}, y_{in}) given c_{in} in five 3DLUTs. The horizontal axis represents (x_{in}, y_{in}) and the longitudinal axis represents c_{out}. Each $\boldsymbol{\phi}^c \in \mathbb{R}^{D\times D\times D}$ consists of D (i.e.,$c_{in} = \{0, 1, ..., D-1\}$) weak correlations in total, and we only visualize three of them (i.e., c_{in}={0, 15, 30}) for clear and intuitive observation.

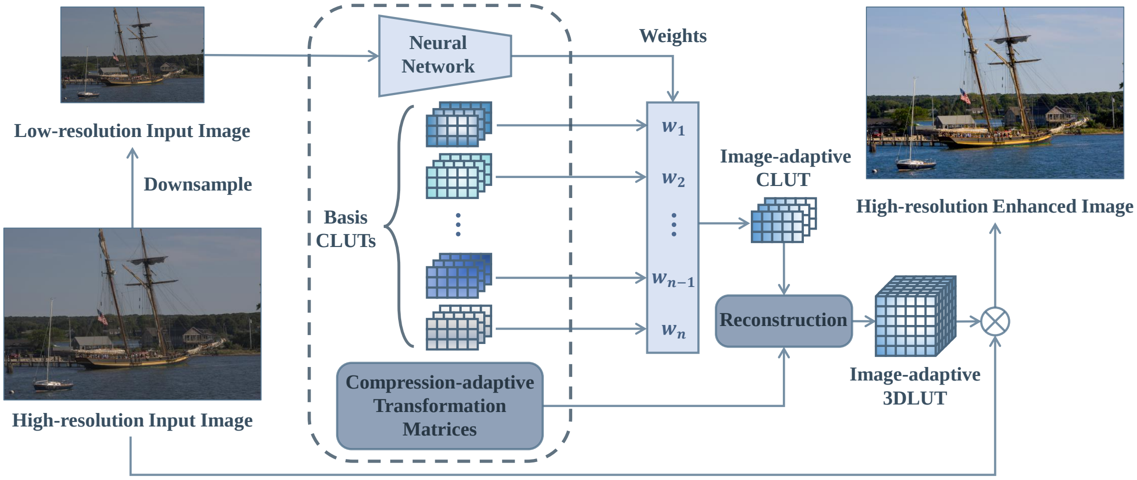
Low-resolution Input Image
Downsample
High-resolution Input Image
Neural Network
Weights
Basis CLUTs
w_1
w_2
w_{n-1}
w_n
Compression-adaptive Transformation Matrices
Image-adaptive CLUT
Reconstruction
Image-adaptive 3DLUT
High-resolution Enhanced Image

Figure 4: Framework of our proposed CLUT-Net which consists of a neural network, N basis CLUTs and two transformation matrices. The N basis CLUTs cover various enhancement effects required in different scenes. The neural network predicts content-dependent weights according to the downsampled input to fuse the basis CLUTs into an image-adaptive one, from which the transformation matrices adaptively reconstruct the corresponding standard 3DLUT to enhance the original input image. All three modules are jointly learned from the annotated data in an end-to-end manner.

3.2 Learning adaptively compressed representations of 3DLUTs

Based on the analysis in the previous subsection, we propose an effective compressed representation of 3DLUT, namely CLUT, denoted by $\boldsymbol{\psi} = [\boldsymbol{\psi}^c]_{c\in\{r,g,b\}} \in \mathbb{R}^{3\times S\times W}$. Compared with the standard form of $\boldsymbol{\phi} = [\boldsymbol{\phi}^c]_{c\in\{r,g,b\}} \in \mathbb{R}^{3\times D\times D\times D}$, CLUT introduces two hyper-parameters S and W instead of one parameter D to precisely control the compression degree of different dimensions. In addition, each sub-table $\boldsymbol{\phi}^c$ of 3DLUT is compressed to $\boldsymbol{\psi}^c$ of CLUT by reducing D_c to S and $D_x \times D_y$ to W through linear spatial transformation, in which two utilized transformation matrices denoted by $\boldsymbol{M} = \{\boldsymbol{M}_s \in \mathbb{R}^{D\times S}, \boldsymbol{M}_w \in \mathbb{R}^{W\times D^2}\}$ serve as the suitable compression schemes for different dimensions. The reconstruction process between CLUT $\boldsymbol{\psi}$ and corresponding 3DLUT $\boldsymbol{\phi}$ is conducted as,

$$\boldsymbol{\phi} = f(\boldsymbol{\psi}, \boldsymbol{M}) = h([\boldsymbol{M}_s\boldsymbol{\psi}^r\boldsymbol{M}_w,\ \boldsymbol{M}_s\boldsymbol{\psi}^g\boldsymbol{M}_w,\ \boldsymbol{M}_s\boldsymbol{\psi}^b\boldsymbol{M}_w]), \quad (1)$$

where h denotes a simple reshape procedure from $\mathbb{R}^{3\times D\times D^2}$ to $\mathbb{R}^{3\times D\times D\times D}$. Notice that D_x and D_y are collectively reduced since there is no obvious difference between the impacts of dimensions corresponding to channel x and channel y on $\boldsymbol{\phi}^c$.

Based on CLUT, we construct CLUT-Net which learns adaptive CLUTs for lightweight image enhancement. Fig. 4 illustrates the overall architecture of our proposed CLUT-Net which consists of three modules as circled by the dotted line, namely a neural network G, N basis CLUTs $\{\boldsymbol{\psi}_n\}_{n=1,\dots,N}$ and two transformation matrices $\boldsymbol{M} = \{\boldsymbol{M}_s, \boldsymbol{M}_w\}$. The basis CLUTs and G act the same roles as the basis 3DLUTs and the neural network in the standard 3DLUT-based methods, respectively. The former aims to cover various enhancement effects required in different scenes but with orders of magnitude fewer parameters compared with the standard 3DLUTs, while the latter is intended to extract features from the input image and predict weights to fuse the basis CLUTs into an image-adaptive one. It is worth mentioning that following the common practice [12, 18, 19, 28, 31], G only needs to work on the downsampled input image to decide how to fuse the basis operators, since a relatively low-resolution version contains enough global context information and could significantly save the computational cost. $\boldsymbol{M}_s$ and $\boldsymbol{M}_w$ are learned to adaptively reconstruct the corresponding standard 3DLUT from the image-adaptive CLUT to enhance the original input image. The hyper-parameters N, S, and W are set to 20, 5, and 20 in our implementation, respectively, through ablation studies. All three modules of CLUT-Net are jointly learned from the annotated data in an end-to-end manner.

Specifically, given an input image $\boldsymbol{I}$, G predicts the content-dependent weights as $G(\tilde{\boldsymbol{I}}) = \{w_n\}_{n=1,\dots,N}$ where $\tilde{\boldsymbol{I}}$ is the downsampled version of $\boldsymbol{I}$. Then the basis CLUTs are linearly combined into an image-adaptive one as,

$$\boldsymbol{\psi} = \sum_{n=1}^{N} w_n\boldsymbol{\psi}_n. \quad (2)$$

After that, the image-adaptive 3DLUT is reconstructed and the corresponding enhanced result O is generated as,

$$O = \boldsymbol{\phi}(\boldsymbol{I}) = f(\boldsymbol{\psi}, \boldsymbol{M})(\boldsymbol{I}), \quad (3)$$

where f is presented in Eq. 1. Given a training set with P pairs of input and target images denoted by $\{(\boldsymbol{I}_p, \boldsymbol{T}_p)\}_{p=1,\dots,P}$, the target of the training stage can be formulated as,

$$\underset{(G,\boldsymbol{M},\{\boldsymbol{\psi}_n\})}{\arg\min} \sum_{p=1}^{P} \mathcal{L}(\boldsymbol{T}_p, \boldsymbol{O}_p), \quad (4)$$

where $\mathcal{L}$ denotes the loss function which in our implementation is defined as,

$$\mathcal{L} = \|\boldsymbol{T}_p - \boldsymbol{O}_p\|_1 + cos(\boldsymbol{T}_p, \boldsymbol{O}_p) + \lambda_s\mathcal{R}_s + \lambda_m\mathcal{R}_m. \quad (5)$$

The former two terms are the L_1 distance and the cosine distance between $\boldsymbol{T}_p$ and $\boldsymbol{O}_p$, respectively. The cosine distance is empirically found to be conducive to achieve smaller accuracy variances and faster convergence. The latter two terms are the smooth and monotonicity regularization constraints proposed by [31], respectively. λ_s and λ_m are the hyper-parameters that control the regularization strength and were set to $1e-4$ and 10, respectively, according to the ablation studies of [31]. Such a combination of loss functions was applied to all the 3DLUT-based methods for fair comparisons.

In CLUT-Net, the neural network module plays the role of extracting features and predicting weights from input images. Theoretically, any backbone network architecture with image features extraction capabilities such as various kinds of CNN and Transformer [26] could be integrated into our enhancement pipeline. On consideration of space and time efficiencies, we utilized a lightweight CNN with only about 264 K parameters in total when $N = 20$, which is similar to the one used in [31]. Specifically, it consists of five convolutional blocks with leaky ReLU [30] and instance normalization [25], one dropout layer, a global average pooling layer, and a hardswish classifier module proposed by [11].

3.3 Discussion and analysis

3.3.1 Image-adaptive and Compression-adaptive properties. CLUT-Net learns adaptive CLUTs end-to-end for lightweight image enhancement. It is worth noting that compared with the image-adaptive 3DLUT-based methods, the implications of adaptability in our scheme are extended into two parts. Image-adaptive: In the training process, the basis CLUTs are directly learned from the annotated data, and thus they are adaptive to the training images. In the testing stage, the neural network predicts content-dependent weights to fuse the basis 3DLUTs, and thus the enhancement effect is adaptive to the test image. Compression-adaptive: The basis CLUTs and the transformation matrices which control the compression and reconstruction processes are jointly learned so they could cooperate well with each other to achieve an optimal balance between the enhancement capability and the parameter amount. We verify the effectiveness of our adaptive compression scheme by comparing it with non-adaptive ones in the ablation study.

3.3.2 Space and time complexities. The main improvement of CLUT-Net compared with the standard 3DLUT-based methods lies in the process of generating the image-adaptive 3DLUT. In this process, the space complexities of CLUT-Net and standard 3DLUT-based methods are $O(NSW + DS + WD^2)$ and $O(ND^3)$, respectively. Obviously, although an extra cost $O(DS + WD^2)$ of two matrices is introduced, it is irrelevant to N. Only the parameter amount of the basis CLUTs part $O(NSW)$ increases with N at a very low speed since $S \ll D$ and $W \ll D^2$ in our implementation. Overall, CLUT-Net significantly reduces the space complexity of the standard 3DLUT-based methods. As an example, under the setting of $D = 33$ and $N = 20$, the parameter amounts of CLUT-Net and the standard 3DLUT-based methods are about 28 K and 2,156 K, respectively, where the former is only about 1.3% of the latter. Similarly, CLUT-Net reduces the $O(ND^3)$ time complexity (measured by numbers of float multiply-add operations) of the standard 3DLUT-based methods to $O(NSW + SD^3 + SWD^2)$, where an extra cost of two matrix multiplication operations is introduced but the growth rate with N is reduced. Under the same setting, the numbers of float multiply-add operations of CLUT-Net and the standard 3DLUT-based methods are about 872 K and 2,156 K, respectively, where the former is only about 40.4% of the latter.

4 EXPERIMENTS

4.1 Experimental setup

Experiments were conducted on three benchmark datasets, namely MIT-Adobe FiveK [3], HDR+ [9] and PPR10K [18]. The MIT-Adobe FiveK dataset is one of the most widely used benchmarks for image enhancement task proposed by [3]. Following the common practice [4, 28, 31], we experimented with the expert-C version groundtruth and adopted the same split with 4,500 training image pairs and 500 testing ones.

The HDR+ dataset is a burst photography dataset proposed by the Google camera group in [9] for research of high dynamic range and low-light imaging on mobile cameras. Following the practice of [28, 31], we utilized the intermediate results of the aligned and merged frames as the input, and the images generated by the manually tuned HDR imaging pipeline as the groundtruth. The same split with 675 training image pairs and 250 testing ones was adopted.

The PPR10K dataset is a large-scale benchmark collected by [18] for the study of automatic portrait photo retouching (PPR). Following the strategy of [18], we experimented with all three versions of groundtruth and both the low-resolution and original high-resolution settings, which are denoted by PPR-a, PPR-b, PPR-c, LR, and HR, respectively. The same split with 8,875 training images and 2,286 testing ones was adopted.

Three most commonly used metrics, namely PSNR, SSIM [29], and ΔE, were employed to quantitatively evaluate the enhancement performance on the aforementioned datasets. The PSNR and ΔE are defined based on the L_2 distance in RGB color space and CIELAB color space, respectively. Notice that higher PSNR and SSIM and lower ΔE indicate better enhancement performance.

Our CLUT-Net was implemented based on PyTorch [23], and the deployment of 3DLUT was implemented via the CUDA parallel code released by [31]. All the experiments were conducted on Titan RTX GPUs. An Adam [14] optimizer with default setting except for a learning rate of $1e-4$ was applied for training. Same to [18, 28, 31], a classical setting of $D = 33$ was adopted in CLUT-Net.

4.2 Ablation study

In this subsection, ablation studies were conducted on MIT-Adobe FiveK [3] to investigate the selection of hyper-parameters and verify the efficacy of the proposed adaptive compression mechanism.

4.2.1 Selection of hyper-parameters. As aforementioned, there are three important hyper-parameters in our proposed CLUT-Net, namely N, S, and W, where the first one determines the number of the basis CLUTs and the latter two take control of the compression degrees of different dimensions. To quantitatively demonstrate their impact on the overall performance and determine the most suitable settings for them, we first evaluated with $S = \{2, 3, 5, 7, 10, 15, 25\}$ without reducing $D_x \times D_y$ to W, and repeated these experiments for $N = \{3, 5, 10, 20, 30\}$ to determine the optimal setting of N and S. Then we evaluated with $W = \{5, 7, 10, 15, 20, 25, 30, 40, 50, 70, 100\}$

under the optimal setting of N and S to determine that of W. Each experiment was repeated three times to avoid the influence of randomness on the results.

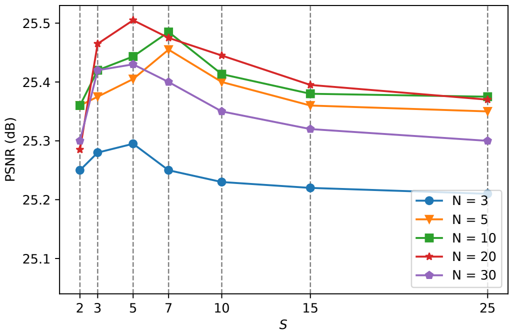

Figure 5: Effects of hyper-parameter N and S on the enhancement performance of CLUT-Net.

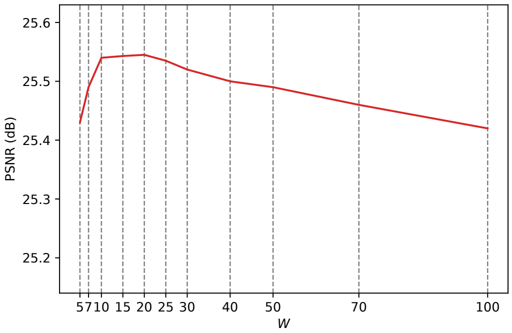

Figure 6: Effects of hyper-parameter W on the enhancement performance of CLUT-Net when $N = 20$ and $S = 5$.

As shown in Fig. 5, although there are some local fluctuations in the line charts, the overall tendency is still obvious. Under the same setting of N, the PSNR first increases and then decreases with the growth of S. The separating points between the rising phase and falling phase are about $5 \sim 7$. In addition, under the optimal setting of S for each N, the enhancement performance improves with N from 1 to 20 and then starts to decrease when $N \geq 20$. Thus, we adopted 20 and 5 as the optimal values for N and S, respectively. As shown in Fig. 6, under the setting of $N = 20$ and $S = 5$, the PSNR increases with W when $W < 20$ and then starts to decrease slowly. Overall, we set $N = 20$, $S = 5$, and $W = 20$ in our implementation of CLUT-Net for a good balance between the capability and the complexity. Notice that as a compressed representation of 3DLUT, CLUT could have S ranging between $[1, 33)$ and W ranging between $[1, 1089)$. The experimental results demonstrate that small values like $S = 5$ and $W = 20$ could achieve higher enhancement performance than the standard 3DLUT-based methods, which fully verifies the huge compressibility of 3DLUT and the effectiveness of CLUT and CLUT-Net.

4.2.2 *Effectiveness of adaptive compression.*

To demonstrate the importance of the compression mechanism to the enhancement capability and verify the effectiveness of our adaptive compression scheme which learns the transformation matrices and the compressed representations jointly, we compared CLUT-Net with three baselines using different compression mechanisms. In this ablation study, we set $N = 20$ for all the experiments. These baselines are presented as follows: **BL-*A*)** standard 3DLUT-based method with smaller D; **BL-*B*)** standard 3DLUT-based method with non-learnable offline PCA compression; **BL-*C*)** Non-learnable CLUT-Net: basis CLUTs with fixed non-learnable transformation matrices.

Specifically, for **BL-*A***, we learned twenty standard basis 3DLUTs with $D = \{7, 9, 11, 13, 15, 17, 21, 25, 33\}$. A 3DLUT with $D < 33$ could be seen as a compressed representation of that with $D = 33$ constructed by linear interpolation. For **BL-*B***, we compressed the twenty standard basis 3DLUTs with $D = 33$ learned in **BL-*A*** through offline PCA decomposition. We present this process by taking the reduction from $D_x \times D_y$ to W as an example. Specifically, all the twenty basis 3DLUTs were firstly permuted and reshaped into $20 \times 3 \times 33 = 1980$ vectors of $\mathbb{R}^{33^2=1089}$ which represent the weak correlations between c_{out} and (x_{in}, y_{in}). Then PCA was applied on these vectors to calculate W basis vectors of $\mathbb{R}^{1089}$, which form the transformation matrices $\boldsymbol{M}_w \in \mathbb{R}^{W \times 1089}$, and the compressed representation with W parameters for each vector, which form the compressed representations ($\mathbb{R}^{3 \times D \times W}$) of the twenty 3DLUTs. For **BL-*C***, we directly fixed the transformation matrices calculated by offline PCA decomposition in **BL-*B*** and then learned the basis CLUTs end-to-end, which can be seen as learning to build the basis 3DLUTs by selecting and combining base vectors from given sets. Notice that **BL-*C*** possesses image-adaptive property but not compression-adaptive property. We experimented with $W = \{10, 20, 30, 60, 120, 200\}$ and D_c uncompressed for **BL-*B*** and **BL-*C***. We reported the performance of CLUT-Net near the optimal setting ($S = 5$ and $W = 20$) to show its overall superiority.

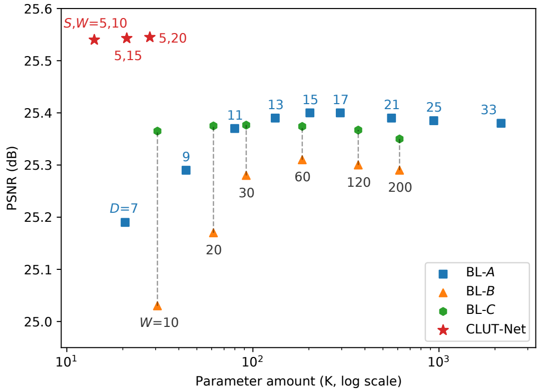

Figure 7: Quantitative comparison of the enhancement performance of different baselines and CLUT-Net.

The quantitative comparisons of the three baselines and CLUT-Net are presented in Fig. 7. To facilitate the comparison, the performance of **BL-*B*** and **BL-*C*** with the same transformation matrices are connected with gray dotted lines. Obviously, simply reducing D in **BL-*A*** could save the parameter amount to a small extent, but a large-scale compression in this way will inevitably lead to unacceptable performance degradation, due to the loss of the color

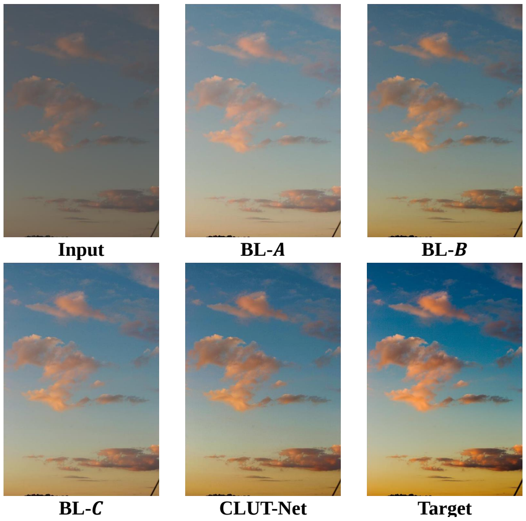

Figure 8: Qualitative comparison of the enhancement results of different baselines and CLUT-Net.

mapping precision of 3DLUT on all three channels. These experimental results verify our analysis in Sect. 3.1 that it is not an effective compression scheme to directly reduce the bin number of each dimension to the same degree. Compared with the standard 3DLUT-based approach with $D = 33$, offline PCA compression in **BL-*B*** reduces the parameter amount at the cost of the largest loss of enhancement performance among all baselines, while **BL-*C*** has the same transformation matrices as **BL-*B*** but performs much better than it, indicating the importance of the online learning strategy. However, **BL-*C*** learns CLUTs in an end-to-end manner but with non-learnable transformation matrices, making it perform worse than CLUT-Net which learns CLUTs and the transformation matrices jointly.

As shown in Fig. 8, **BL-*A*** failed to achieve a vivid enhancement effect with sufficient saturation and contrast due to the severe loss of the color mapping precision. **BL-*B*** generated enhancement result with distorted visual quality since as a post-processing compression approach, the PCA decomposition only aims at the maximum reconstruction of the basis 3DLUTs but without considering the image enhancement effects. In comparison, both **BL-*C*** and our CLUT-Net generated visual-pleasing enhancement results, where CLUT-Net apparently outperformed the former approach on the overall perceptual quality. In conclusion, compared with all the other baselines, our CLUT-Net achieved the highest PSNR with the maximum compression degree and generated the most visual-pleasing enhancement result, which fully verifies the effectiveness of our adaptive compression scheme.

Table 1: Quantitative comparison on MIT-Adobe FiveK [3] and HDR+ [9] datasets. For the 3DLUT-based methods, *N* is shown after the name, and the parameter amount is composed of the neural network part and the basis 3DLUTs part.

Dataset	Method	PSNR ↑	SSIM ↑	ΔE ↓	Param. (K)
FiveK	UPE [27]	21.88	0.853	10.80	999
	DPE [4]	23.75	0.908	9.34	5,750
	HDRNet [7]	24.32	0.912	8.49	482
	CSRNet [10]	25.23	0.923	7.70	**37**
	LUT-3 [31]	25.23	0.912	7.61	269+323
	LUT-20 [31]	25.38	0.922	**7.46**	269+2,156
	sLUT-30 [28]	25.40	0.925	**7.46**	921+3,234
	CLUT-20	**25.55**	**0.927**	**7.46**	264+28
HDR+	UPE [27]	21.21	0.816	13.05	999
	DPE [4]	22.56	0.872	10.45	5,750
	HDRNet [7]	23.04	0.879	8.97	482
	CSRNet [10]	23.32	0.888	8.51	**37**
	LUT-3 [31]	23.54	0.885	7.93	269+323
	LUT-20 [31]	23.91	0.891	7.67	269+2,156
	sLUT-30 [28]	26.94	0.927	**6.04**	921+3,234
	sCLUT-30	**26.98**	**0.928**	**6.04**	921+31

Table 2: Quantitative comparison on PPR10K [18] datasets. For the 3DLUT-based methods, *N* is shown after the name, and the parameter amount is composed of the neural network part and the basis 3DLUTs part.

Dataset	Method	PSNR↑		ΔE ↓		Param. (K)
		LR	HR	LR	HR	
PPR-a	HDRNet [7]	23.93	23.06	8.70	9.13	482
	CSRNet [10]	22.72	22.01	9.75	10.20	**37**
	sLUT-30 [28]	25.85	25.39	6.84	7.11	921+3,234
	LUT-HRP-5 [31]	25.99	25.55	6.76	7.02	11,177+539
	CLUT-HRP-5	**26.11**	**25.69**	**6.68**	**6.95**	294+28
PPR-b	HDRNet [7]	23.96	23.51	8.84	9.13	482
	CSRNet [10]	23.76	23.29	8.77	9.28	**37**
	sLUT-30 [28]	25.01	24.54	7.67	7.88	921+3,234
	LUT-HRP-5 [31]	25.06	24.66	7.51	7.73	11,177+539
	CLUT-HRP-5	**25.22**	**24.84**	**7.49**	**7.70**	294+28
PPR-c	HDRNet [7]	24.08	23.66	8.87	9.05	482
	CSRNet [10]	23.17	22.85	9.45	9.87	**37**
	sLUT-30 [28]	25.36	24.85	7.54	7.78	921+3,234
	LUT-HRP-5 [31]	25.46	25.05	7.43	7.69	11,177+539
	CLUT-HRP-5	**25.62**	**25.21**	**7.41**	**7.65**	294+28

4.3 Comparison with SOTA

We compared our proposed CLUT-Net with several SOTA learning-based image enhancement methods including UPE [27], DPE [4], HDRNet [7], CSRNet [10], 3DLUT [31] and spatial-aware 3DLUT [28] under the aforementioned experimental settings. For simplicity, 3DLUT [31], spatial-aware 3DLUT [28] and CLUT-Net are denoted by LUT, sLUT and CLUT, respectively. Notice that in the PPR task the resolution of the HR human portrait is very high (ranging from 4K to 8K), which hinders the applications of some compared methods because of their heavy computational and memory costs. Thus following the practice of [18], we evaluated HDRNet [7], CSRNet [10], LUT [31], sLUT [28] and our CLUT on PPR10K [18] dataset. As a fundamental improvement, CLUT-Net could be simply integrated with the standard 3DLUT-based methods to boost their space efficiency. To verify this capability, we integrated our schemes with [28] and [18]. The consequential two CLUT-based methods are denoted by sCLUT and CLUT-HRP and were evaluated on HDR+ [9] and PPR10K [18] datasets, respectively.

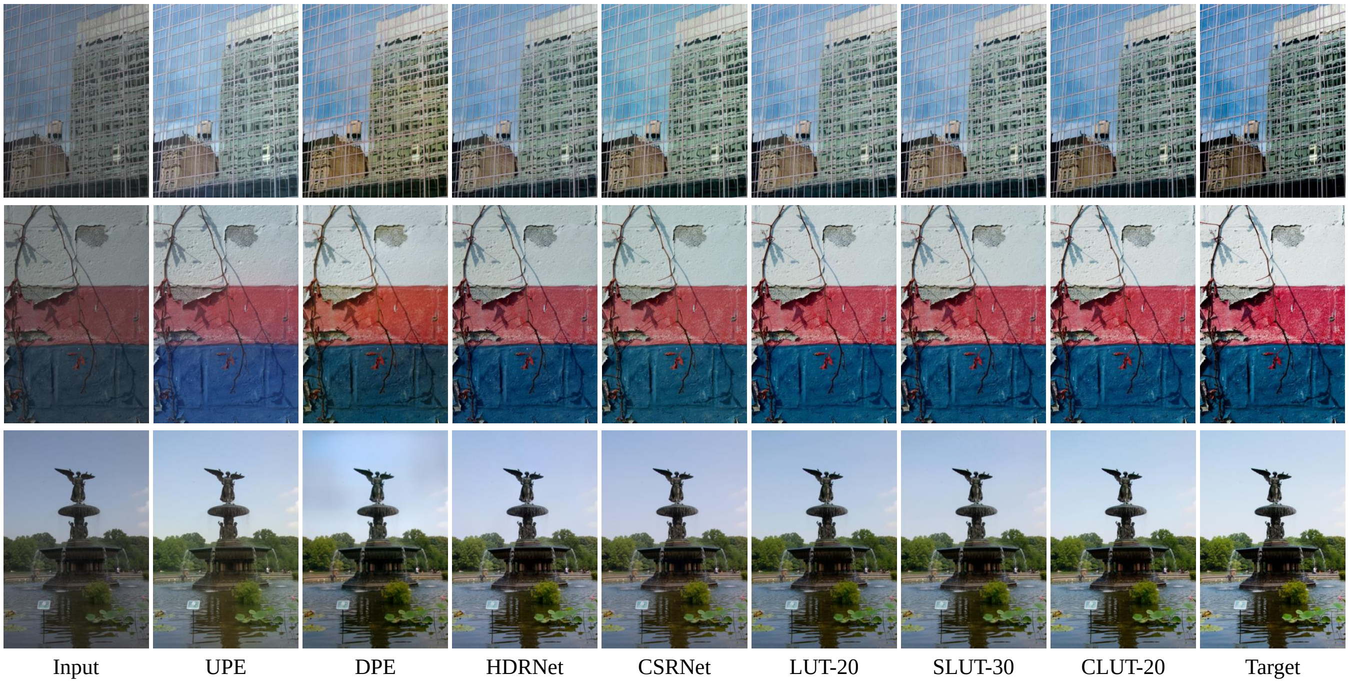

Figure 9: Qualitative comparison with SOTA methods on MIT-Adobe FiveK [3] dataset.

The quantitative experimental results and the parameter amounts are reported in Table 1 and Table 2. As highlighted in bold, CLUT-based methods achieved the best performance on each dataset and metric. It can be seen that by integrating with CLUT-Net, the parameter amounts of the corresponding 3DLUT-based methods are significantly reduced, especially when N is relatively large. For example, sLUT [28] utilized thirty basis 3DLUTs and improved the performance significantly on HDR+ [9] dataset. By integrating with CLUT-Net, the parameter amount of the basis 3DLUTs part is reduced from 3,234 K to 31 K, which is a more than 99% compression rate, while the enhancement performance is further improved.

In addition, our CLUT-Net inherits the superior time efficiency of the standard 3DLUT-based methods, and both [31] and ours could enhance an input image of 1920 × 1080 resolution in less than 0.7 ms on a single Titan RTX GPU. In comparison, under the same device and resolution settings, the time cost of UPE [27], DPE [4], HDRNet [7] and CSRNet [10] are 45 ms, 86 ms, 45 ms and 6 ms, respectively, which are orders slower than [31] and ours. Among all evaluated approaches, although CSRNet [10] has fewer parameters than ours, CLUT-Net outperforms it on each dataset in terms of all three metrics and speed by a large margin.

The qualitative comparison of the enhancement effects is presented in Fig. 9. It can be seen that the enhancement results of UPE [27], DPE [4], HDRNet [7] and CSRNet [10] suffered from unpleasing color deviations to some extent. In comparison, the 3DLUT-based methods generated relatively stable enhancement results, among which CLUT-Net achieved the most vivid effects in terms of hue, saturation, and contrast. Overall, by learning the adaptive compressed representations of the basis 3DLUTs, CLUT-Net not only improved the quantitative performance of the standard 3DLUT-based methods with a much smaller parameter amount but also generated enhancement effects with higher visual quality. Our CLUT-Net achieves SOTA on consideration of both the enhancement performance and the model complexity.

5 CONCLUSION

In this paper, we focus on the automatic image enhancement field and proposed a novel lightweight solution based on 3-Dimensional LookUp Table (3DLUT), which is a widely used powerful enhancement operator but with a large parameter amount. To boost the space efficiency of the 3DLUT-based methods, we first conducted thorough analyses of the inherent compressibility of 3DLUT. Then we proposed an effective compressed representation of 3DLUT and built an efficient image enhancement network based on it. Extensive experiments on three benchmark datasets demonstrated that our scheme outperforms the existing state-of-the-art image enhancement methods with a significantly reduced parameter amount.

ACKNOWLEDGMENTS

This work was supported in part by the National Natural Science Foundation of China under Grants 61972285 and 61973235, in part by the Natural Science Foundation of Shanghai under Grant 19ZR1461300, in part by the Shanghai Science and Technology Innovation Plan under Grant 20510760400, in part by the Shuguang Program of Shanghai Education Development Foundation and Shanghai Municipal Education Commission under Grant 21SG23, in part by the Shanghai Municipal Science and Technology Major Project under Grant 2021SHZDZX0100, and in part by the Fundamental Research Funds for the Central Universities.

REFERENCES

[1] Mahmoud Afifi and Michael S. Brown. 2020. Deep white-balance editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 1397–1406. https://doi.org/10.1109/CVPR42600.2020.00147

[2] Mahmoud Afifi, Brian Price, Scott Cohen, and Michael S. Brown. 2019. When color constancy goes wrong: Correcting improperly white-balanced images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 1535–1544. https://doi.org/10.1109/CVPR.2019.00163

[3] Vladimir Bychkovsky, Sylvain Paris, Eric Chan, and Frédo Durand. 2011. Learning photographic global tonal adjustment with a database of input/output image pairs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 97–104. https://doi.org/10.1109/CVPR.2011.5995332

[4] Yu-Sheng Chen, Yu-Ching Wang, Man-Hsin Kao, and Yung-Yu Chuang. 2018. Deep photo enhancer: Unpaired learning for image enhancement from photographs with gans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 6306–6314. https://doi.org/10.1109/CVPR.2018.00660

[5] Wenyan Cong, Xinhao Tao, Li Niu, Jing Liang, Xuesong Gao, Qihao Sun, and Liqing Zhang. 2021. High-resolution image harmonization via collaborative dual transformations. *CoRR* abs/2109.06671 (2021). https://arxiv.org/abs/2109.06671

[6] Yuchen Fan, Jiahui Yu, Yiqun Mei, Yulun Zhang, Yun Fu, Ding Liu, and Thomas S Huang. 2020. Neural Sparse Representation for Image Restoration. In *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (Eds.), Vol. 33. Curran Associates, Inc., 15394–15404. https://proceedings.neurips.cc/paper/2020/file/b090409688550f3cc93f4ed88ec6cafb-Paper.pdf

[7] Michaël Gharbi, Jiawen Chen, Jonathan T Barron, Samuel W Hasinoff, and Frédo Durand. 2017. Deep bilateral learning for real-time image enhancement. *ACM Transactions on Graphics* 36, 4 (2017), 1–12. https://doi.org/10.1145/3072959.3073592

[8] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Proceedings of Advances in Neural Information Processing Systems*. 2672–2680.

[9] Samuel W Hasinoff, Dillon Sharlet, Ryan Geiss, Andrew Adams, Jonathan T Barron, Florian Kainz, Jiawen Chen, and Marc Levoy. 2016. Burst photography for high dynamic range and low-light imaging on mobile cameras. *ACM Transactions on Graphics* 35, 6 (2016), 1–12. https://doi.org/10.1145/2980179.2980254

[10] Jingwen He, Yihao Liu, Yu Qiao, and Chao Dong. 2020. Conditional sequential modulation for efficient global image retouching. In *Proceedings of the European Conference on Computer Vision*. 679–695. https://doi.org/10.1007/978-3-030-58601-0_40

[11] Andrew Howard, Mark Sandler, Bo Chen, Weijun Wang, Liang-Chieh Chen, Mingxing Tan, Grace Chu, Vijay Vasudevan, Yukun Zhu, Ruoming Pang, Hartwig Adam, and Quoc Le. 2019. Searching for MobileNetV3. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 1314–1324. https://doi.org/10.1109/ICCV.2019.00140

[12] Han-Ul Kim, Young Jun Koh, and Chang-Su Kim. 2020. Global and local enhancement networks for paired and unpaired image enhancement. In *Proceedings of the European Conference on Computer Vision*. 339–354. https://doi.org/10.1007/978-3-030-58595-2_21

[13] Han-Ul Kim, Young Jun Koh, and Chang-Su Kim. 2020. PieNet: Personalized image enhancement network. In *Proceedings of the European Conference on Computer Vision*. 374–390. https://doi.org/10.1007/978-3-030-58577-8_23

[14] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proceedings of the International Conference on Learning Representations*. 1–15. https://doi.org/10.48550/arXiv.1412.6980

[15] Chongyi Li, Chunle Guo, Qiming Ai, Shangchen Zhou, and Chen Change Loy. 2020. Flexible piecewise curves estimation for photo enhancement. *CoRR* abs/2010.13412 (2020). arXiv:2010.13412 https://arxiv.org/abs/2010.13412

[16] Yawei Li, Shuhang Gu, Christoph Mayer, Luc Van Gool, and Radu Timofte. 2020. Group Sparsity: The Hinge Between Filter Pruning and Decomposition for Network Compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 8015–8024. https://doi.org/10.1109/CVPR42600.2020.00804

[17] Yawei Li, Shuhang Gu, Luc Van Gool, and Radu Timofte. 2019. Learning Filter Basis for Convolutional Neural Network Compression. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 5622–5631. https://doi.org/10.1109/ICCV.2019.00572

[18] Jie Liang, Hui Zeng, Miaomiao Cui, Xuansong Xie, and Lei Zhang. 2021. PPR10K: A large-scale portrait photo retouching dataset with human-region mask and group-level consistency. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 653–661. https://doi.org/10.1109/CVPR46437.2021.00071

[19] Enyu Liu, Songnan Li, and Shan Liu. 2020. Color enhancement using global parameters and local features learning. In *Proceedings of the Asian Conference on Computer Vision*. 202–216. https://doi.org/10.1007/978-3-030-69532-3_13

[20] Sean Moran, Pierre Marza, Steven McDonagh, Sarah Parisot, and Gregory Slabaugh. 2020. DeepLPF: Deep local parametric filters for image enhancement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 12823–12832. https://doi.org/10.1109/CVPR42600.2020.01284

[21] Sean Moran, Steven McDonagh, and Gregory Slabaugh. 2021. CURL: Neural curve layers for global image enhancement. In *Proceedings of the IEEE International Conference on Pattern Recognition*. 9796–9803. https://doi.org/10.1109/ICPR48806.2021.9412677

[22] Anton Obukhov, Maxim Rakhuba, Stamatios Georgoulis, Menelaos Kanakis, Dengxin Dai, and Luc Van Gool. 2020. T-Basis: a Compact Representation for Neural Networks. In *Proceedings of the 37th International Conference on Machine Learning*, Vol. 119. 7392–7404.

[23] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. Automatic differentiation in pytorch.

[24] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. 2015. U-Net: Convolutional networks for biomedical image segmentation. In *Proceedings of International Conference on Medical image computing and computer-assisted intervention*. 234–241. https://doi.org/10.1007/978-3-319-24574-4_28

[25] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. 2016. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022* (2016).

[26] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of Advances in Neural Information Processing Systems*. 6000–6010. https://doi.org/10.5555/3295222.3295349

[27] Ruixing Wang, Qing Zhang, Chi-Wing Fu, Xiaoyong Shen, Wei-Shi Zheng, and Jiaya Jia. 2019. Underexposed photo enhancement using deep illumination estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 6849–6857. https://doi.org/10.1109/CVPR.2019.00701

[28] Tao Wang, Yong Li, Jingyang Peng, Yipeng Ma, Xian Wang, Fenglong Song, and Youliang Yan. 2021. Real-time image enhancer via learnable spatial-aware 3d lookup tables. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2471–2480. https://doi.org/10.1109/ICCV48922.2021.00247

[29] Zhou Wang, Alan C. Bovik, Hamid R. Sheikh, and Eero P. Simoncelli. 2004. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing* 13, 4 (2004), 600–612. https://doi.org/10.1109/TIP.2003.819861

[30] Bing Xu, Naiyan Wang, Tianqi Chen, and Mu Li. 2015. Empirical evaluation of rectified activations in convolutional network. *arXiv preprint arXiv:1505.00853* (2015).

[31] Hui Zeng, Jianrui Cai, Lida Li, Zisheng Cao, and Lei Zhang. 2022. Learning image-adaptive 3d lookup tables for high performance photo enhancement in real-time. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44, 4 (2022), 2058–2073. https://doi.org/10.1109/TPAMI.2020.3026740