DOI: 10.1111/cgf.70013
EUROGRAPHICS 2025 / A. Bousseau and A. Dai
(Guest Editors)

COMPUTER GRAPHICS *forum*
Volume 44 (2025), Number 2

SAHLUT: Efficient Image Enhancement using Spatial-Aware High-Light Compensation Look-up Tables

Xin Chen[†1,2], Linge Li[‡1], Linhong Mu[1], Yan Chen[2], Jingwei Guan[3§]

[1]Huawei Technologies Co., Ltd, China
[2]Tsinghua University, China
[3]Shenzhen Technology University, China

Abstract

Recently, the look-up table (LUT)-based method has achieved remarkable success in image enhancement tasks with its high efficiency and lightweight nature. However, when considering edge scenarios with limited computational resources, most existing methods fail to meet practical requirements due to their costly floating-point operations on convolution layers, which limit their general use. Moreover, most LUT-based methods may not perform well in handling high-light regions. To address these issues, we propose SAHLUT, an efficient and practical image enhancement method by using spatial-aware high-light compensation look-up tables (LUTs), which comprise two parts. Firstly, we propose a spatial-aware weight predictor to reduce the computational burden. A lightweight network is trained to predict spatial-aware weight values, and then we transfer the values to the LUTs. Additionally, to correct overexposure in high-light regions, we propose a high-light compensation 3D LUT. Our proposed method allows us to directly retrieve the values from the LUTs to achieve efficient image enhancement at test time. Extensive experimental results demonstrate that SAHLUT exhibits competitive performance compared to other LUT-based methods both quantitatively and qualitatively in a more efficient manner. For instance, SAHLUT significantly reduces computational resources (at least 18 times in GFLOPs compared to other LUT-based methods), while excelling in high-light region handling.

CCS Concepts

• ***Computing methodologies*** → *Collision detection;*

1. Introduction

The goal of image enhancement is to adjust the visual quality of an image to make it more closely resemble the target image. The traditional image enhancement methods involve adjusting images to meet human preferences using cascade modules, which are complex and lack flexibility. This has led to a high demand for automatic and adaptable enhancement methods. With the rapid development of artificial intelligence technology, researchers have proposed various deep learning-based automatic image enhancement methods, achieving state-of-the-art (SOTA) results. Although end-to-end image enhancement using fully convolutional networks (FCNs) and generative adversarial networks (GANs) has been realized [YZW*16, CWKC18, PYL*21, WZF*19, WXL22, HHX*18, GLG*20, HLQD20], high computational costs and high memory consumption limit their application in real-time high-resolution image enhancement.

Recent studies have shown that combining LUTs with deep learning can effectively address these issues. [ZCL*20, WLP*21, YJJ*22, YJX*22, LYFQ23, ZLY*22]. These methods propose learnable image-adaptive LUTs for image enhancement. They combine efficient LUTs with convolutional neural networks (CNNs) to achieve adaptive real-time enhancement of high-resolution images. Specifically, they utilize CNNs to process low-resolution input images, extracting image attribute information for subsequent LUT-based enhancement of high-resolution images. These hybrid methods leverage LUTs for efficiency and low memory consumption but still require substantial floating-point convolution computations during testing. Although they partially reduce computational costs, there is still a reliance on high-performance computing units such as GPUs. Applications on general devices with limited computational resources, such as smartphones and TVs, remain challenging due to the computational burden. Moreover, most LUT-based methods face challenges when dealing with images containing high-light regions [ZCL*20, YJJ*22, YJX*22].

Although they use low-resolution input images to get color transformation parameters and efficiently adjust the entire image globally, they often tend to overexpose high-light regions due to ignoring the relationships between high-light pixels.

† Co-first author
‡ Co-first author
§ Corresponding author

© 2025 Eurographics - The European Association for Computer Graphics and John Wiley & Sons Ltd.

[Correction added on 17 April 2025, after first online publication: Article title has been updated from "SHLUT: Efficient Image Enhancement using Spatial-Aware High-Light Compensation Look-up Tables" to "SAHLUT: Efficient Image Enhancement using Spatial-Aware High-Light Compensation Look-up Tables".]

delivered by EUROGRAPHICS DIGITAL LIBRARY www.eg.org diglib.eg.org

To solve these issues, we propose an efficient image enhancement method based on LUTs, which incorporates spatial and color information of input pixels and introduces high-light compensation to the traditional 3D LUT to improve its performance. The LUTs are generated after training and remain fixed during inference. Particularly, by jointly modeling the nonlinear relationship between spatial information and color information, we proposed a spatial-aware weight predictor with a small receptive field (RF) to increase attention to spatial information. Then, we introduce a high-light compensation 3D LUT to correct overexposure. Our method addresses the heavy computational and memory burden during testing by directly retrieving values from these pre-computed LUTs, providing an efficient application solution on limited computational devices. In summary, the contributions of this paper are summarized as follows:

- We propose an efficient and novel method for image enhancement by using spatial-aware high-light compensation LUTs. To our knowledge, this is the first demonstration of the excellence achieved by using LUTs for image enhancement.
- We design a spatial-aware weight predictor that learns spatial and color information in pixels with low computational overhead.
- We introduce a high-light compensation 3D LUT for effectively improving the enhancement of high-light regions.
- Extensive qualitative and quantitative experiments on three benchmark datasets show that SAHLUT outperforms LUT-based methods more efficiently.

2. RELATED WORK

2.1. Learnable Lookup Tables

A lookup table (LUT) stores values and uses indices to access them Recently, LUT-based deep learning methods have gained traction in low-level vision tasks. In image super-resolution, Jo *et al.* introduced the innovative full lookup tables method [JK21] by establishing the correspondence of low-resolution (LR) input patterns and high-resolution (HR) output patterns. Subsequently, to address the limitation in the RF size, Ma *et al.* [MZZL22] proposed the parallel lookup tables (SPLUT) framework, while Li *et al.* [LCCX22] employed multiple LUTs with complementary indexing for hierarchical access. Taking a convolutional perspective, Liu *et al.* [LDL*23] introduced a novel reconstructed convolution (RC) module to overcome the challenges of RF size. By reconstructing the convolution operation, the RC module effectively expands the RF without significantly increasing computational complexity.

In video enhancement, She *et al.* [SX22] introduced a learnable 2D LUT to mitigate temporal flickering with a virtual camera motion engine. Li *et al.* [LWW*23] tackled inter-frame brightness consistency using an intensity-aware LUT module. In terms of the image enhancement field, these methods combine CNNs and LUTs, proposing learnable, image-adaptive LUTs for image enhancement [ZCL*20, WLP*21, YJJ*22, YJX*22, LYFQ23].

2.2. Deep Learning Based Image Enhancement

In traditional image enhancement methods, color mapping relationships are often modeled using linear or nonlinear equations [HLSKJK14, WYX11]. Recent advances leverage deep learning for image enhancement [YZW*16, GLG*20, HLQD20, HHX*18, CWKC18, PYL*21, WZF*19, WXL22, ZZG19, LMZ*21]. Some studies focus on pixel-level mappings [YZW*16, CWKC18, PYL*21], while others train neural networks to fit color mapping models or curves [GLG*20, HLQD20, HHX*18]. Retinex-based methods [WZF*19, ZZG19, LMZ*21, WXL22] decompose images into illumination and reflectance layers to address underexposure and overexposure. For example, UPE [WZF*19] estimates illumination maps for underexposed images, and Wang *et al.* [WXL22] propose a dual-illumination mechanism for uneven lighting. Despite achieving state-of-the-art results, these methods face challenges in practical deployment due to high computational costs and limited input resolution.

2.3. Hybrid Based Image Enhancement

To reduce computational costs while maintaining enhancement quality, hybrid methods combining CNNs with color transformation functions have been proposed. These methods typically use CNNs to extract multi-level features and predict parameters for guiding color transformations.

A common transformation function is the 3D LUT. Zeng *et al.* [ZCL*20] pioneered CNN-guided 3D LUTs for image-adaptive color transformation. Wang *et al.* [WLP*21] extended this by incorporating local spatial information, while Yang *et al.* [YJJ*22] introduced adaptive sampling intervals to model local nonlinearities. Simultaneously, they decoupled a single color transformation into component-independent and component-dependent transformations [YJX*22], improving the unit utilization of 3D LUTs. Cong *et al.* [CTN*22] integrated a LUT-based sub-module for high-resolution image harmonization, while Liu *et al.* [LYFQ23] introduced 4D LUTs, extending 3D LUTs with a context map channel. Zhang *et al.* [ZLY*22] emphasized incorporating 3D LUT information into adaptive weight estimation, while Shi *et al. [SCH*23] developed a cross-attention framework for global color and semantic feature learning. Kim et al. [KY23] proposed a LUT-based compression method using straight line referencing, validated on FPGA platforms.* Beyond LUTs, Gharbi *et al.* [GCB*17] proposed learning transformation coefficients in bilateral space, and others explored affine transformation matrices [WZF*19, CGW19, LLL20], demonstrating efficiency across diverse scenes.

While extensive research has focused on efficient, high-quality image enhancement with some achieving real-time performance on GPUs, these methods remain computationally intensive for resource-constrained devices due to their heavy reliance on convolutional operations. Moreover, they often inadequately address high-light region enhancement.

In contrast, our proposed method efficiently addresses the computational burden issue by retrieving values from LUTs in memory. Furthermore, our method introduces a high-light compensation 3D LUT to correct overexposure. It might be a suitable choice for resource-constrained devices, while other hybrid-based methods may have limited applicability due to their reliance on computationally intensive convolutional layers.

© 2025 Eurographics - The European Association
for Computer Graphics and John Wiley & Sons Ltd.

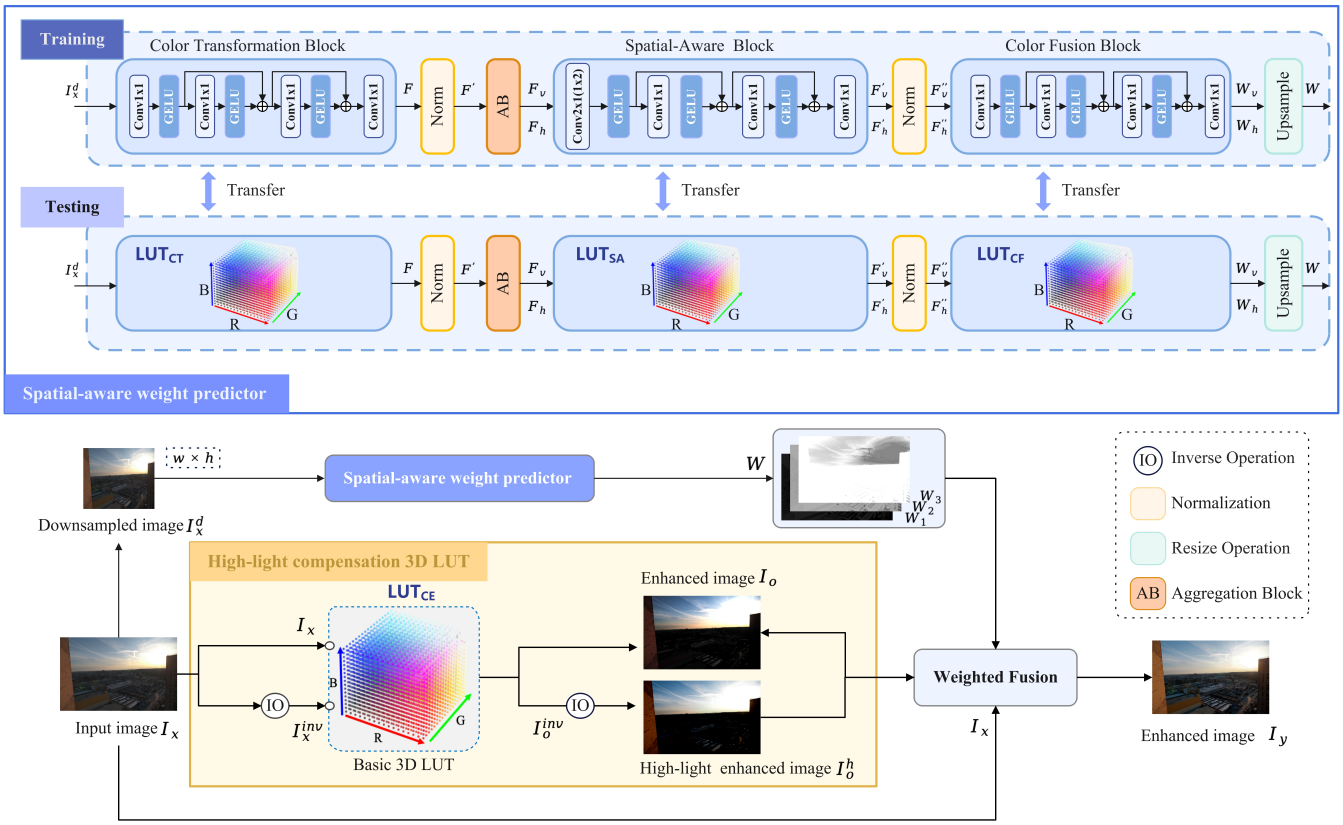


Figure 1: *Framework of the proposed method. During training, we utilize a lightweight network (with a small receptive field) to predict spatial-aware weight values, and jointly optimize the high-light compensation 3D LUT, initializing it as the basic 3D LUT (named* $\mathrm{LUT_{CE}}$*). After training, we transfer the output values of the network to the corresponding* $\mathrm{LUT_{CT}}$*,* $\mathrm{LUT_{SA}}$*, and* $\mathrm{LUT_{CF}}$*, then retrieve the values from the four LUTs at test time. The inverse operation means* $I_x^{inv} = 1 - I_x$*.*

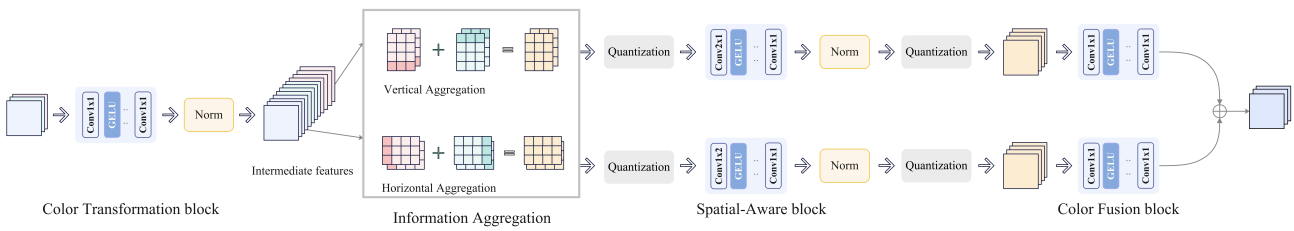


Figure 2: *Illustration of the spatial-aware weight predictor. In the aggregation block, dark squares represent dark squares represent reflection padding results. "Quantization" refers to converting 32-bit floating-point features to 8-bit integers. Note that quantization is only required during training, not testing. "Norm" represents normalization operation* $X' = \left(\frac{X-\mu}{\sigma}\right)$*, where* μ *is the mean, and* σ *is the standard deviation.*

3. METHOD

In this section, we present in detail our network framework and give a thorough analysis of spatial-aware weight predictor and high-light compensation 3D LUT in Sect. 3.2 and 3.3, respectively.

3.1. Overall Framework

Our network architecture consists of a spatial-aware weight predictor and a high-light compensation 3D LUT as shown in Fig. 1. Specifically, a spatial-aware weight predictor is employed to predict pixel-level weight mappings $W = \{W_1, W_2, W_3\} \in \mathbb{R}^{3\times H\times W}$. In the high-light compensation 3D LUT, we employ $\mathrm{LUT_{CE}}$ for color mapping on the input image I_x and the inverted image I_x^{inv} to obtain the enhanced image I_o and high-light enhanced image I_o^h. Then our final enhanced image I_y can be represented as follows,

$$I_y = I_o \cdot W_1 + I_o^h \cdot W_2 + I_x \cdot W_3, \tag{1}$$

© 2025 Eurographics - The European Association for Computer Graphics and John Wiley & Sons Ltd.

where $W_1, W_2, W_3 \in \mathbb{R}^{1\times H\times W}$ represent the weight mappings, $I_x \in [0,1]^{3\times H\times W}$ represents the input image. $I_o \in [0,1]^{3\times H\times W}$ represents the enhanced output, and $I_o^h \in [0,1]^{3\times H\times W}$ represents the high-light enhanced output. Our LUT generation operates in RGB color space using the Rec.709 color gamut, where all three dimensions (R, G, B) are treated equally for uniform sampling.

3.2. Spatial-aware weight predictor

Existing LUT-based methods primarily leverage global information from downsampled images for color mapping, often compromising local details. While SA-3DLUT [WLP*21] incorporates spatial information through UNet-like architectures, its CNN-based pixel-wise mapping incurs substantial computational overhead during inference, highlighting the challenge of balancing local detail preservation and computational efficiency.

Our proposed method, a spatial-aware weight predictor, employs a small receptive field to adaptively learn local spatial and color features, while utilizing LUT-based value retrieval during inference to reduce computational costs. Fig. 2 illustrates the spatial-aware weight predictor. We will provide a detailed explanation in the following sections. Note that to maintain consistency in accuracy between training and testing, we add quantization operations during training.

3.2.1. Color Transformation Block.

We employ 1×1 convolution to transform individual pixels from RGB space to a higher-dimensional pixel-aware color space, leveraging neural networks' non-linear mapping capabilities. Suppose that an down-sampled image $I_x^d \in [0,1]^{3\times w\times h}$ is given, the transformed features $F \in \mathbb{R}^{c\times w\times h}$ are computed as:

$$F = h_{CT}(I_x^d), \tag{2}$$

where h_{CT} represents the color transformation mapping and c denotes the output dimension.

3.2.2. Aggregation Block.

To further emphasize local information, our method explores the spatial dependencies of input images at two levels : the aggregation block and the spatial-aware block. Firstly, we utilize an aggregation block for spatial information interaction, similar to the method employed in SPLUT [MZZL22]. As shown in Fig. 2, after the color transformation block, we normalize the features F to adjust its distribution and get the intermediate features F', then input F' into the aggregation block.

Taking the vertical information as an example, half of the intermediate features F' are reflected and padded on the bottom, and the other half on the top, resulting in a final feature map F_v with $C/2$ channels after pixel-wise addition and generate $F_v \in \mathbb{R}^{\frac{c}{2}\times w\times h}$. After adding two padded feature maps together, the vertical RF size is enlarged.

Similar operations are performed in the horizontal direction and generate $F_h \in \mathbb{R}^{\frac{c}{2}\times w\times h}$. Through the aggregation block, our method can increase the RF size in the vertical and horizontal directions respectively.

3.2.3. Spatial-Aware Block.

To further enhance spatial dependency, we use a network with a small RF size (2x1 and 1x2) to more efficiently learn spatial information. As depicted in Fig. 2, we model the spatial correlation separately in both the vertical and horizontal directions, and the corresponding result is generated as,

$$\begin{aligned} F_v^{'} &= h_{SA1}(Q(F_v)) \\ F_h^{'} &= h_{SA2}(Q(F_h)), \end{aligned} \tag{3}$$

where $F_v^{'}, F_h^{'} \in \mathbb{R}^{3\times w\times h}$ respectively denoted the fused information from the horizontal and vertical directions, Q represents quantizing the feature, h_{SA1} and h_{SA2} represent the mappings with kernels 2x1 and 1x2, respectively.

3.2.4. Color Fusion Block.

In addition to spatial local information, we also consider the correlation between color channels. Thus, it is crucial to separately handle the horizontal and vertical features. After normalization, the color fusion block takes the $F_v^{''}$ and $F_h^{''}$ as input, and then predict output values $W_v, W_h \in \mathbb{R}^{3\times w\times h}$ by,

$$\begin{aligned} W_v &= h_{CF1}(Q(F_v^{''})) \\ W_h &= h_{CF2}(Q(F_h^{''})) \\ W &= \frac{1}{2}(W_v + W_h). \end{aligned} \tag{4}$$

Finally, the outputs $W = \{W_1, W_2, W_3\} \in \mathbb{R}^{3\times w\times h}$ is upsampled to original resolution $W \in \mathbb{R}^{3\times W\times H}$ using bilinear interpolation.

3.3. High-light compensation 3D LUT

3.3.1. Color enhancement using a 3D LUT

In our method, we view a 3D LUT as a discrete sampling of a complete 3D color enhancement function. The sampled results are stored in a 3D lattice of output color values $T = (T_{r(i,j,k)}, T_{b(i,j,k)}, T_{g(i,j,k)})_{i,j,k\in I_0^{N_s-1}}$. T can be queried by sets of input color coordinates P, which can be denoted as $P = (P_{r(i,j,k)}, P_{b(i,j,k)}, P_{g(i,j,k)})_{i,j,k\in I_0^{N_s-1}}$. where N_s is the number of sampling coordinates along each of three dimensions and $I_0^{N_s-1}$ denotes the set of $\{0, 1, ..., N_s-1\}$. Such a lattice defines a total of N_s^3 sampling points on the complete 3D color transform function.

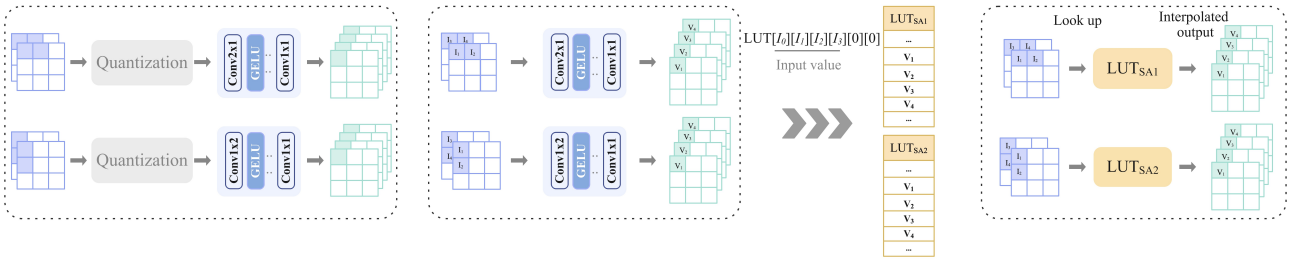
Suppose that N_s (we set $N_s = 33$, following the common practice in recent works [ZCL*20, YJJ*22]) sampling coordinates along each dimension, balancing efficiency and precision. Given an input image $X \in [0,1]^{3\times H\times W}$, the output color values of a 3D LUT can be formulated as,

$$T = f(X), \tag{5}$$

where f is a function mapping an input image X and predicts all output color values in T, the specific implementation details of color enhancement mapping using 3D LUTs can be referred in [ZCL*20].

Obviously, traditional 3D LUT mapping, based solely on pixel values without positional context, often fails to handle transitional

© 2025 Eurographics - The European Association for Computer Graphics and John Wiley & Sons Ltd.



(a) Traning of $\mathrm{LUT}_{\mathrm{SA}}$ (b) Transferring of $\mathrm{LUT}_{\mathrm{SA}}$ (c) Testing of $\mathrm{LUT}_{\mathrm{SA}}$

Figure 3: *The process of the our method, using* $\mathrm{LUT}_{\mathrm{SA}}$ *as an example. We train a small network with the RF size (2) and kernel shape (1 x 2 and 2 x 1) to transfer the trained outputs to the* $\mathrm{LUT}_{\mathrm{SA}}$. *During testing, we exclusively use* $\mathrm{LUT}_{\mathrm{SA}}$ *for lookup and interpolation processing.*

high-light regions, leading to banding artifacts and overexposure. We address this limitation through a single high-light compensated 3D LUT, in contrast to conventional approaches requiring multiple LUTs (at least three) [ZCL*20,ZLY*22]. The details are elaborated in subsequent sections.

3.3.2. High-light compensation.

In the Retinex-based image enhancement theory [WZHL13, GLL16,ZYX*18], it is assumed that an image I can be represented as the pixel-wise product of the desired enhanced image I_y and a single-channel illumination mapping L as $I = I_y \times L$. where $\times$ denotes pixel-wise multiplication. However, these methods struggle with overexposed images. Zhang *et al.* [ZNZ19] correct the overexposed regions by reversing the input image, and combine the forward illumination map with the inverted backward illumination map to obtain the enhanced image.

Inspired by [ZNZ19], we propose a high-light compensation 3D LUT. It involves two steps: Firstly, we directly enhance the input image I_x using $\mathrm{LUT}_{\mathrm{CE}}$ to get the enhanced image I_o by $I_o = f(I_x)$, f is a 3D LUT color transform function. This step is similar to forward illumination estimation in [ZNZ19], but our method can directly enhance the image without estimating an illumination map. Next, we further considered enhancing the high-light regions by inverting the input image as $I_x^{inv} = 1 - I_x$ and applying the same 3D $\mathrm{LUT}_{\mathrm{CE}}$ to get image I_o^{inv}, then inverse the image to get the high-light enhanced image I_o^h by $I_o^h = 1 - I_o^{inv}$. The second step is aiming to effectively correct overexposure in high-light regions. As in the inverted image I_x^{inv}, the originally high-light and overexposed regions now appear as low-light and underexposed. By correcting the underexposed regions in the inverted image I_x^{inv}, we can restore the overexposed regions in the original image I_x.

Finally, the enhanced I_o, the high-light enhanced I_o^h and the input image I_x are fused according to the weight maps $\{W_1, W_2, W_3\}$ to obtain the enhanced image I_y.

3.4. Deployment process

In this section, we will provide the detailed deployment process. Taking the spatial-aware block as an example, Fig. 3 illustrates the detailed deployment process.

3.4.1. Training lightweight network

We train a lightweight network with a small RF size to predict the values. The training process shown in Fig. 1 demonstrates the convolutional networks used in three convolution modules. Only the first convolutional layer in the spatial-aware block has an RF size of 2, while all other convolutional layers in our network have an RF size of 1. Benefiting from the small RF size, we can enumerate the predicted values in the following transferring process.

3.4.2. Transferring to LUT

After training the network, we transfer the predicted values from each convolution module to the corresponding $\mathrm{LUT}_{\mathrm{CT}}$, $\mathrm{LUT}_{\mathrm{SA1}}$, $\mathrm{LUT}_{\mathrm{SA2}}$, $\mathrm{LUT}_{\mathrm{CF1}}$, and $\mathrm{LUT}_{\mathrm{CF2}}$, where subscripts 1 and 2 represent LUTs generated from vertical and horizontal aggregation respectively, and we collectively refer to them as W-LUT. Note that $\mathrm{LUT}_{\mathrm{CE}}$ is initialized as a 3D LUT during training, then there is no need to transfer it. Along with the pre-stored $\mathrm{LUT}_{\mathrm{CE}}$, they are collectively referred to as SAHLUT.

In practice, we employ a uniformly sampled LUT due to the large size of the complete LUT (the full W-LUT takes 228GB). The uniformly sampled LUT divides the original input space uniformly with a sampling interval of 2^4.

3.4.3. Testing Using SAHLUT

After transferring the W-LUT, image enhancement exclusively relies on LUTs. When utilizing the sampled LUTs, output values are produced through suitable interpolation techniques. In W-LUT we use the high tetrahedral interpolation method in SR-LUT [JK21] to index the nearest sampled points as it is more suitable for interpolating among four points. For the high-light compensation, we adopt the same method in [ZCL*20]. The whole testing process is illustrated in Fig. 1.

4. Experiments

4.1. Experimental Setting

4.1.1. Datasets and metrics.

We evaluate our proposed method using three publicly datasets: MIT-Adobe FiveK [BPCD11], PPR10K [LZC*21] and HDR+

© 2025 Eurographics - The European Association
for Computer Graphics and John Wiley & Sons Ltd.

[HSG*16]. MIT-Adobe FiveK dataset is a commonly used dataset for image enhancement with ground truth in versions $A/B/C/D/E/F$. We select version C (both the 480 version and the original version) and follow the common practice in recent works [ZCL*20, WLP*21, YJJ*22, YJX*22], with 4500 images for training and 500 images for testing. For the HDR+ dataset, we use the settings aligned with [ZCL*20], utilizing 16-bit TIF images as input and conducting experiments on the 480p version. The PPR10K dataset comprises 11,161 RAW portrait photos with ground truth versions $(a/b/c)$. We use all versions for training (8,875 pairs) and testing (2,286 pairs), following [LZC*21, YJJ*22, YJX*22], using the 360p version in experiments.

We conduct experiments in two typical application scenarios: photo retouching and tone mapping. Photo retouching aims to enhance the visual quality of input images in the sRGB format, while tone mapping converts 16-bit sRGB input images into 8-bit sRGB target images, adjusting the dynamic range and preserving details to ensure a natural appearance.

We perform both tasks on the FiveK dataset, photo retouching on the PPR10K dataset, and tone mapping on the HDR+ dataset. We use PSNR, SSIM, L_2-distance in CIE LAB color space (ΔE_{ab}), and GFLOPs to evaluate different methods.

4.1.2. Training Setting.

Our training process is end-to-end, utilizing mean squared error (MSE) for reconstruction loss, along with smoothness loss and monotonicity loss [ZCL*20] to constrain the output of the color enhancement 3D LUTs. We use Adam [KB15] as the optimizer with a learning rate of 10^{-4}. We perform our experiments with Pytorch framework [PGM*19].

4.2. Comparisons with State-of-the-Arts

4.2.1. Quantitative Comparisons.

We select typical settings for our method with $c = 12$, $w = 256$, and $h = 256$, and compare its performance against several state-of-the-art (SOTA) image enhancement methods, including UPE [WZF*19], DPE [CWKC18], HDRNet [GCB*17], DeepLPF [MMM*20], 3D-LUT [ZCL*20], AdaInt [YJJ*22], SepLUT [YJX*22], and DualBLN [ZLY*22]. The results for these methods are obtained using their publicly available code with default settings. Table 1, Table 2 and Table 3 report the results in terms of PSNR, SSIM, ΔE_{ab} and GFLOPs. We compare the performance in both photo retouching and tone mapping using the MIT-Adobe FiveK, HDR+, and PPR10K datasets. Table 1 and Table 3 show the results in photo retouching scenario.

Compared to other methods, our method achieves optimal performance across all metrics on the original FiveK dataset with extremely low floating-point computation, as high-lighted in the right of Table 1. When compared with 3D-LUT [ZCL*20] on 480p FiveK and PPR10K ($E(b)$, $E(c)$) dataset, our method achieves a slight improvement while reducing the computational burden significantly (at least 18 times in GFLOPs). Note that AdaInt [YJJ*22] and SepLUT [YJX*22] perform slightly better than our method. However, they also consume more computational resources. Similar conclusions can also be drawn in the tone mapping scenario in Table 2, our method outperforms the second best by 0.33dB on PSNR and improves over all other compared methods across all metrics on the HDR+ dataset.

Overall, our method performs comparably to all competing methods on three datasets, with a significant advantage in reducing the computational burden.

4.2.2. Qualitative Comparisons.

In Fig. 4, we also provide some visual comparisons where our method yields more visually pleasing results. Our method outperforms other LUT-based methods in handling high-light regions, avoiding severe overexposure and banding artifacts, and achieving smoother transitions. It successfully enhances the details in bright areas, making the image more realistic.

As shown in the first row of Fig. 4, our method naturally enhances high-light regions with clouds. By preserving and highlighting fine details in these areas, it delivers a more realistic and visually appealing result. In contrast, 3D-LUT [ZCL*20], AdaInt [YJJ*22], and SepLUT [YJX*22] struggle with noticeable overexposure and detail loss. In the second row, the superiority of our method becomes evident in the rendering of the sky. Our method achieves a smoother and more natural color transition, avoiding abrupt shifts in color. The third and fourth rows further highlight our method's advantage in maintaining smooth boundaries in high-light regions. While the other LUT-based methods suffer from banding artifacts—visible as harsh, unnatural transitions between different levels of brightness.

4.3. Ablation Study and Analysis

In this section, we conduct several ablation studies on the photo retouching task with the FiveK dataset (480p) to verify the key components of our proposed method.

4.3.1. Feature capacity of Color Transformation Block.

The color transformation block in the weight predictor plays a vital role in generating pixel-level features.

The output feature capacity may directly affect the overall performance. To verify this, we conducted experiments by varying the output dimensions through the hyperparameter c. Results in Table 4 show that increasing feature capacity doesn't necessarily enhance performance, it might introduce redundancy and training complexity. To balance performance and memory usage, we adopt $c = 12$.

4.3.2. Normalization block.

While transferring the predicted values from CNN modules to LUTs, values may exceed the LUTs' effective range. To improve LUT utilization, a normalization block adjusts the features to ensure they are appropriately distributed to subsequent LUTs. We conduct ablation experiments to verify this.

In Table 5, the model with all normalization blocks outperforms the model without normalization by 3.36 dB in PSNR, underscoring the importance of normalization for model performance. Additionally, adding normalization after the spatial-aware block and color transformation block yields PSNR differences of 3.23 dB and 2.89 dB, respectively, further confirming its significance.

© 2025 Eurographics - The European Association for Computer Graphics and John Wiley & Sons Ltd.

Table 1: *Quantitative comparisons on the **FiveK** dataset for **photo retouching**. "-" means the result is not available due to insufficient GPU memory. Red represents the best result, and blue represents the second best. GFLOPs represent the floating-point computational complexity of the backbone CNNs in other methods, as well as the lookup and interpolation operations of the LUTs in our method.*

Method	GFLOPs↓	FiveK(480p)			FiveK(ori)		
		PSNR↑	SSIM↑	ΔE_{ab} ↓	PSNR↑	SSIM↑	ΔE_{ab} ↓
UPE [WZF*19]	0.1522	21.88	0.853	10.80	20.09	0.790	11.79
DPE [CWKC18]	-	23.75	0.908	9.34	-	-	-
HDRNet [GCB*17]	0.1701	24.32	0.912	8.49	22.34	0.822	11.18
DeepLPF [MMM*20]	43.3313	24.73	0.916	7.99	-	-	-
3D-LUT [ZCL*20]	0.1477	25.21	0.922	7.61	23.56	0.833	10.65
AdaInt [YJJ*22]	0.1480	25.48	0.926	7.47	23.43	0.901	10.09
SepLUT [YJX*22]	0.1478	25.42	0.921	7.51	23.36	0.899	10.21
4DLUT [CVCBT24]	0.1478	25.06	0.892	9.54	22.50	0.815	11.27
Ours	0.0081	25.24	0.920	7.54	23.63	0.904	9.890

Table 2: *Quantitative comparisons on the **FiveK** and **HDR+** dataset for **tone mapping**.*

Method	FiveK(480p)			Method	HDR+		
	PSNR↑	SSIM↑	ΔE_{ab} ↓		PSNR↑	SSIM↑	ΔE_{ab} ↓
UPE [WZF*19]	21.56	0.837	12.29	UPE [WZF*19]	21.21	0.879	13.05
DPE [CWKC18]	22.93	0.894	11.09	DPE [CWKC18]	22.56	0.872	10.45
HDRNet [GCB*17]	24.52	0.915	8.14	HDRNet [GCB*17]	23.04	0.879	8.97
3D-LUT [ZCL*20]	25.07	0.920	7.55	3D-LUT [ZCL*20]	23.54	0.885	7.93
AdaInt [YJJ*22]	25.28	0.925	7.48	AdaInt [YJJ*22]	23.80	0.860	9.35
SepLUT [YJX*22]	25.43	0.922	7.43	SepLUT [YJX*22]	23.44	0.850	9.63
DualBLN [ZLY*22]	24.23	0.914	8.80	DualBLN [ZLY*22]	23.77	0.885	7.93
4DLUT [CVCBT24]	25.08	0.905	9.37	4DLUT [CVCBT24]	23.28	0.848	10.41
Ours	25.03	0.919	7.72	Ours	24.13	0.914	7.39

Table 3: *Quantitative comparisons on the **PPR10K** dataset for **photo retouching**, where "E" denotes "Expert", and a, b, c indicate the ground truth version retouched by three experts.*

Method	E(a)		E(b)		E(c)	
	PSNR↑	ΔE_{ab} ↓	PSNR↑	ΔE_{ab} ↓	PSNR↑	ΔE_{ab} ↓
3D-LUT [ZCL*20]	24.41	8.02	24.16	8.37	24.45	8.25
AdaInt [YJJ*22]	24.53	7.80	24.31	8.03	24.48	8.17
SepLUT [YJX*22]	24.55	7.88	24.10	8.23	24.45	8.23
4DLUT [CVCBT24]	24.16	10.46	23.99	9.92	24.16	10.04
Ours	24.26	8.20	24.17	8.30	24.47	8.32

Table 4: *Ablation study on the feature capacity of color transformation.*

c	4	8	12	16
PSNR↑	25.04	25.07	25.24	25.22
SSIM↑	0.916	0.917	0.920	0.919
ΔE_{ab}↓	7.72	7.64	7.55	7.60

Table 5: *Ablation study on the normalization block. CT and SA respectively stands for color transformation and spatial-aware block. ✓ indicates the addition of a normalization block afterward.*

CT	SA	PSNR↑	SSIM↑	ΔE_{ab} ↓
		21.88	0.864	10.25
✓		24.77	0.909	7.92
	✓	25.11	0.920	7.66
✓	✓	25.24	0.920	7.55

4.3.3. RF size and Kernel shape.

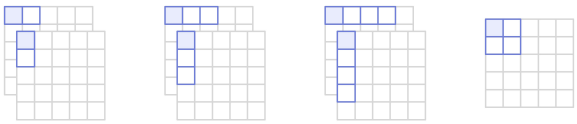
We conduct experiments with various configurations to analyze the impact of receptive field (RF) size and kernel shape. Specifically, we test RF sizes ranging from 2 to 4, with detailed results presented in Table 6 and Fig. 5. Our approach, utilizing an RF size of 2 with both 1x2 and 2x1 kernels, incorporates a total of 4 input pixels (2 channels), which we designate as configuration B.

With an RF size of 3, using 1x3 and 3x1 kernels, it covers only 3 input pixels (1 channel). Similarly, an RF size of 4 (configurations C and D) covers 4 input pixels (1 channel). We used a small number of channels to avoid increasing the complexity of interpolation. As shown in Table 6, configuration B (Ours) shows much better PSNR, SSIM and ΔE_{ab} values than others. We infer this is because it uses

© 2025 Eurographics - The European Association
for Computer Graphics and John Wiley & Sons Ltd.

(a) Input (b) 3D-LUT (c) AdaInt (d) SepLUT (e) Ours (f) GT

Figure 4: *Qualitative comparisons on the **FiveK** dataset for **photo retouching**. The boxes indicate the high-light regions. Our method enhances the high-light regions more naturally compared to other LUT-based methods, bringing them closer to the GT.*



B C D E

Figure 5: *The detailed configurations B/C/D/E in Table 6.*

Table 6: *Ablation study on the RF size and kernel shape. "c" indicates the channels of kernel.*

	RF	kernel	c	PSNR↑	SSIM↑	ΔE_{ab} ↓
A	2	1x2 and 2x1	1	25.16	0.912	7.65
B	2	1x2 and 2x1	2	25.24	0.920	7.55
C	3	1x3 and 3x1	1	25.09	0.903	7.66
D	4	1x4 and 4x1	1	25.02	0.886	7.73
E	4	2x2	1	25.12	0.914	9.39

dual channels, focusing on both horizontal and vertical direction information at the same time.

4.3.4. Sampling interval.

Here, we have conducted experiments comparing different sampling intervals as shown in Table 7. For sampling size 2^4, the size of all LUTs greatly decreases to 4.73MB (Compared to the full LUT table, the 2^4 LUT table saves about 50,000 times the storage space) while minimizing the loss of the original performance (-0.07dB, -0.002 and 0.08 for PSNR, SSIM and ΔE_{ab} respectively). For practical implementation, the trade-off between the accuracy and the size should be considered and we consider a sampling interval of 2^4 is the most suitable compromise. Therefore, we choose the size 2^4 as our default setting.

Table 7: *Comparison of between the full LUT and the sampled LUT. Size indicates the total size of the W-LUT.*

Sampling	Size	PSNR↑	SSIM↑	ΔE_{ab} ↓
2^5	397KB	25.07	0.914	7.66
2^4	4.73MB	25.24	0.920	7.55
2^3	65.64MB	25.28	0.920	7.55
2^2	0.96GB	25.31	0.920	7.53
2^1	14.52GB	25.31	0.921	7.53
(Full LUT)	228GB	25.31	0.922	7.47

4.3.5. Credibility of high-light compensation.

To demonstrate the optimization of SAHLUT, we utilize gradient magnitude similarity deviation (GMSD, where lower values indicate better performance) and gradient magnitude heatmaps (with lighter colors signifying more natural transitions) to compare LUT-based methods in highlight regions.

As shown in Fig. 7, SAHLUT outperforms other methods both qualitatively and quantitatively. Our method achieves the lowest GMSD values (0.0133 and 0.0134), significantly surpassing other LUT-based approaches and indicating optimal enhancement in highlights. The gradient magnitude heatmap analysis reveals that our method has the lightest colors in Example 1, reflecting superior enhancement. In Example 2, our method closely matches the ground truth, while other methods exhibit pronounced colors at highlight boundaries, confirming their lack of smooth transitions.

© 2025 Eurographics - The European Association for Computer Graphics and John Wiley & Sons Ltd.

(a) Input (b) *2* (c) *3* (d) *4* (e) *1* + HL (f) *2* + HL (g) *3* + HL (h) GT

Figure 6: *Enhanced images of different numbers of 3D LUTs and high-light compensation inputs. In 'x', x means numbers of 3D LUTs, HL denotes the input for high-light compensation.*

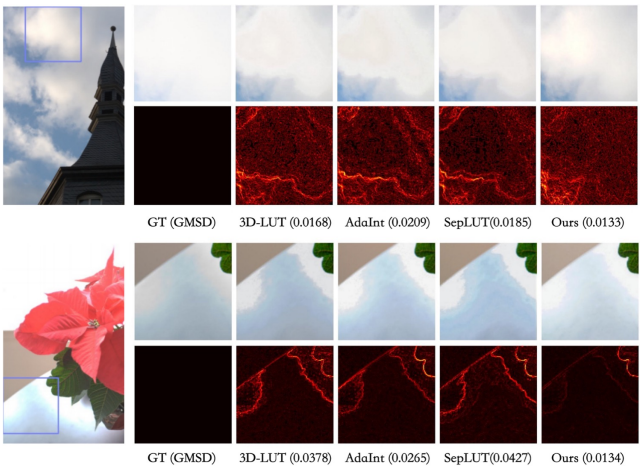
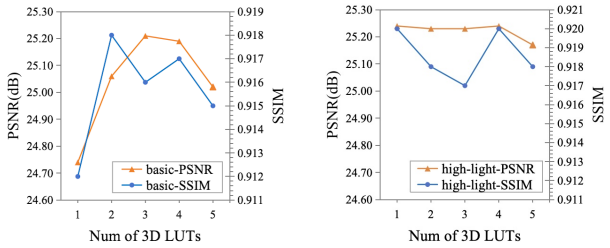


Figure 7: *Gradient magnitude heatmaps and GMSD comparisons in high-light regions on the FiveK dataset.*



(a) input I_x (b) input $I_x + I_x^{inv}$

Figure 8: *Fig(a)-(b) indicate the ablation study on the High-light Compensation property vs Num of 3D LUTs.*

4.3.6. High-light compensation property vs Num of 3D LUTs.

We investigate different configurations of high-light compensation input and the number of 3D LUTs, as shown in Fig. 6 and Fig. 8. Our experiments reveal that increasing the number of 3D LUTs does not necessarily improve performance, and may even increase training cost and degrade image quality, particularly in high-light regions. When considering subjective performance and objective metrics, using a single 3D LUT is the optimal choice for simultaneously inputting I_x and I_x^{inv}. This may be because I_x^{inv} helps adaptively adjust certain high-light color units in a single 3D LUT.

5. Limitation

Our method achieves efficient photo enhancement but has several limitations. First, it currently focuses only on exposure correction, without addressing low-light region enhancement, which we will explore in future work. Additionally, our performance evaluation has limitations. While GFLOPs is a widely used hardware-agnostic complexity metric, it does not fully capture actual processing time. Factors such as hardware architecture (CPU/GPU), memory access patterns, and interpolation algorithm parallelization significantly affect real-world performance. Despite these constraints, we use GFLOPs as our primary metric for standardized cross-method comparisons. Our approach demonstrates competitive results with lower computational complexity, enabling efficient CPU execution through pre-computed LUT lookups.

6. Conclusion

In this study, we propose an efficient image enhancement method using Lookup Tables (LUTs). Our approach delivers competitive visual quality, particularly in high-light regions, comparable to existing LUT-based methods. By relying on precomputed values from the SAHLUT and performing minimal calculations for the final output, we simplify the enhancement process. This efficiency reduces computational demands, making it suitable for real-time applications. We believe our method will be the preferred choice in real-world scenarios due to its practicality and efficiency. Extensive experimental validation further confirms its effectiveness.

References

[BPCD11] BYCHKOVSKY V., PARIS S., CHAN E., DURAND F.: Learning photographic global tonal adjustment with a database of input/output image pairs. In *CVPR 2011* (2011), IEEE, pp. 97–104. 5

[CGW19] CHAI Y., GIRYES R., WOLF L.: Supervised and unsupervised learning of parameterized color enhancement. *2020*

© 2025 Eurographics - The European Association
for Computer Graphics and John Wiley & Sons Ltd.

IEEE Winter Conference on Applications of Computer Vision (WACV) (2019), 981–989. URL: https://api.semanticscholar.org/CorpusID:210701010. 2

[CTN*22] CONG W., TAO X., NIU L., LIANG J., GAO X., SUN Q., ZHANG L.: High-resolution image harmonization via collaborative dual transformations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2022), pp. 18470–18479. 2

[CVCBT24] CONDE M. V., VAZQUEZ-CORRAL J., BROWN M. S., TIMOFTE R.: Nilut: Conditional neural implicit 3d lookup tables for image enhancement. In *Proceedings of the AAAI Conference on Artificial Intelligence* (2024), vol. 38, pp. 1371–1379. 7

[CWKC18] CHEN Y.-S., WANG Y.-C., KAO M.-H., CHUANG Y.-Y.: Deep photo enhancer: Unpaired learning for image enhancement from photographs with gans. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2018), pp. 6306–6314. 1, 2, 6, 7

[GCB*17] GHARBI M., CHEN J., BARRON J. T., HASINOFF S. W., DURAND F.: Deep bilateral learning for real-time image enhancement. *ACM Transactions on Graphics (TOG) 36*, 4 (2017), 1–12. 2, 6, 7

[GLG*20] GUO C., LI C., GUO J., LOY C. C., HOU J., KWONG S., CONG R.: Zero-reference deep curve estimation for low-light image enhancement. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (2020), pp. 1780–1789. 1, 2

[GLL16] GUO X., LI Y., LING H.: Lime: Low-light image enhancement via illumination map estimation. *IEEE Transactions on image processing 26*, 2 (2016), 982–993. 5

[HHX*18] HU Y., HE H., XU C., WANG B., LIN S.: Exposure: A white-box photo post-processing framework. *ACM Transactions on Graphics (TOG) 37*, 2 (2018), 1–17. 1, 2

[HLQD20] HE J., LIU Y., QIAO Y., DONG C.: Conditional sequential modulation for efficient global image retouching. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIII 16* (2020), Springer, pp. 679–695. 1, 2

[HLSKJK14] HWANG Y., LEE J.-Y., SO KWEON I., JOO KIM S.: Color transfer using probabilistic moving least squares. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2014), pp. 3342–3349. 2

[HSG*16] HASINOFF S. W., SHARLET D., GEISS R., ADAMS A., BARRON J. T., KAINZ F., CHEN J., LEVOY M.: Burst photography for high dynamic range and low-light imaging on mobile cameras. *ACM Transactions on Graphics (ToG) 35*, 6 (2016), 1–12. 6

[JK21] JO Y., KIM S. J.: Practical single-image super-resolution using look-up table. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2021), pp. 691–700. 2, 5

[KB15] KINGMA D., BA J.: Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)* (San Diega, CA, USA, 2015). 6

[KY23] KIM S., YOU J.: Lossless lut compressions for image enhancement. *JOURNAL OF SEMICONDUCTOR TECHNOLOGY AND SCIENCE 23*, 3 (2023), 162–175. 2

[LCCX22] LI J., CHEN C., CHENG Z., XIONG Z.: Mulut: Cooperating multiple look-up tables for efficient image super-resolution. In *European Conference on Computer Vision* (2022), Springer, pp. 238–256. 2

[LDL*23] LIU G., DING Y., LI M., SUN M., WEN X., WANG B.: Reconstructed convolution module based look-up tables for efficient image super-resolution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2023), pp. 12217–12226. 2

[LLL20] LIU E., LI S., LIU S.: Color enhancement using global parameters and local features learning. In *Asian Conference on Computer Vision* (2020). URL: https://api.semanticscholar.org/CorpusID:229648539. 2

[LMZ*21] LIU R., MA L., ZHANG J., FAN X., LUO Z.: Retinex-inspired unrolling with cooperative prior architecture search for low-light image enhancement. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (2021), pp. 10561–10570. 2

[LWW*23] LI W., WU G., WANG W., REN P., LIU X.: Fastllve: Real-time low-light video enhancement with intensity-aware look-up table. In *Proceedings of the 31st ACM International Conference on Multimedia* (2023), pp. 8134–8144. 2

[LYFQ23] LIU C., YANG H., FU J., QIAN X.: 4d lut: learnable context-aware 4d lookup table for image enhancement. *IEEE Transactions on Image Processing 32* (2023), 4742–4756. 1, 2

[LZC*21] LIANG J., ZENG H., CUI M., XIE X., ZHANG L.: 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* (2021), pp. 653–661. 5, 6

[MMM*20] MORAN S., MARZA P., MCDONAGH S., PARISOT S., SLABAUGH G.: Deeplpf: Deep local parametric filters for image enhancement. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (2020), pp. 12826–12835. 6, 7

[MZZL22] MA C., ZHANG J., ZHOU J., LU J.: Learning series-parallel lookup tables for efficient image super-resolution. In *European Conference on Computer Vision* (2022), Springer, pp. 305–321. 2, 4

[PGM*19] PASZKE A., GROSS S., MASSA F., LERER A., BRADBURY J., CHANAN G., KILLEEN T., LIN Z., GIMELSHEIN N., ANTIGA L., DESMAISON A., KÖPF A., YANG E., DEVITO Z., RAISON M., TEJANI A., CHILAMKURTHY S., STEINER B., FANG L., BAI J., CHINTALA S.: Pytorch: An imperative style, high-performance deep learning library. *ArXiv abs/1912.01703* (2019). URL: https://api.semanticscholar.org/CorpusID:202786778. 6

[PYL*21] PAN Z., YUAN F., LEI J., LI W., LING N., KWONG S.: Miegan: Mobile image enhancement via a multi-module cascade neural network. *IEEE Transactions on Multimedia 24* (2021), 519–533. 1, 2

[SCH*23] SHI T., CHEN C., HE Y., SONG W., HAO A.: Rgb and lut based cross attention network for image enhancement. In *BMVC* (2023), pp. 348–350. 2

[SX22] SHE D., XU K.: An image-to-video model for real-time video enhancement. In *Proceedings of the 30th ACM International Conference on Multimedia* (2022), pp. 1837–1846. 2

[WLP*21] WANG T., LI Y., PENG J., MA Y., WANG X., SONG F., YAN Y.: Real-time image enhancer via learnable spatial-aware 3d lookup tables. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (2021), pp. 2471–2480. 1, 2, 4, 6

[WXL22] WANG H., XU K., LAU R. W.: Local color distributions prior for image enhancement. In *European Conference on Computer Vision* (2022), Springer, pp. 343–359. 1, 2

[WYX11] WANG B., YU Y., XU Y.-Q.: Example-based image color and tone style enhancement. *ACM Transactions on Graphics (TOG) 30*, 4 (2011), 1–12. 2

[WZF*19] WANG R., ZHANG Q., FU C.-W., SHEN X., ZHENG W.-S., JIA J.: Underexposed photo enhancement using deep illumination estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (2019), pp. 6849–6857. 1, 2, 6, 7

[WZHL13] WANG S., ZHENG J., HU H.-M., LI B.: Naturalness preserved enhancement algorithm for non-uniform illumination images. *IEEE transactions on image processing 22*, 9 (2013), 3538–3548. 5

[YJJ*22] YANG C., JIN M., JIA X., XU Y., CHEN Y.: Adaint: Learning adaptive intervals for 3d lookup tables on real-time image enhancement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (2022), pp. 17522–17531. 1, 2, 4, 6, 7

[YJX*22] YANG C., JIN M., XU Y., ZHANG R., CHEN Y., LIU H.: Seplut: Separable image-adaptive lookup tables for real-time image enhancement. In *European Conference on Computer Vision* (2022), Springer, pp. 201–217. 1, 2, 6, 7

[YZW*16] YAN Z., ZHANG H., WANG B., PARIS S., YU Y.: Automatic photo adjustment using deep neural networks. *ACM Transactions on Graphics (TOG) 35*, 2 (2016), 1–15. 1, 2

© 2025 Eurographics - The European Association
for Computer Graphics and John Wiley & Sons Ltd.

[ZCL*20] ZENG H., CAI J., LI L., CAO Z., ZHANG L.: Learning image-adaptive 3d lookup tables for high performance photo enhancement in real-time. *IEEE Transactions on Pattern Analysis and Machine Intelligence 44*, 4 (2020), 2058–2073. 1, 2, 4, 5, 6, 7

[ZLY*22] ZHANG X., LU C., YAN D., DONG W., YAN Q.: Dualbln: Dual branch lut-aware network for real-time image retouching. In *Proceedings of the Asian Conference on Computer Vision* (2022), pp. 2139–2155. 1, 2, 5, 6, 7

[ZNZ19] ZHANG Q., NIE Y., ZHENG W.: Dual illumination estimation for robust exposure correction. *Computer Graphics Forum 38* (2019). URL: https://api.semanticscholar.org/CorpusID:204961229. 5

[ZYX*18] ZHANG Q., YUAN G., XIAO C., ZHU L., ZHENG W.-S.: High-quality exposure correction of underexposed photos. In *Proceedings of the 26th ACM international conference on Multimedia* (2018), pp. 582–590. 5

[ZZG19] ZHANG Y., ZHANG J., GUO X.: Kindling the darkness: A practical low-light image enhancer. In *Proceedings of the 27th ACM international conference on multimedia* (2019), pp. 1632–1640. 2

© 2025 Eurographics - The European Association
for Computer Graphics and John Wiley & Sons Ltd.