4K-Resolution Photo Exposure Correction at 125 FPS with ∼8K Parameters

Yijie Zhou[1] Chao Li[1] Jin Liang[1] Tianyi Xu[1] Xin Liu[2,3,] Jun Xu[1,4,*]
[1]Nankai University [2]Tianjin University [3]Lappeenranta-Lahti University of Technology
[4]Guangdong Provincial Key Laboratory of Big Data Computing, CUHK (Shenzhen)

Abstract

The illumination of improperly exposed photographs has been widely corrected using deep convolutional neural networks or Transformers. Despite with promising performance, these methods usually suffer from large parameter amounts and heavy computational FLOPs on high-resolution photographs. In this paper, we propose extremely light-weight (with only ∼8K parameters) Multi-Scale Linear Transformation (MSLT) networks under the multi-layer perception architecture, which can process 4K-resolution sRGB images at 125 Frame-Per-Second (FPS) by a Titan RTX GPU. Specifically, the proposed MSLT networks first decompose an input image into high and low frequency layers by Laplacian pyramid techniques, and then sequentially correct different layers by pixel-adaptive linear transformation, which is implemented by efficient bilateral grid learning or 1×1 *convolutions. Experiments on two benchmark datasets demonstrate the efficiency of our MSLTs against the state-of-the-arts on photo exposure correction. Extensive ablation studies validate the effectiveness of our contributions. The code is available at* https://github.com/Zhou-Yijie/MSLTNet.

1. Introduction

The prevalence of smartphones with cameras encourages people to take snapshots of their daily life like photographers. However, inaccurate setting of shutter speed, focal-aperture ratio and/or ISO value may bring improper exposure to the captured photographs with degradation on visual quality [4]. To adjust the photo exposure properly for visually appealing purpose, it is essential to develop efficient exposure correction methods for edge devices.

In last decades, low-light enhancement methods [10, 25, 38] and over exposure correction methods [3, 9] have been proposed to adjust the brightness of under-exposed and over-exposed images, respectively. However, low-light enhancement methods could hardly correct over-exposed images while over-exposure correction methods would fail on under-exposed images [4]. High dynamic range (HDR) tone-mapping methods [18, 19, 31, 33] can also adjust improper illumination of the contents to some extent, but mainly enhance local details in improperly-exposed areas along with dynamic range reduction. In the end, all these methods are not suitable for exposure correction, which requires globally adjustment on improper exposure in images.

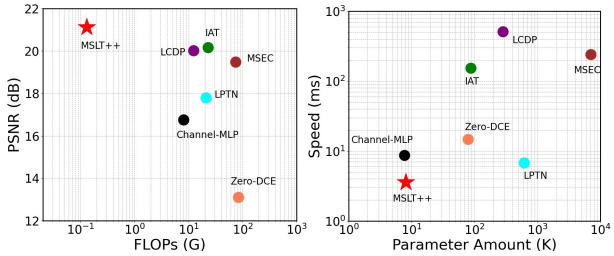

Figure 1. **Comparison of the proposed MSLT++ and state-of-the-art exposure correction methods** on the ME dataset [4]. Left: comparison of PSNR results and computational costs (FLOPs). Right: comparison of speed (inference time on a 1024×1024 sRGB image) and parameter amounts.

Recently, there emerges several exposure correction methods based on Convolutional Neural Networks (CNN) [4] or Transformer [13]. For example, Multi-Scale Exposure Correction (MSEC) [4] performs hierarchical exposure correction with Laplacian pyramid techniques [6, 15, 28] and the UNet architecture [39]. Later, the work of [48] exploits the Local Color Distributions Prior (LCDP) to locate and enhance the improperly exposed region. The attention-based Illumination Adaptive Transformer (IAT) [13] estimates the parameters related to the Image Signal Processor (ISP) under the Transformer architecture [47]. Despite with promising performance, these exposure correction CNNs or Transformers are limited by huge parameter amounts and computational costs [4, 13].

To produce visually pleasing results while still improving the model efficiency, in this paper, we propose extremely light-weight Multi-Scale Linear Transformation (MSLT) networks for high-resolution image exposure correction. Specifically, we first decompose the input image into high-frequency and low-frequency layers via Laplacian

*Corresponding author: csjunxu@nankai.edu.cn.

pyramid techniques [6, 15, 28] to perform coarse-to-fine exposure correction. We then design simple linear transformation networks to progressively correct these layers, consuming small parameter amounts and computational costs. For the low-frequency layer, we adopt the bilateral grid learning (BGL) framework [20, 51, 53] to learn pixel-wise affine transformation between improper and proper exposed image pairs. To learn context-aware transformation coefficients in BGL, we propose a parameter-free Context-aware Feature Decomposition (CFD) module and extend it for multi-scale affine transformation. For the high-frequency layers, we simply learn pixel-wise correction masks by two channel-wise 1×1 convolutional layers.

Benefited by using channel-wise multi-layer perception (MLP) for coarse-to-fine exposure correction, our largest network MSLT++ has 8,098 parameters, while requiring only 0.14G and 3.67ms to process a $1024 \times 1024 \times 3$ image with a RTX GPU. As a comparison, the parameter amounts of CNN-based MSEC [4], LCDP [48] and transformer-based IAT [13] are $\sim$7,015K, $\sim$282K and $\sim$86.9K, respectively, while the corresponding FLOPs/speed are 73.35G/240.46ms, 17.33G/507.67ms and 22.96G/153.96ms, respectively. Experiments on two benchmark datasets [4, 8] show that our MSLTs achieve better quantitative and qualitative performance than state-of-the-art exposure correction methods. A quick glimpse of comparison on the ME dataset is shown in Figure 1.

Our main contributions are summarized as follows:

- We develop Multi-Scale Linear Transformation networks with at most 8,098 parameters, which run at most 125 FPS on 4K-resolution ($3840 \times 2160 \times 3$) images with effective exposure correction performance.
- To accelerate the multi-scale decomposition, we design a bilateral grid network (BGN) to pixel-wisely correct the exposure of low-frequency layer. Here, we implement BGN via a channel-wise MLP, rather than CNNs or Transformers, to endow our MSLTs with small parameter amounts and computational costs.
- We propose a Context-aware Feature Decomposition (CFD) module to learn hierarchical transformation coefficients in our BGN for effective exposure correction.

2. Related Work

2.1. Image Exposure Correction Methods

The exposure correction task is similar but different to the tasks of low-light image enhancement [10, 25], over-exposure correction [3, 9], and HDR tone mapping [18, 19, 31, 33]. As far as we know, the work of MSEC [4] is among the first deep learning based method for exposure correction. It decomposes an image into high-frequency and low-frequency parts, and progressively corrects the exposure errors. However, MSEC has over 7M parameters and is not efficient enough on high-resolution images. The Local Color Distributions Prior (LCDP) [48] exploits the local color distributions to uniformly tackle the under-exposure and over-exposure, with about 282K parameters and requires huge computational costs, *e.g.*, 17.33G FLOPs, to process a $1024 \times 1024 \times 3$ image. The Transformer based Illumination-Adaptive-Transformer (IAT) [13] has about 86.9K parameters, but suffering from large computational costs and slow inference speed on high-resolution images.

In this paper, we propose light-weight and efficient Multi-Scale Linear Transformation (MSLT) networks, which at most have 8,098 parameters and run at 125 FPS to correct 4K resolution images with improper exposures.

2.2. Image Processing MLPs

Multi-layer perceptions (MLPs) [40] play an important role in visual tasks before the prosperity of convolutional neural networks (CNNs) and Transformers. MLP based networks have attracted the attention of researchers again for its simplicity. The method of MLP-Mixer [41] is a purely MLP-based network without convolutions or self-attention. Later, ResMLP [42] is proposed using only linear layers and GELU non-linearity. The work of gMLP [32] utilizes MLPs with gating to achieve comparable results with Transformers [17, 43] on image classification [14]. Ding et al. [16] proposed a re-parameterization technique to boost the capability of MLP on image classification. The recently developed MAXIM [44] is a multi-axis MLP based network for general image processing tasks. In this paper, we develop an extremely efficient exposure correction network, which mainly utilizes channel-wise (not spatial-wise) MLPs to globally perceive the exposure information of the image.

2.3. Light-weight Image Enhancement Networks

In pursuit for light-weight and efficient models, one naive way is to apply the model at a low-resolution input and then resize the output into high-resolutions. But the high-frequency details would be lost. To this end, the Laplacian Pyramid decomposition [4, 6] is used to preserve high-frequency information. A further approach is to learn an approximate operator at downsampled inputs and then apply this operator to the original image [11, 20, 34]. Such approximate operators are usually simple and efficient. Later, this approximation insight is also studied by bilateral grid learning [12], to accelerate diverse image processing methods on the tasks of image enhancement [20], image dehazing [53], and stereo matching [51], *etc*.

In this paper, we design light-weight and efficient image exposure correction networks with Laplacian pyramid technique and bilateral grid learning framework. Differently, our bilateral grid network is purely implemented by

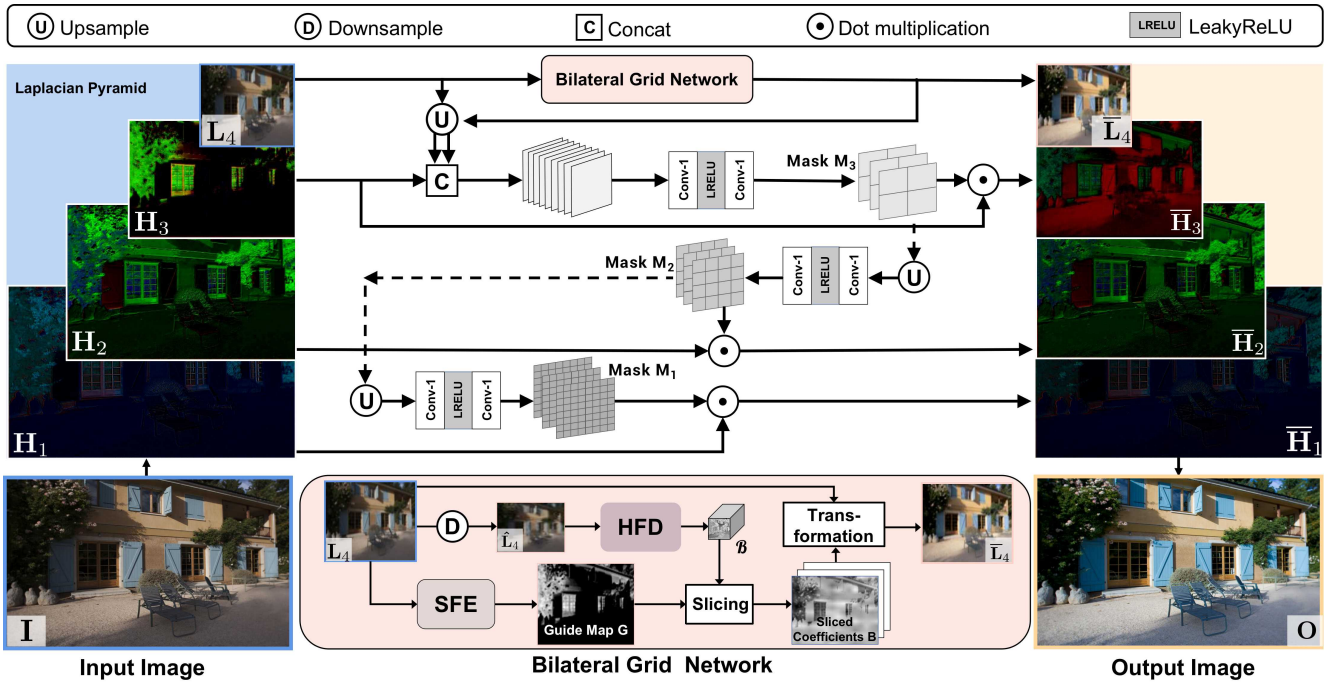

Figure 2. **Overview of our Multi-Scale Linear Transformation (MSLT) network** with $n = 4$. Given an input image $\mathbf{I} \in \mathbb{R}^{H\times W\times 3}$ with improper exposure, our MSLT firstly decomposes the image $\mathbf{I}$ into high frequency layers $\{\mathbf{H}_i \in \mathbb{R}^{\frac{H}{2^{i-1}}\times\frac{W}{2^{i-1}}\times 3} | i = 1, 2, 3\}$ and a low frequency layer $\mathbf{L}_4$ by Laplacian pyramid decomposition. The $\mathbf{L}_4$ is corrected by the proposed Bilateral Grid Network: 1) the $\mathbf{L}_4$ is input to Self-modulated Feature Extraction (SFE) module to obtain a guidance map $\mathbf{G}$, 2) the $\mathbf{L}_4$ is downsampled to $\hat{\mathbf{L}}_4$ of size $48 \times 48 \times 3$, which is used to learn the 3D bilateral grid of affine coefficients $\mathcal{B}$ by the Hierarchical Feature Decomposition (HFD) module, 3) with the guidance map $\mathbf{G}$, the coefficients $\mathcal{B}$ are sliced to produce a 2D grid of coefficients $\mathbf{B}$, which is used to pixel-wisely correct the $\mathbf{L}_4$. The high frequency layers $\{\mathbf{H}_i | i = 1, 2, 3\}$ are corrected by learning corresponding masks via two 1×1 convolutions. Finally, the corrected low/high-frequency layers are reconstructed to output the exposure corrected image $\mathbf{O}$. The SFE and HFD modules are detailed in Figure 3.

channel-wise MLP, consuming much less parameters and computational costs than CNNs and Transformers.

3. Proposed Method

3.1. Network Overview

As illustrated in Figure 2, our Multi-Scale Linear Transformation (MSLT) network for exposure correction is consisted of four close-knit parts introduced as follows.

Multi-Scale Image Decomposition. As suggested in [4], the coarse-to-fine architecture is effective for the exposure correction task. Given an input image $\mathbf{I} \in \mathbb{R}^{H\times W\times 3}$, we employ the Laplacian pyramid technique [6] to decompose the image $\mathbf{I}$ into a sequence of $n-1$ high-frequency layers $\{\mathbf{H}_i \in \mathbb{R}^{\frac{H}{2^{i-1}}\times\frac{W}{2^{i-1}}\times 3} | i = 1, ..., n-1\}$ and one low-frequency layer $\mathbf{L}_n \in \mathbb{R}^{\frac{H}{2^{n-1}}\times\frac{W}{2^{n-1}}\times 3}$.

Low-Frequency Layer Correction is performed by learning pixel-adaptive exposure correction under the bilateral gird learning framework [51]. To learn meaningful bilateral grid of affine coefficients, we propose a parameter-free Context-aware Feature Decomposition (CFD) module and extend it to a hierarchical version for better performance.

High-Frequency Layers Correction is implemented by multiplying each layer pixel-wisely with a comfortable mask, predicted by two consecutive 1×1 convolutions.

Final Reconstruction is performed by Laplacian reconstruction [6] on the exposure-corrected layers of different frequencies to output a well-exposed $\mathbf{O} \in \mathbb{R}^{H\times W\times 3}$.

3.2. Low-Frequency Layer Correction

The illumination information is mainly in low-frequency [4], so we pay more attention to the low-frequency layer $\mathbf{L}_n$ for effective exposure correction. Inspired by its success on efficient image processing [11, 51, 53], we employ the bilateral grid learning [12] to correct the exposure of low-frequency layer $\mathbf{L}_n$. As shown in Figure 2, our Bilateral Grid Network contains three components: 1) learning the guidance map, 2) estimating the bilateral grid of affine coefficients, and 3) coefficients transformation.

Learning guidance map. We propose a Self-modulated Feature Extraction (SFE) module to learn the guidance map G with the same size as $\mathbf{L}_n$. As shown in Figure 3 (b), the SFE module uses two 1×1 convolutions and global average pooling (GAP) to modulate the extracted feature map.

Estimating bilateral grid of affine coefficients. We first downsample the low-frequency layer $\mathbf{L}_n$ to $\hat{\mathbf{L}}_n \in$

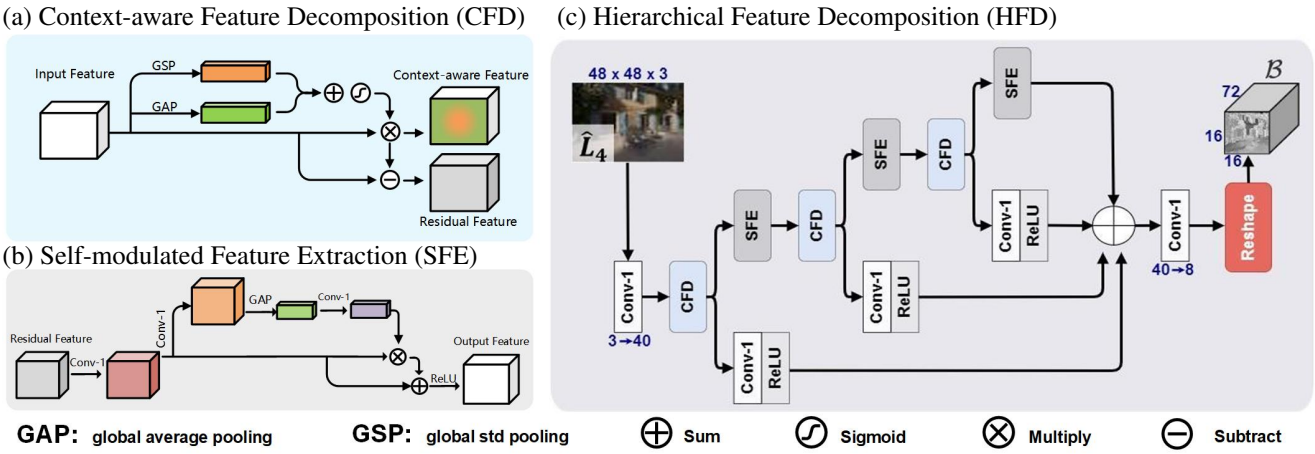

Figure 3. **Architectures of our CFD, SFE, and HFD modules**. Our HFD (c) mainly contains of three pairs of CFD (a) and SFE (b) modules. For the downsampled low-frequency layer $\hat{\mathbf{L}}_4 \in \mathbb{R}^{48\times48\times3}$, we first use a 1×1 convolution to increases its channel dimension from 3 to 40. Then our CFD separates the feature into context-aware feature and residual feature, which are subsequently refined by 1×1 convolution followed by a ReLU function and an SFE module, respectively. The three hierarchical context-aware feature maps and the residual feature from the third SFE module are summed and fused by a 1×1 convolution, with decreased channel dimension from 40 to 8. Finally, the fused feature is reshaped into a 3D bilateral grid of affine transformation coefficients $\mathcal{B} \in \mathbb{R}^{16\times16\times72}$.

$\mathbb{R}^{48\times48\times3}$. The mean and standard deviation (std) of each channel roughly reflect the brightness and contrast, respectively, of that feature map [46]. Exploiting these information is useful to estimate the bilateral grid of affine coefficients for exposure correction. For this, we propose a parameter-free Context-aware Feature Decomposition (CFD) module to extract the context-aware feature and the residual feature. As shown in Figure 3 (a), the context-aware feature is obtained by multiplying the original feature channel-wisely with the sum of mean and std calculated by global average pooling and global std pooling, respectively. We extend CFD to a Hierarchical Feature Decomposition (HFD) module by cascading three parameter-sharing CFD and SFE modules, as shown in Figure 3 (c). The goal is to learn a 3D bilateral grid of affine coefficients $\mathcal{B} \in \mathbb{R}^{16\times16\times72}$, in which every 12 channels representing a 3×4 affine matrix. We implement our HFD module by channel-wise 1×1 convolutions to perform spatial consistent and pixel-adaptive brightness adjustment. Three 1×1 convolutions shared parameters before ReLU, with small parameter amounts and computational costs (Figure 3 (c)).

Coefficients transformation. With the guidance map $\mathbf{G} \in \mathbb{R}^{\frac{H}{2^{n-1}}\times\frac{W}{2^{n-1}}}$, we upsample the 3D bilateral grid of affine coefficients $\mathcal{B} \in \mathbb{R}^{16\times16\times72}$ back to a 2D bilateral grid of coefficients $\mathbf{B} \in \mathbb{R}^{\frac{H}{2^{n-1}}\times\frac{W}{2^{n-1}}}$ and then correct the low-frequency layer $\mathbf{L}_n$ by tri-linear interpolation [11]. Each cell of grid $\mathbf{B}$ contains a 3×4 matrix for pixel-adaptive affine transformation. At last, the affine transformations in $\mathbf{B}$ will act on the low-frequency layer $\mathbf{L}_n$ pixel-by-pixel to obtain the exposure-corrected low-frequency layer $\overline{\mathbf{L}}_n$.

3.3. High-Frequency Layers Correction

With the corrected low-frequency layer, now we correct the high-frequency layers $\{\mathbf{H}_i|i=1,...,n-1\}$ in the order of $i=n-1,...,1$. The correction is implemented by multiplying each high-frequency layer $\mathbf{H}_i$ with a comfortable mask in an element-wise manner. Each mask is predicted by a small MLP consisted of two 1×1 convolutional layers with a LeakyReLU [36] between them.

To correct the high-frequency layer $\mathbf{H}_{n-1}$, we first concatenate it with the upsampled low-frequency layer $\mathbf{L}_n$ and the upsampled corrected layer $\overline{\mathbf{L}}_n$ along the channel dimension. Then the concatenated layers are put into the small MLP to predict the mask $\mathbf{M}_{n-1}$. Since the concatenated layers have nine channels, we set the numbers of input and output channels as nine for the first 1×1 convolutional layer in the small MLP, and set those of the second 1×1 convolutional layer as nine and three, respectively. By element-wisely multiplying high-frequency layer $\mathbf{H}_{n-1}$ with the mask $\mathbf{M}_{n-1}$, we obtain the exposure corrected high-frequency layer $\overline{\mathbf{H}}_{n-1}$. Besides, the predicted mask $\mathbf{M}_{n-1}$ will be reused as the input of the MLP in the correction of next high-frequency layer for mask prediction.

For $i=n-2,...,1$, we upsample the mask $\mathbf{M}_{i+1}$ output in previous layer into the MLP of current layer to predict a new mask $\mathbf{M}_i$. Unlike the MLP in predicting the mask $\mathbf{M}_{n-1}$, the MLPs for predicting masks $\{\mathbf{M}_{i+1}|i=n-2,...,1\}$ have three input and output channels for both two 1×1 convolutional layers. Similarly, each mask $\mathbf{M}_i$ is multiplied with the high-frequency layer $\mathbf{H}_i$ element-wisely to output the exposure-corrected high-frequency layer $\overline{\mathbf{H}}_i$. Finally, we reconstruct the output image $\mathbf{O}$ from the exposure-corrected low/high-frequency layers $\{\overline{\mathbf{H}}_1,...,\overline{\mathbf{H}}_{n-1},\overline{\mathbf{L}}_n\}$. Here, we set $n=4$ for our MSLT.

To study the effect of exposure correction by our MSLT, we convert the input image $\mathbf{I}$ and output image $\mathbf{O}$ from the sRGB color space to the CIELAB color space. We denote the lightness channels of $\mathbf{I}$ and $\mathbf{O}$ as $\mathbf{I}_L$ and $\mathbf{O}_L$,

(a) Input Image (b) Corrected Image (c) Ground Truth (d) Heatmap of Correction Strength

Figure 4. **Heatmap of Correction Strength** in our MSLT. (a) the under/over exposed input images. (b) the corrected images by our MSLT. (c) the "ground truth" images. (d) the heatmaps of correction strength described in §3.3. The values in $(0, 1]$ (or $[-1, 0)$) indicate brightness enhancement (or shrinkage). Darker color indicates larger absolute values and stronger correction strength in brightness.

respectively, and compute their difference residual $\mathbf{R} = \mathbf{O}_L - \mathbf{I}_L$. Denote $\mathbf{R}_{max}$ as the maximum absolute value of $\mathbf{R}$, *i.e.*, $\mathbf{R}_{max} = \max|\mathbf{R}|$. The residual $\mathbf{R}$ is normalized into $[-1, 1]$ by $\mathbf{R}/\mathbf{R}_{max}$ to represent pixel-wise correction strength, where $(0, 1]$ (or $[-1, 0)$) indicates brightness enhancement (or shrinkage). The heatmap of correction strength, as shown in Figure 4, exhibits close relationship to the context of input $\mathbf{I}$. This demonstrates that our MSLT indeed performs pixel-adaptive exposure correction.

3.4. Network Acceleration

The proposed MSLT network implements Laplacian pyramid decomposition via standard Gaussian kernel [5], which is not optimized in current deep learning frameworks [2, 37]. To speed up our MSLT, we replace the Gaussian kernel with learnable 3×3 convolution kernel, which is highly optimized by the PyTorch framework [29]. By introducing 3×3 convolutional kernels into our MSLT, we break its fully MLP architecture with more parameters and computational costs. The speed of our MSLT is clearly improved from 4.34ms to 4.07ms on 1024×1024 sRGB images and from 19.27ms to 11.04ms on 3840×2160 sRGB images. We call this variant network as MSLT+. Through experiments, we also observe that the learnable 3×3 convolutional kernels can perform adaptive decomposition for each image to better correct the exposure of different layers.

Considering that the high-frequency layer $\mathbf{H}_1$ is of the largest resolution with the finest information among all layers, it is worth to study whether it is feasible to avoid the correction of this layer for further model acceleration. In fact, even without correcting $\mathbf{H}_1$, the learnable convolution kernels in MSLT+ would still produce adaptive Laplacian pyramid decomposition to compensate the overall exposure correction performance. To illustrate this point, we remove the mask prediction MLP in correcting the high-frequency layer $\mathbf{H}_1$ in MSLT+, and directly using the $\mathbf{H}_1$ together with other corrected layers $\{\overline{\mathbf{L}}_4, \overline{\mathbf{H}}_3, \overline{\mathbf{H}}_2\}$ for final reconstruction. We call this variant network as MSLT++. As shown in Figure 5, on two under-exposed and over-exposed images, we observe similar visual quality of the exposure-corrected images by MSLT, MSLT+, and MSLT++. This indicates that removing the correction of the high-frequency layer $\mathbf{H}_1$ potentially influences little our MSLT++ on exposure correction, and brings additional reduction on the computational costs and inference time of MSLT+. For example, our MSLT++ improves the speed of MSLT+ from 4.07ms to 3.67ms on 1024×1024 sRGB images and from 11.04ms to 7.94ms on 3840×2160 (4K) sRGB images.

Figure 5. **Corrected images** by our MSLT, MSLT+ and MSLT++.

3.5. Implementation Details

Our MSLT networks are optimized by Adam [26] with β_1=0.9 and β_2=0.999, using the mean-square error (MSE) loss function. The initial learning rate is set as 1×10^{-3} and is decayed to 1×10^{-7} with cosine annealing schedule for every 5 epochs. The batch size is 32. For the training set, we randomly crop the images into 512×512 patches. Here, we have $n = 4$ Laplacian pyramid layers, the 64×64 low-frequency layers are downsampled to 48×48 for learning accurate 3D bilateral grid of affine coefficients. Our MSLT networks, implemented by PyTorch [29] and MindSpore [1], are trained in 200 epochs on a Titan RTX GPU, which takes about 18 hours.

Table 1. **Quantitative results of different methods** on the ME dataset [4].We take the correctly exposed images rendered by five experts as the ground truth images, respectively. The best, second best and third best results are highlighted in red, blue and **bold**, respectively.

Method	Expert A			Expert B			Expert C			Expert D			Expert E			Average		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS	PSNR↑	SSIM↑	LPIPS↓
LPTN [30]	17.50	0.746	0.2236	18.28	0.789	0.2067	18.08	0.780	0.2121	17.70	0.770	0.2154	17.45	0.768	0.2235	17.80	0.771	0.2519
Zero-DCE [21]	12.16	0.658	0.3103	13.16	0.725	0.2649	12.61	0.694	0.3022	13.47	0.720	0.2678	14.18	0.749	0.2643	13.11	0.709	0.2819
SCI [35]	16.11	0.737	0.2064	17.15	0.805	0.1725	16.36	0.764	0.2079	16.51	0.766	0.1899	16.09	0.761	0.2125	16.44	0.767	0.1978
MSEC w/o adv [4]	19.16	0.796	0.1802	20.10	0.815	0.1724	20.21	0.817	0.1805	18.98	0.796	0.1816	18.98	0.805	0.1911	19.48	0.806	0.1812
MSEC w/ adv [4]	19.11	0.784	0.1861	19.96	0.813	0.1802	20.08	0.815	0.1875	18.87	0.793	0.1901	18.86	0.803	0.1999	19.38	0.802	0.1888
LCDP [48]	**20.59**	**0.814**	**0.1540**	21.95	0.845	**0.1399**	**22.30**	**0.856**	**0.1448**	20.22	0.825	**0.1526**	20.07	0.827	**0.1617**	21.02	**0.833**	**0.1506**
IAT [13]	19.63	0.780	0.1962	21.21	0.816	0.1771	21.21	0.820	0.1828	19.58	0.805	0.1871	19.21	0.797	0.1947	20.17	0.804	0.1876
FECNet [24]	**20.73**	**0.815**	0.1861	**22.87**	**0.861**	0.1636	**22.92**	**0.858**	0.1700	**20.67**	**0.835**	0.1808	**20.22**	**0.829**	0.1913	**21.48**	**0.839**	0.1783
Channel-MLP	16.21	0.708	0.2577	17.48	0.784	0.2255	16.96	0.741	0.2421	16.59	0.746	0.2442	16.53	0.750	0.2481	16.75	0.746	0.2435
MSLT	**20.21**	**0.805**	**0.1724**	22.47	**0.864**	0.1460	22.03	**0.844**	**0.1639**	20.33	**0.830**	**0.1637**	20.04	**0.832**	**0.1758**	21.02	**0.835**	**0.1644**
MSLT+	**20.21**	0.799	**0.1677**	**22.49**	0.858	**0.1410**	**22.09**	0.840	**0.1588**	**20.59**	**0.828**	**0.1585**	**20.53**	**0.830**	**0.1687**	**21.18**	0.831	**0.1589**
MSLT++	20.09	0.797	0.1745	**22.55**	**0.860**	**0.1452**	22.07	0.838	**0.1639**	**20.54**	0.826	0.1640	**20.36**	0.828	0.1762	**21.12**	0.830	0.1648

Table 2. **Quantitative results of different methods** on SICE dataset [50]. The best, second best and third best results are highlighted in red, blue and **bold**, respectively.

Method	PSNR↑	SSIM↑	LPIPS↓
LPTN [30]	15.46	0.609	0.4150
Zero-DCE [21]	12.05	0.592	0.4439
SCI [35]	12.85	0.569	0.3776
MSEC w/o adv [4]	17.86	**0.664**	**0.3761**
MSEC w/ adv [4]	17.67	**0.664**	0.3875
LCDP [48]	18.50	0.609	0.4749
IAT [13]	**18.55**	**0.672**	**0.3325**
FECNet [24]	**19.39**	**0.691**	0.3939
Channel-MLP	15.21	0.546	0.5370
MSLT	18.22	0.661	**0.3557**
MSLT+	18.32	0.642	0.3883
MSLT++	**18.69**	0.653	0.3900

4. Experiments

4.1. Dataset and Metric

Dataset. We evaluate our MSLT networks on two benchmark datasets: the **ME** dataset [4] and the **SICE** dataset [8].

The **ME** dataset is built upon the MIT-Adobe FiveK dataset [7], from which each raw-sRGB image was rendered with five relative exposure values $\{-1.5, -1, 0, +1, +1.5\}$ to mimic improperly exposed images. Five expert photographers (A-E) manually retouched the raw-sRGB images to produce the correctly exposed images ("ground truths"). As suggested in [4], we use the images retouched by Expert C as the training targets. This dataset contains 17,675 training images, 750 validation images, and 5,905 test images.

The **SICE** dataset is randomly divided into 412, 44, and 100 sequences as train, validation, and test sets respectively. We set the second and the last second images in each sequence as the under or over exposed inputs, as suggested by [23]. For each image in the training set, we randomly crop 30 patches of size 512×512 for training.

Evaluation metrics. We use three evaluation metrics of Peak Signal-to-noise Ratio (PSNR), Structural Similarity Index (SSIM) [49], and Learned Perceptual Image Patch Similarity (LPIPS) [52] to measure the distance between the exposure corrected images and the "ground truths". For LPIPS, we use the AlexNet [27] to extract feature maps.

4.2. Comparison Results

We compare our MSLTs with four exposure correction methods (MSEC [4], LCDP [48], FECNet [24] and IAT [13]), two enhancement methods (Zero-DCE [21] and SCI [35]), and one image translation method (LPTN [30]). To validate the design of our MSLTs with MLPs, we also compare with a plain Channel-MLP with 7,683 parameters (more details are provided in the *Supplementary File*).

Table 3. **Comparison of model size, computational costs, and speed (ms)**. The speed is test on a Titan RTX GPU. MSEC indicates "MSEC w/o adv". The best, second best and third best results are highlighted in red, blue and **bold**, respectively.

Method	# Param (K)	FLOPs (G)		Speed (ms)	
		1024×1024	3840×2160	1024×1024	3840×2160
LPTN [30]	616.215	21.55	170.46	6.90	55.96
Zero-DCE [21]	79.416	83.27	658.71	22.98	197.36
SCI [35]	**0.348**	0.55	4.38	6.55	48.37
MSEC [4]	7015.449	73.35	579.98	240.46	2250.74
LCDP [48]	281.758	17.33	127.79	507.67	3305.73
IAT [13]	86.856	22.96	182.59	153.96	1226.73
FECNet [24]	151.97	94.61	748.35	139.12	1277.24
Channel-MLP	**7.683**	8.05	63.73	8.69	66.87
MSLT	**7.594**	**0.08**	**0.42**	**4.34**	**19.27**
MSLT+	8.098	**0.17**	**1.10**	**4.07**	**11.04**
MSLT++	8.098	**0.14**	**0.88**	**3.67**	**7.94**

Objective results. For the ME and SICE datasets, as shown in Table 1 and Table 2, our MSLTs obtain better PSNR, SSIM and LPIPS results than LPTN, Zero-DCE, SCI and Channel-MLP. On ME, our MSLTs achieve better results than MSEC and IAT, and are comparable to LCDP and FECNet. On SICE, our MSLTs achieve comparable performance with MSECs and a little inferior results to IAT and FECNet. However, our MSLTs exhibit higher efficiency than all the other comparison methods, as shown in Table 3.

Speed. In order to be deployed into practical application, the inference speed is put forward high requirements. To measure the speed of the models, we randomly generate an "image" of size $1024 \times 1024 \times 3$ or $3840 \times 2160 \times 3$, repeat the inference test for 100 times, and average the results as the speed of comparison methods. The speed tests are all run on a Titan RTX GPU. The results are shown in Table 3. One can see that the inference speed of our MSLT++ on a $1024 \times 1024 \times 3$ tensor is 3.67 ms, much faster than all the other methods. On a high-resolution tensor of size $3840 \times 2160 \times 3$, our MSLT++ reaches an inference speed of 7.94ms, also faster than the other comparison methods.

Visual quality. The ultimate goal of exposure correction task is to restore more realistic images and improve the visual experience of the observer. Thus, the visual quality of images is also an important factor to consider. In Fig-

Figure 6. **Visual quality comparison of exposure corrected images by different methods**. 1st and 2nd rows: visual results on one over-exposed image from the ME dataset [4]. 3rd and 4th rows: visual results on one under-exposed image from the SICE dataset [8].

ure 6, we provide the corrected images of "Manor" in ME dataset and "Mountain" in SICE dataset by the comparison methods, respectively. More visual comparison results can be found in the *Supplementary File*. On over-exposed "Manor" image, one can see that Zero-DCE, SCI, LPTN and Channel-MLP are hardly able to weaken the exposure. Our MSLTs generate better details in clouds, walls and lawns than those of LCDP and IAT. The corrected image by MSEC has too high contrasts to be realistic. On under-exposed "Mountain", our MSLTs outperform the others in terms of overall brightness and details of the green leaves.

4.3. Ablation Study

Here, we provide detailed experiments of our MSLT on exposure correction to study: 1) the number of Laplacian pyramid layers in our MSLT; 2) how to design the Context-aware Feature Decomposition (CFD) module; 3) the number of CFD modules in our HFD; 4) how to develop the Hierarchical Feature Decomposition (HFD) module in the bilateral grid network; 5) how the correction of high-frequency layers influences our MSLT and MSLT+. All experiments are performed on the ME dataset [4]. The images retouched by five experts are respectively considered as the "ground-truth" images to calculate average PSNR, SSIM and LPIPS values. We compute FLOPs and speed on a 1024×1024 sRGB image. The rows with light shadow indicate the results of our MSLT networks on exposure correction. More results are provided in *Supplementary File*.

1) The number of Laplacian pyramid (LP) layers in our MSLT. The Laplacian pyramid structure is deployed in our MSLT networks to reduce the computational costs and inference time (speed). As shown in Table 4, generally, the Laplacian pyramid with more layers produces smaller low-frequency layer. Since the main costs are paid to this layer, our MSLT will be faster. However, when the number of LP layers is 5, the low-frequency layer is small, which degrades our MSLT network. Besides, the decomposition of 5 LP layers offsets the overall acceleration, and slow down our MSLT for exposure correction. By considering both the performance and inference speed of our MSLT, we set $n = 4$ for the LP decomposition in our MSLT networks.

Table 4. **Results of exposure correction by our MSLT with different number (n) of Laplacian pyramid levels**. "w/o LP" means we do not use Laplacian pyramid.

LP Layers	PSNR ↑	SSIM ↑	LPIPS ↓	# Param	FLOPs (M)	Speed (ms)
w/o LP	21.06	0.830	0.1615	7,448	303.85	6.50
2	20.98	0.835	0.1631	7,594	237.79	4.91
3	20.92	0.828	0.1643	7,594	114.32	4.41
4	21.02	0.835	0.1644	7,594	83.45	4.34
5	20.55	0.825	0.1646	7,594	75.73	4.66

2) How to design the Context-aware Feature Decomposition (CFD) module? In our CFD, we use the mean and standard deviation of each channel to learn the context-aware feature. To demonstrate its effect, we replace this part with Instance Normalization (IN) [45] or Channel Attention (CA) [22], and remain the rest of our MSLT. As shown in Table 5, our CFD achieves highest PSNR and LPIPS among the three methods and it has comparable SSIM with the "IN" version. This shows that the method using mean and standard deviation information of each channel does work.

Table 5. **Results of our MSLT with different variants of CFD module in our HFD**. "CFD": Context-aware Feature Decomposition. "IN": Instance Normalization [45] with feature decomposition. "CA": Channel Attention [22] with feature decomposition.

Variant	PSNR ↑	SSIM ↑	LPIPS ↓	# Param	FLOPs (M)	Speed (ms)
IN	20.82	0.831	0.1652	7,684	83.45	4.28
CA	20.60	0.829	0.1701	7,912	83.45	4.22
CFD	21.02	0.835	0.1644	7,594	83.45	4.34

Table 6. **Results of our MSLT with different number of CFD modules** in the proposed HFD module.

# CFD	PSNR ↑	SSIM ↑	LPIPS ↓	# Param	FLOPs (M)	Speed (ms)
1	20.31	0.824	0.1845	7,594	60.59	3.54
2	20.50	0.826	0.1818	7,594	72.02	3.82
3	21.02	0.835	0.1644	7,594	83.45	4.34
4	20.73	0.832	0.1699	7,594	94.88	4.56
5	20.63	0.827	0.1714	7,594	106.31	4.91

3) The number of CFD modules in our HFD. To better learn bilateral grid of affine coefficients, we extend Context-aware Feature Decomposition (CFD) module to a hierarchical structure. As a comparison, we set different number of CFD modules as the composition of Hierarchical Feature Decomposition (HFD). From Table 6, it can be found that when the number of CFD modules of HFD increases from 1 to 5, the performance of our MSLT improves and then decreases, reaching the best results with three CFDs. This demonstrates that the power of context transformation is enhanced by multiple modules. However, it is unnecessary to use too many CFD modules to extract redundant features. Therefore, we use three CFD modules in our HFD module.

4) How to develop the Hierarchical Feature Decomposition (HFD) module in the bilateral grid network? To answer this question, we apply a variety of networks with comparable parameters with our HFD module to conduct experiments. For ease of presentation, we denote the network consisting of multiple 1×1 convolutional layers and ReLU activation layers as "Conv-1". Similarly, when only using 3×3 convolutions, the network is denoted as "Conv-3". More details are provided in the *Supplementary File*. As shown in Table 7, although "Conv-1" and "Conv-3" also achieve fast speed, our MSLT with HFD achieves better quantitative results in terms of PSNR, SSIM and LPIPS. This shows that our HFD module well estimates the 3D bilateral grid of affine coefficients for exposure correction.

5) How the correction of high-frequency layers influences our MSLT and MSLT+? To this end, for both MSLT and MSLT+, we use partial instead of all corrected high-frequency layers for LP reconstruction. Specifically, our experimental setting could be seen in Table 8. The $\overline{\mathbf{H}}_i$ means that we use the corrected high-frequency layer for LP reconstruction. These high-frequency layers are used for LP reconstruction with $\overline{\mathbf{L}}_4$. Similarly, the $\mathbf{H}_i$ means we directly use the unprocessed high-frequency layer for LP reconstruction. As shown in Table 8, from $\overline{\mathbf{H}}_3$+$\overline{\mathbf{H}}_2$+$\overline{\mathbf{H}}_1$ to $\overline{\mathbf{H}}_3$+$\overline{\mathbf{H}}_2$+$\mathbf{H}_1$, we clearly reduce the FLOPs and inference time (speed) of our MSLT and MSLT+, with little influence on the objective metrics. In our MSLT+, $\mathbf{H}_1$ is generated by learnable convolutions, which can partly compensate for the effect of not processing $\mathbf{H}_1$. This is why our acceleration strategy has little impact on the objective results of MSLT+. All these results show that our acceleration strategy applied on MSLT+ influences little on the objective metrics, but can clearly reduce the computational costs and inference speed.

Table 7. **Results of our MSLT with different variants of HFD module** in the developed Bilateral Grid Network. "Conv-1" (or "Conv-3"): the network consisting of multiple 1×1 (or 3×3) convolutional layers and ReLU activation function. "HFD": our Hierarchical Feature Decomposition module.

Variant	PSNR ↑	SSIM ↑	LPIPS ↓	# Param	FLOPs (M)	Speed (ms)
"Conv-1"	19.31	0.810	0.2103	7,676	64.47	3.58
"Conv-3"	19.10	0.795	0.2167	8,410	65.54	3.70
HFD	21.02	0.835	0.1644	7,594	83.45	4.34

Table 8. **Results of our MSLT and MSLT+ with some high-frequency layers in Laplacian pyramid unprocessed by MSLT/MSLT+.** "H_i": the unprocessed high-frequency layer. "$\overline{\mathrm{H}}_i$": the exposure-corrected high-frequency layer.

Model	Layers	PSNR ↑	SSIM↑	LPIPS ↓	# Param	FLOPs (M)	Speed (ms)
MSLT	$\overline{\mathrm{H}}_3$+$\overline{\mathrm{H}}_2$+$\overline{\mathrm{H}}_1$	21.02	0.835	0.1644	7,594	83.45	4.34
	$\overline{\mathrm{H}}_3$+$\overline{\mathrm{H}}_2$+H_1	20.82	0.831	0.1704	7,594	55.14	3.97
	$\overline{\mathrm{H}}_3$+H_2+H_1	20.60	0.818	0.1841	7,568	48.06	3.72
	H_3+H_2+H_1	20.46	0.820	0.2004	7,448	39.61	3.60
MSLT+	$\overline{\mathrm{H}}_3$+$\overline{\mathrm{H}}_2$+$\overline{\mathrm{H}}_1$	21.18	0.831	0.1589	8,098	170.15	4.07
	$\overline{\mathrm{H}}_3$+$\overline{\mathrm{H}}_2$+H_1	21.12	0.830	0.1648	8,098	141.84	3.67
	$\overline{\mathrm{H}}_3$+H_2+H_1	21.15	0.827	0.1723	8,072	134.77	3.59
	H_3+H_2+H_1	20.57	0.817	0.1806	7,952	126.31	3.36

5. Conclusion

In this paper, we proposed a light-weight and efficient Multi-Scale Linear Transformation (MSLT) network for exposure correction. The proposed MSLT sequentially corrects the exposures of multi-scale low/high-frequency layers decomposed by Laplacian pyramid technique. For the low-frequency layer, we developed a bilateral grid network to learn context-aware affine transformation for pixel-adaptive correction. The high-frequency layers are multiplied in an element-wise manner by comfortable masks learned by channel-wise MLPs. We also accelerated our MSLT by learnable multi-scale decomposition and removing the correction of the largest high-frequency layer. The resulting MSLT++ network has 8,098 parameters, and can process a 4K-resolution image at a 125 FPS speed with only 0.88G FLOPs. Experiments on two benchmarks demonstrated that, our MSLT networks are very efficient and exhibit promising exposure correction performance.

Acknowledgements. Jun Xu is partially sponsored by the National Natural Science Foundation of China (No. 62002176, 62176068, and 62171309), CAAI-Huawei MindSpore Open Fund, and the Open Research Fund (No. B10120210117-OF03) from the Guangdong Provincial Key Laboratory of Big Data Computing, The Chinese University of Hong Kong, Shenzhen.

References

[1] https://github.com/mindspore-ai/mindspore. 6

[2] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*, 2016. 5

[3] Mekides Assefa Abebe. Content fidelity of deep learning methods for clipping and over-exposure correction. In *London Imaging Meeting*, volume 2021, pages 43–48. Society for Imaging Science and Technology, 2021. 1, 2

[4] Mahmoud Afifi, Konstantinos G Derpanis, Bjorn Ommer, and Michael S Brown. Learning multi-scale photo exposure correction. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 9157–9167, 2021. 1, 2, 3, 6, 7

[5] Peter J Burt. Fast filter transform for image processing. *Computer Graphics and Image Processing*, 16(1):20–51, 1981. 5

[6] Peter J Burt and Edward H Adelson. The laplacian pyramid as a compact image code. In *Readings in Computer Vision*, pages 671–679. Elsevier, 1987. 1, 2, 3

[7] Vladimir Bychkovsky, Sylvain Paris, Eric Chan, and Frédo Durand. 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*, pages 97–104. IEEE, 2011. 6

[8] Jianrui Cai, Shuhang Gu, and Lei Zhang. Learning a deep single image contrast enhancer from multi-exposure images. *IEEE Transactions on Image Processing*, 27(4):2049–2062, 2018. 2, 6, 7

[9] Yuhui Cao, Yurui Ren, Thomas H Li, and Ge Li. Over-exposure correction via exposure and scene information disentanglement. In *Proceedings of the Asian Conference on Computer Vision*, 2020. 1, 2

[10] Chen Chen, Qifeng Chen, Jia Xu, and Vladlen Koltun. Learning to see in the dark. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3291–3300, 2018. 1, 2

[11] Jiawen Chen, Andrew Adams, Neal Wadhwa, and Samuel W Hasinoff. Bilateral guided upsampling. *ACM Transactions on Graphics*, 35(6):1–8, 2016. 2, 3, 4

[12] Jiawen Chen, Sylvain Paris, and Frédo Durand. Real-time edge-aware image processing with the bilateral grid. *ACM Transactions on Graphics*, 26(3):103–es, 2007. 2, 3

[13] Ziteng Cui, Kunchang Li, Lin Gu, Shenghan Su, Peng Gao, Zhengkai Jiang, Yu Qiao, and Tatsuya Harada. You only need 90k parameters to adapt light: A light weight transformer for image enhancement and exposure correction. In *British Machine Vision Conference*, 2022. 1, 2, 6, 7

[14] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009. 2

[15] Emily L Denton, Soumith Chintala, Rob Fergus, et al. Deep generative image models using a laplacian pyramid of adversarial networks. *Advances in Neural Information Processing Systems*, 28, 2015. 1, 2

[16] Xiaohan Ding, Chunlong Xia, Xiangyu Zhang, Xiaojie Chu, Jungong Han, and Guiguang Ding. Repmlp: Re-parameterizing convolutions into fully-connected layers for image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. 2

[17] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021. 2

[18] Gabriel Eilertsen, Saghi Hajisharif, Param Hanji, Apostolia Tsirikoglou, Rafał K Mantiuk, and Jonas Unger. How to cheat with metrics in single-image hdr reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3998–4007, 2021. 1, 2

[19] Gabriel Eilertsen, Joel Kronander, Gyorgy Denes, Rafał K Mantiuk, and Jonas Unger. Hdr image reconstruction from a single exposure using deep cnns. *ACM Transactions on Graphics*, 36(6):1–15, 2017. 1, 2

[20] Michaël Gharbi, Jiawen Chen, Jonathan T Barron, Samuel W Hasinoff, and Frédo Durand. Deep bilateral learning for real-time image enhancement. *ACM Transactions on Graphics (TOG)*, 36(4):1–12, 2017. 2

[21] Chunle Guo, Chongyi Li, Jichang Guo, Chen Change Loy, Junhui Hou, Sam Kwong, and Runmin Cong. Zero-reference deep curve estimation for low-light image enhancement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1780–1789, 2020. 6, 7

[22] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7132–7141, 2018. 7, 8

[23] Jie Huang, Yajing Liu, Xueyang Fu, Man Zhou, Yang Wang, Feng Zhao, and Zhiwei Xiong. Exposure normalization and compensation for multiple-exposure correction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6043–6052, June 2022. 6

[24] Jie Huang, Yajing Liu, Feng Zhao, Keyu Yan, Jinghao Zhang, Yukun Huang, Man Zhou, and Zhiwei Xiong. Deep fourier-based exposure correction network with spatial-frequency interaction. In *European Conference on Computer Vision*, pages 163–180. Springer, 2022. 6

[25] Yifan Jiang, Xinyu Gong, Ding Liu, Yu Cheng, Chen Fang, Xiaohui Shen, Jianchao Yang, Pan Zhou, and Zhangyang Wang. Enlightengan: Deep light enhancement without paired supervision. *IEEE Trans. Image Process.*, 30:2340–2349, 2021. 1, 2

[26] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015. 5

[27] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C.J. Burges, L. Bottou, and K.Q. Wein-

berger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012. 6

[28] Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. Deep laplacian pyramid networks for fast and accurate super-resolution. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 624–632, 2017. 1, 2

[29] Xin Li, Xin Jin, Tao Yu, Simeng Sun, Yingxue Pang, Zhizheng Zhang, and Zhibo Chen. Learning omni-frequency region-adaptive representations for real image super-resolution. In *Association for the Advancement of Artificial Intelligence*, volume 35, pages 1975–1983, 2021. 5, 6

[30] Jie Liang, Hui Zeng, and Lei Zhang. High-resolution photo-realistic image translation in real-time: A laplacian pyramid translation network. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9392–9400, 2021. 6, 7

[31] Zhetong Liang, Jun Xu, David Zhang, Zisheng Cao, and Lei Zhang. A hybrid l1-l0 layer decomposition model for tone mapping. In *IEEE Conf. Comput. Vis. Pattern Recog.*, June 2018. 1, 2

[32] Hanxiao Liu, Zihang Dai, David So, and Quoc V Le. Pay attention to mlps. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 9204–9215. Curran Associates, Inc., 2021. 2

[33] Yu-Lun Liu, Wei-Sheng Lai, Yu-Sheng Chen, Yi-Lung Kao, Ming-Hsuan Yang, Yung-Yu Chuang, and Jia-Bin Huang. Single-image hdr reconstruction by learning to reverse the camera pipeline. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1651–1660, 2020. 1, 2

[34] Kede Ma, Zhengfang Duanmu, Hanwei Zhu, Yuming Fang, and Zhou Wang. Deep guided learning for fast multi-exposure image fusion. *IEEE Transactions on Image Processing*, 29:2808–2819, 2019. 2

[35] Long Ma, Tengyu Ma, Risheng Liu, Xin Fan, and Zhongxuan Luo. Toward fast, flexible, and robust low-light image enhancement. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5637–5646, 2022. 6, 7

[36] Andrew L Maas, Awni Y Hannun, Andrew Y Ng, et al. Rectifier nonlinearities improve neural network acoustic models. In *Proceedings of the 30th International Conference on Machine Learning*, volume 30, page 3. Atlanta, Georgia, USA, 2013. 4

[37] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. 5

[38] Wenqi Ren, Sifei Liu, Lin Ma, Qianqian Xu, Xiangyu Xu, Xiaochun Cao, Junping Du, and Ming-Hsuan Yang. Low-light image enhancement via a deep hybrid network. *IEEE Transactions on Image Processing*, 28(9):4364–4375, 2019. 1

[39] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 234–241. Springer, 2015. 1

[40] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning internal representations by error propagation. *Readings in Cognitive Science*, 323(6088):399–421, 1988. 2

[41] Ilya O Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Andreas Steiner, Daniel Keysers, Jakob Uszkoreit, et al. Mlp-mixer: An all-mlp architecture for vision. *Advances in Neural Information Processing Systems*, 34:24261–24272, 2021. 2

[42] Hugo Touvron, Piotr Bojanowski, Mathilde Caron, Matthieu Cord, Alaaeldin El-Nouby, Edouard Grave, Gautier Izacard, Armand Joulin, Gabriel Synnaeve, Jakob Verbeek, et al. Resmlp: Feedforward networks for image classification with data-efficient training. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. 2

[43] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Herve Jegou. Training data-efficient image transformers amp; distillation through attention. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 10347–10357. PMLR, 18–24 Jul 2021. 2

[44] Zhengzhong Tu, Hossein Talebi, Han Zhang, Feng Yang, Peyman Milanfar, Alan Bovik, and Yinxiao Li. Maxim: Multi-axis mlp for image processing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5769–5780, 2022. 2

[45] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Instance normalization: The missing ingredient for fast stylization. *arXiv preprint arXiv:1607.08022*, 2016. 7, 8

[46] Dmitry Ulyanov, Andrea Vedaldi, and Victor S. Lempitsky. Instance normalization: The missing ingredient for fast stylization. *CoRR*, abs/1607.08022, 2016. 4

[47] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. 1

[48] Haoyuan Wang, Ke Xu, and Rynson WH Lau. Local color distributions prior for image enhancement. In *Eur. Conf. Comput. Vis.*, pages 343–359, 2022. 1, 2, 6, 7

[49] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Trans. Image Process.*, 13(4):600–612, 2004. 6

[50] Chen Wei, Wenjing Wang, Wenhan Yang, and Jiaying Liu. Deep retinex decomposition for low-light enhancement. In *British Machine Vision Conference*, 2018. 6

[51] Bin Xu, Yuhua Xu, Xiaoli Yang, Wei Jia, and Yulan Guo. Bilateral grid learning for stereo matching networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12497–12506, 2021. 2, 3

[52] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 586–595, 2018. 6

[53] Zhuoran Zheng, Wenqi Ren, Xiaochun Cao, Xiaobin Hu, Tao Wang, Fenglong Song, and Xiuyi Jia. Ultra-high-definition image dehazing via multi-guided bilateral learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16185–16194, 2021. 2, 3