Learning Multi-Scale Photo Exposure Correction

Mahmoud Afifi[1,2*] Konstantinos G. Derpanis[1] Björn Ommer[3] Michael S. Brown[1]

[1]Samsung AI Centre (SAIC), Toronto, Canada
[2]York University, Canada [3]Heidelberg University, Germany

Abstract

Capturing photographs with wrong exposures remains a major source of errors in camera-based imaging. Exposure problems are categorized as either: (i) overexposed, where the camera exposure was too long, resulting in bright and washed-out image regions, or (ii) underexposed, where the exposure was too short, resulting in dark regions. Both under- and overexposure greatly reduce the contrast and visual appeal of an image. Prior work mainly focuses on underexposed images or general image enhancement. In contrast, our proposed method targets both over- and underexposure errors in photographs. We formulate the exposure correction problem as two main sub-problems: (i) color enhancement and (ii) detail enhancement. Accordingly, we propose a coarse-to-fine deep neural network (DNN) model, trainable in an end-to-end manner, that addresses each subproblem separately. A key aspect of our solution is a new dataset of over 24,000 images exhibiting the broadest range of exposure values to date with a corresponding properly exposed image. Our method achieves results on par with existing state-of-the-art methods on underexposed images and yields significant improvements for images suffering from overexposure errors.

1. Introduction

The exposure used at capture time directly affects the overall brightness of the final rendered photograph. Digital cameras control exposure using three main factors: (i) capture shutter speed, (ii) f-number, which is the ratio of the focal length to the camera aperture diameter, and (iii) the ISO value to control the amplification factor of the received pixel signals. In photography, exposure settings are represented by exposure values (EVs), where each EV refers to different combinations of camera shutter speeds and f-numbers that result in the same exposure effect—also referred to as 'equivalent exposures' in photography.

*This work was done while Mahmoud Afifi was an intern at the SAIC.

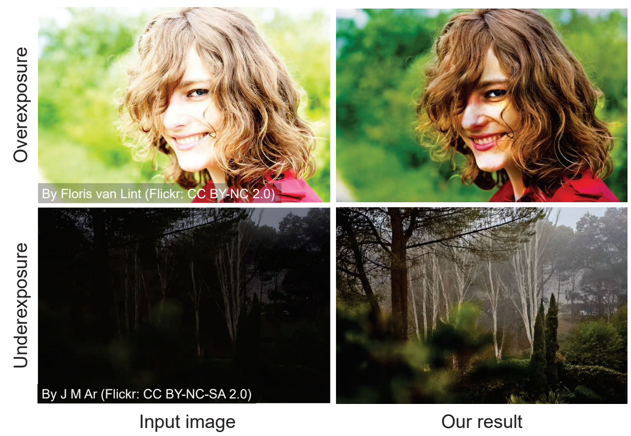


Figure 1: Photographs with over- and underexposure errors and the results of our method using a *single* model for exposure correction. These sample input images are taken from outside our dataset to demonstrate the generalization of our trained model.

Digital cameras can adjust the exposure value of captured images for the purpose of varying the brightness levels. This adjustment can be controlled manually by users or performed automatically in an auto-exposure (AE) mode. When AE is used, cameras adjust the EV to compensate for low/high levels of brightness in the captured scene using through-the-lens (TTL) metering that measures the amount of light received from the scene [49].

Exposure errors can occur due to several factors, such as errors in measurements of TTL metering, hard lighting conditions (e.g., very low lighting and backlighting), dramatic changes in the brightness level of the scene, and errors made by users in the manual mode. Such exposure errors are introduced early in the capture process and are thus hard to correct after rendering the final 8-bit image. This is due to the highly nonlinear operations applied by the camera image signal processor (ISP) afterwards to render the final 8-bit standard RGB (sRGB) image [31].

Fig. 1 shows typical examples of images with exposure errors. In Fig. 1, exposure errors result in either very bright image regions, due to overexposure, or very dark regions, caused by underexposure errors, in the final rendered images. Correcting images with such errors is a challenging task even for well-established image enhancement software packages, see Fig. 9. Although both over- and underexposure errors are common in photography, most prior work is mainly focused on correcting underexposure errors [23, 56, 58, 65, 66] or generic image quality enhancement [11, 18].

Contributions We propose a coarse-to-fine deep learning method for exposure error correction of *both* over- and underexposed sRGB images. Our approach formulates the exposure correction problem as two main sub-problems: (i) color and (ii) detail enhancement. We propose a coarse-to-fine deep neural network (DNN) model, trainable in an end-to-end manner, that begins by correcting the global color information and subsequently refines the image details. In addition to our DNN model, a key contribution to the exposure correction problem is a new dataset containing over 24,000 images[1] rendered from raw-RGB to sRGB with different exposure settings with broader exposure ranges than previous datasets. Each image in our dataset is provided with a corresponding properly exposed reference image. Lastly, we present an extensive set of evaluations and ablations of our proposed method with comparisons to the state of the art. We demonstrate that our method achieves results on par with previous methods dedicated to underexposed images and yields significant improvements on overexposed images. Furthermore, our model generalizes well to images outside our dataset.

2. Related Work

The focus of our paper is on correcting exposure errors in camera-rendered 8-bit sRGB images. We refer the reader to [9, 24, 25, 38] for representative examples for rendering linear raw-RGB images captured with low-light or exposure errors.

Exposure Correction Traditional methods for exposure correction and contrast enhancement rely on image histograms to adjust image intensity values [8, 19, 36, 50, 69]. Alternatively, tone curve adjustment is used to correct images with exposure errors. This process is performed by relying either solely on input image information [63] or trained deep learning models [21, 46, 48, 62]. The majority of prior work adopts the Retinex theory [34] by assuming that improperly exposed images can be formulated as a pixel-wise multiplication of target images, captured with correct exposure settings, by illumination maps. Thus, the goal of these methods is to predict illumination maps to recover the well-exposed target images. Representative

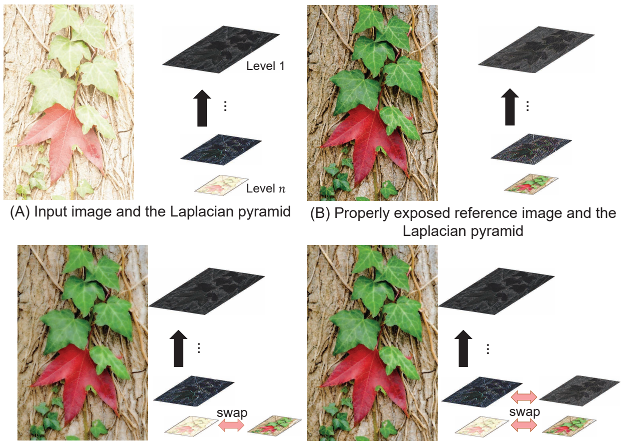
[1]Project page: https://github.com/mahmoudnafifi/Exposure_Correction

Figure 2: Dataset overview. Our dataset contains images with different exposure error types and their corresponding properly exposed reference images. Shown is a t-SNE visualization [42] of all images in our dataset and the low-light (LOL) paired dataset (outlined in red) [58]. Notice that LOL covers a relatively small fraction of the possible exposure levels, as compared to our introduced dataset. Our dataset was rendered from linear raw-RGB images taken from the MIT-Adobe FiveK dataset [6]. Each image was rendered with different relative exposure values (EVs) by an accurate emulation of the camera ISP processes.

Retinex-based methods include [23, 29, 34, 44, 57, 64, 65] and the most recent deep learning ones [56, 58, 66]. Most of these methods, however, are restricted to correcting underexposure errors [23, 56, 58–60, 65, 66, 68]. In contrast to the majority of prior work, our work is the first deep learning method to explicitly correct *both* overexposed and underexposed photographs with a single model.

HDR Restoration and Image Enhancement HDR restoration is the process of reconstructing scene radiance HDR values from one or more low dynamic range (LDR) input images. Prior work either require access to multiple LDR images [16, 30, 43] or use a single LDR input image, which is converted to an HDR image by hallucinating missing information [15, 47]. Ultimately, these reconstructed HDR images are mapped back to LDR for perceptual visualization. This mapping can be directly performed from the input multi-LDR images [7, 13], the reconstructed HDR image [61], or directly from the single input LDR image without the need for radiance HDR reconstruction [11, 18]. There are also methods that focus on general image enhancement that can be applied to enhancing images with poor exposure. In particular, work by [26, 27] was developed primarily to enhance images captured on smartphone cameras by mapping captured images to appear as high-quality images captured by a DSLR. Our work does not seek to reconstruct HDR images or general enhancement, but instead is trained to explicitly address exposure errors.

Paired Dataset Paired datasets are crucial for supervised learning for image enhancement tasks. Existing paired datasets for exposure correction focus only on low-light underexposed images. Representative examples include Wang et al.'s dataset [56] and the low-light (LOL) paired dataset



(A) Input image and the Laplacian pyramid

(B) Properly exposed reference image and the Laplacian pyramid

(C) Reconstructed image using the pyramid in (A) after swapping the last level of the pyramid with the corresponding one in (B)

(D) Reconstructed image using the pyramid in (A) after swapping the last two levels of the pyramid with the corresponding levels in (B)

Figure 3: Motivation behind our coarse-to-fine exposure correction approach. Example of an overexposed image and its corresponding properly exposed image shown in (A) and (B), respectively. The Laplacian pyramid decomposition allows us to enhance the color and detail information sequentially, as shown in (C) and (D), respectively.

[58]. Unlike existing datasets for exposure correction, we introduce a large image dataset rendered with a wide range of exposure errors. Fig. 2 shows a comparison between our dataset and the LOL dataset in terms of the number of images and the variety of exposure errors in each dataset. The LOL dataset covers a relatively small fraction of the possible exposure levels, as compared to our introduced dataset. Our dataset is based on the MIT-Adobe FiveK dataset [6] and is accurately rendered by adjusting the high tonal values provided in camera sensor raw-RGB images to realistically emulate camera exposure errors. An alternative worth noting is to use a large HDR dataset to produce training data—for example, the Google HDR+ dataset [24]. One drawback, however, is that this dataset is a composite of a varying number of smartphone captured raw-RGB images that were first aligned to a composite raw-RGB image. The target ground truth image is based on an HDR-to-LDR algorithm applied to this composite raw-RGB image [18, 24]. We opt instead to use the FiveK dataset as it starts with a single high-quality raw-RGB image and the ground truth result is generated by an expert photographer.

3. Our Dataset

To train our model, we need a large number of training images rendered with realistic over- and underexposure errors and corresponding properly exposed ground truth images. As discussed in Sec. 2, such datasets are currently not publicly available to support exposure correction research. For this reason, our first task is to create a new dataset. Our dataset is rendered from the MIT-Adobe FiveK dataset [6], which has 5,000 raw-RGB images and corresponding sRGB images rendered manually by five expert photographers [6].

For each raw-RGB image, we use the Adobe Camera Raw SDK [1] to emulate different EVs as would be applied by a camera [53]. Adobe Camera Raw accurately emulates the nonlinear camera rendering procedures using metadata embedded in each DNG raw file [2, 53]. We render each raw-RGB image with different digital EVs to mimic real exposure errors. Specifically, we use the relative EVs -1.5, -1, $+0$, $+1$, and $+1.5$ to render images with underexposure errors, a zero gain of the original EV, and overexposure errors, respectively. The zero-gain relative EV is equivalent to the original exposure settings applied onboard the camera during capture time.

As the ground truth images, we use images that were manually retouched by an expert photographer (referred to as Expert C in [6]) as our target correctly exposed images, rather than using our rendered images with $+0$ relative EV. The reason behind this choice is that a significant number of images contain backlighting or partial exposure errors in the original exposure capture settings. The expert adjusted images were performed in ProPhoto RGB color space [6] (rather than raw-RGB), which we converted to a standard 8-bit sRGB color space encoding.

In total, our dataset contains 24,330 8-bit sRGB images with different digital exposure settings. We discarded a small number of images that had misalignment with their corresponding ground truth image. These misalignments are due to different usage of the DNG crop area metadata by Adobe Camera Raw SDK and the expert. Our dataset is divided into three sets: (i) training set of 17,675 images, (ii) validation set of 750 images, and (iii) testing set of 5,905 images. The training, validation, and testing sets do not share any scenes in common. Fig. 2 shows examples of our generated 8-bit sRGB images and the corresponding properly exposed 8-bit sRGB reference images.

4. Our Method

Given an 8-bit sRGB input image, $\mathbf{I}$, rendered with the incorrect exposure setting, our method aims to produce an output image, $\mathbf{Y}$, with fewer exposure errors than those in $\mathbf{I}$. As we simultaneously target both over- and underexposed errors, our input image, $\mathbf{I}$, is expected to contain regions of nearly over- or under-saturated values with corrupted color and detail information. We propose to correct color and detail errors of $\mathbf{I}$ in a sequential manner. Specifically, we process a multi-resolution representation of $\mathbf{I}$, rather than directly dealing with the original form of $\mathbf{I}$. We use the Laplacian pyramid [4] as our multiresolution decomposition, which is derived from the Gaussian pyramid [5] of $\mathbf{I}$.

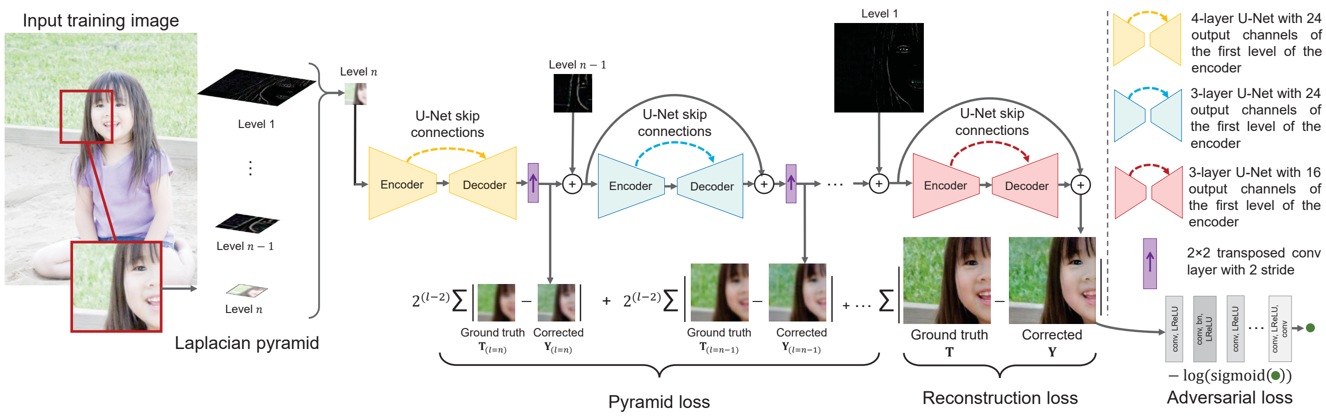


Figure 4: Overview of our image exposure correction architecture. We propose a coarse-to-fine deep network to progressively correct exposure errors in 8-bit sRGB images. Our network first corrects the global color captured at the final level of the Laplacian pyramid and then the subsequent frequency layers.

4.1. Coarse-to-Fine Exposure Correction

Let $\mathbf{X}$ represent the Laplacian pyramid of $\mathbf{I}$ with n levels, such that $\mathbf{X}_{(l)}$ is the l^{th} level of $\mathbf{X}$. The last level of this pyramid (i.e., $\mathbf{X}_{(n)}$) captures low-frequency information of $\mathbf{I}$, while the first level (i.e., $\mathbf{X}_{(1)}$) captures the high-frequency information. Such frequency levels can be categorized into: (i) global color information of $\mathbf{I}$ stored in the low-frequency level and (ii) image coarse-to-fine details stored in the mid- and high-frequency levels. These levels can be later used to reconstruct the full-color image $\mathbf{I}$.

Fig. 3 motivates our coarse-to-fine approach to exposure correction. Figs. 3-(A) and (B) show an example over-exposed image and its corresponding well-exposed target, respectively. As observed, a significant exposure correction can be obtained by using only the low-frequency layer (i.e., the global color information) of the target image in the Laplacian pyramid reconstruction process, as shown in Fig. 3-(C). We can then improve the final image by enhancing the details in a sequential way by correcting each level of the Laplacian pyramid, as shown in Fig. 3-(D). Practically, we do not have access to the properly exposed image in Fig. 3-(B) at the inference stage, and thus our goal is to predict the missing color/detail information of each level in the Laplacian pyramid.

Inspired by this observation and the success of coarse-to-fine architectures for various other computer vision tasks (e.g., [14, 33, 41, 54]), we design a DNN that corrects the global color and detail information of $\mathbf{I}$ in a sequential manner using the Laplacian pyramid decomposition. The remaining parts of this section explain the technical details of our model (Sec. 4.2), including details of the losses (Sec. 4.3), inference phase (Sec. 4.4), and training (Sec. 4.5).

4.2. Coarse-to-Fine Network

Our image exposure correction architecture sequentially processes the n-level Laplacian pyramid, $\mathbf{X}$, of the input image, $\mathbf{I}$, to produce the final corrected image, $\mathbf{Y}$. The proposed model consists of n sub-networks. Each of these sub-networks is a U-Net-like architecture [52] with untied weights. We allocate the network capacity in the form of weights based on how significantly each sub-problem (i.e., global color correction and detail enhancement) contributes to our final result. Fig. 4 provides an overview of our network. As shown, the largest (in terms of weights) sub-network in our architecture is dedicated to processing the global color information in $\mathbf{I}$ (i.e., $\mathbf{X}_{(n)}$). This sub-network (shown in yellow in Fig. 4) processes the low-frequency level $\mathbf{X}_{(n)}$ and produces an upscaled image $\mathbf{Y}_{(n)}$. The upscaling process scales up the output of our sub-network by a factor of two using strided transposed convolution with trainable weights. Next, we add the first mid-frequency level $\mathbf{X}_{(n-1)}$ to $\mathbf{Y}_{(n)}$ to be processed by the second sub-network in our model. This sub-network enhances the corresponding details of the current level and produces a residual layer that is then added to $\mathbf{Y}_{(n)} + \mathbf{X}_{(n-1)}$ to reconstruct image $\mathbf{Y}_{(n-1)}$, which is equivalent to the corresponding Gaussian pyramid level $n-1$. This refinement-upsampling process proceeds until the final output image, $\mathbf{Y}$, is produced. Our network is fully differentiable and thus can be trained in an end-to-end manner. Additional details of our network are provided in the supplementary materials.

4.3. Losses

We train our model end-to-end to minimize the following loss function:

$$\mathcal{L} = \mathcal{L}_{\text{rec}} + \mathcal{L}_{\text{pyr}} + \mathcal{L}_{\text{adv}}, \tag{1}$$

where $\mathcal{L}_{\text{rec}}$ denotes the reconstruction loss, $\mathcal{L}_{\text{pyr}}$ the pyramid loss, and $\mathcal{L}_{\text{adv}}$ the adversarial loss. The individual losses are defined next.

Reconstruction Loss We use the L_1 loss function between the reconstructed and properly exposed reference images. This loss can be expressed as follows:

$$\mathcal{L}_{\text{rec}} = \sum_{p=1}^{3hw} \left| \mathbf{Y}(p) - \mathbf{T}(p) \right|, \tag{2}$$

where h and w denote the height and width of the training image, respectively, and p is the index of each pixel in our corrected image, $\mathbf{Y}$, and the corresponding properly exposed reference image, $\mathbf{T}$, respectively.

Pyramid Loss To guide each sub-network to follow the Laplacian pyramid reconstruction procedure, we introduce dedicated losses at each pyramid level. Let $\mathbf{T}_{(l)}$ denote the l^{th} level of the Gaussian pyramid of our reference image, $\mathbf{T}$, after upsampling by a factor of two. We use a simple interpolation process for the upsampling operation [43]. Our pyramid loss is computed as follows:

$$\mathcal{L}_{\text{pyr}} = \sum_{l=2}^{n} 2^{(l-2)} \sum_{p=1}^{3h_l w_l} \left| \mathbf{Y}_{(l)}(p) - \mathbf{T}_{(l)}(p) \right|, \tag{3}$$

where h_l and w_l are twice the height and width of the l^{th} level in the Laplacian pyramid of the training image, respectively, and p is the index of each pixel in our corrected image at the l^{th} level $\mathbf{Y}_{(l)}$ and the properly exposed reference image at the same level $\mathbf{T}_{(l)}$, respectively. The pyramid loss not only gives a principled interpretation of the task of each sub-network but also results in less visual artifacts compared to training using only the reconstruction loss, as can be seen in Fig. 5. Notice that without the intermediate pyramid losses, the output of each sub-network, shown in Fig. 5 (top), deviates widely from the intermediate Gaussian targets compared to using the pyramid loss at each scale, as shown in Fig. 5 (bottom). We provide supporting justification for this loss with an ablation study in the supplementary materials.

Adversarial Loss To perceptually enhance the reconstruction of the corrected image output in terms of realism

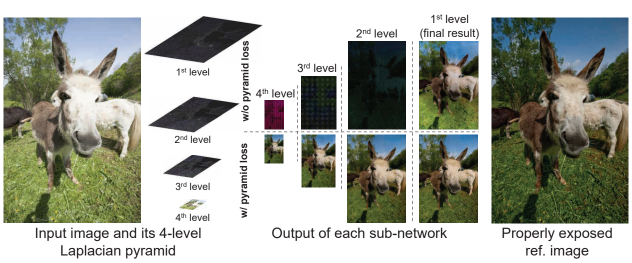


Figure 5: Multiscale losses. Shown are the output of each sub-net trained with and without the pyramid loss (Eq. 3).

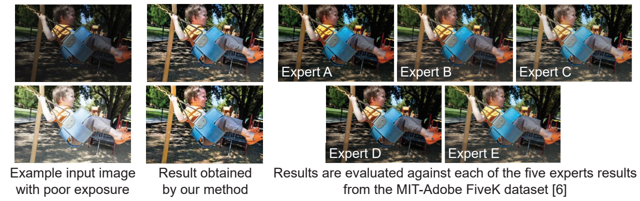


Figure 6: We evaluate the results of input images against all five expert photographers' edits from the FiveK dataset [6].

and appeal, we also consider an adversarial loss as a regularizer. This adversarial loss term can be described by the following equation [20]:

$$\mathcal{L}_{\text{adv}} = -3hwn \log \left(\mathcal{S} \left(\mathcal{D} \left(\mathbf{Y} \right) \right) \right), \tag{4}$$

where $\mathcal{S}$ is the sigmoid function and $\mathcal{D}$ is a discriminator DNN that is trained together with our main network. We provide the details of our discriminator network and visual comparisons between our results using non-adversarial and adversarial training in the supplementary materials.

4.4. Inference Stage

Our network is fully convolutional and can process input images with different resolutions. While our model requires a reasonable memory size (~7M parameters), processing high-resolution images requires a high computational power that may not always be available. Furthermore, processing images with considerably higher resolution (e.g., 16-megapixel) than the range of resolutions used in the training process can affect our model's robustness with large homogeneous image regions. This issue arises because our network was trained on a certain range of effective receptive fields, which is very low compared to the receptive fields required for images with very high resolution. To that end, we use the bilateral guided upsampling method [10] to process high-resolution images. First, we resize the input test image to have a maximum dimension of 512 pixels. Then, we process the downsampled version of the input image using our model, followed by applying the fast upsampling technique [10] with a bilateral grid of $22 \times 22 \times 8$ cells. This process allows us to process a 16-megapixel image in $\sim$4.5 seconds on average. This time includes $\sim$0.5 seconds to run our network on an NVIDIA® GeForce GTX 1080™ GPU and $\sim$4 seconds on an Intel® Xeon® E5-1607 @ 3.10 GHz machine for the guided upsampling process. Note the runtime of the guided upsampling step can be significantly improved with a Halide implementation [51].

4.5. Training Details

In our implementation, we use a Laplacian pyramid with four levels (i.e., $n = 4$) and thus we have four sub-networks in our model—an ablation study evaluating the effect on the

Figure 7: Qualitative results of correcting images with exposure errors. Shown are the input images from our test set, results from the DPED [26], results from the Deep UPE [11], our results, and the corresponding ground truth images.

number of Laplacian levels, including a comparison with a vanilla U-Net architecture, is presented in the supplementary materials. We trained our model on patches randomly extracted from training images with different dimensions. We first train on patches of size 128×128 pixels. Next, we continue training on 256×256 patches, followed by training on 512×512 patches. We use the Adam optimizer [32] to minimize our loss function in Eq. 1. Inspired by previous work [40], we initially train without the adversarial loss term $\mathcal{L}_{\text{adv}}$ to speed up the convergence of our main network. Upon convergence, we then add the adversarial loss term $\mathcal{L}_{\text{adv}}$ and fine-tune our network to enhance our initial results. Additional training details are provided in the supplementary materials.

5. Empirical Evaluation

We compare our method against a broad range of existing methods for exposure correction and image enhancement. We first present quantitative results and comparisons in Sec. 5.1, followed by qualitative comparisons in Sec. 5.2.

5.1. Quantitative Results

To evaluate our method, we use our test set, which consists of 5,905 images rendered with different exposure settings, as described in Sec. 3. Specifically, our test set includes 3,543 well-exposed/overexposed images rendered with $+0$, $+1$, and $+1.5$ relative EVs, and 2,362 underexposed images rendered with -1 and -1.5 relative EVs.

We adopt the following three standard metrics to evaluate the pixel-wise accuracy and the perceptual quality of our results: (i) peak signal-to-noise ratio (PSNR), (ii) structural similarity index measure (SSIM) [67], and (iii) perceptual index (`PI`) [3]. The `PI` is given by:

$$\texttt{PI} = 0.5(10 - \texttt{Ma} + \texttt{NIQE}), \tag{5}$$

where both `Ma` [39] and `NIQE` [45] are *no-reference* image quality metrics.

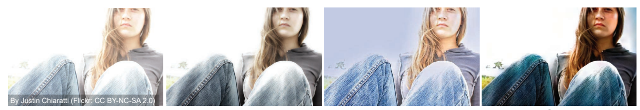


Figure 8: Qualitative comparison with Adobe Photoshop's local adaptation HDR function [12] and DPE [11]. Input images are taken from Flickr.

For the pixel-wise error metrics – namely, PSNR and SSIM – we compare the results not only against the properly exposed rendered images by Expert C but also with *all* five expert photographers in the MIT-Adobe FiveK dataset [6]. Though the expert photographers may render the same image in different ways due to differences in the camera-based rendering settings (e.g., white balance and tone mapping), a common characteristic over all rendered images by the expert photographers is that they all have fairly proper exposure settings [6] (see Fig. 6). For this reason, we evaluate our method against the *five* expert rendered images as they all represent satisfactory exposed reference images.

We also evaluate a variety of previous non-learning and learning-based methods on our test set for comparison: histogram equalization (HE) [19], contrast-limited adaptive histogram equalization (CLAHE) [69], the weighted variational model (WVM) [17], the low-light image enhancement method (LIME) [22, 23], HDR CNN [15], DPED models [26], deep photo enhancer (DPE) models [11], the high-quality exposure correction method (HQEC) [65], RetinexNet [58], deep underexposed photo enhancer (UPE) [56], and the zero-reference deep curve estimation method (Zero-DCE) [21]. To render the reconstructed HDR images generated by the HDR CNN method [15] back into LDR, we tested both the deep reciprocating HDR transformation method (RHT) [61], and Adobe Photoshop's (PS) HDR tool [12].

Table 1 summarizes the quantitative results obtained by each method. As shown in the top portion of the table, our method achieves the best results for overexposed images under all metrics. In the underexposed image correction set-

Table 1: Quantitative evaluation on our introduced test set. **The best results are highlighted with green and bold. The second- and third-best results are highlighted in yellow and red, respectively.** We compare each method with properly exposed reference image sets rendered by five expert photographers [6]. For each method, we present peak signal-to-noise ratio (PSNR), structural similarity index measure (SSIM) [67], and perceptual index (PI) [3]. We denote methods designed for underexposure correction in gray. Non-deep learning methods are marked by $*$. The terms U and S stand for unsupervised and supervised, respectively. Notice that higher PSNR and SSIM values are better, while lower PI values indicate better perceptual quality.

Method	Expert A		Expert B		Expert C		Expert D		Expert E		Avg.		PI
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	
$+0$, $+1$, and $+1.5$ relative EVs (3,543 properly exposed and overexposed images)													
HE [19] $*$	16.140	0.686	16.277	0.672	16.531	0.699	16.643	0.669	17.321	0.691	16.582	0.683	2.351
CLAHE [69] $*$	13.934	0.568	14.689	0.586	14.453	0.584	15.116	0.593	15.850	0.612	14.808	0.589	2.270
WVM [17] $*$	12.355	0.624	13.147	0.656	12.748	0.645	14.059	0.669	15.207	0.690	13.503	0.657	2.342
LIME [22, 23] $*$	09.627	0.549	10.096	0.569	9.875	0.570	10.936	0.597	11.903	0.626	10.487	0.582	2.412
HDR CNN [15] w/ RHT [61]	13.151	0.475	13.637	0.478	13.622	0.497	14.177	0.479	14.625	0.503	13.842	0.486	4.284
HDR CNN [15] w/ PS [12]	14.804	0.651	15.622	0.689	15.348	0.670	16.583	0.685	18.022	0.703	16.076	0.680	2.248
DPED (iPhone) [26]	12.680	0.562	13.422	0.586	13.135	0.581	14.477	0.596	15.702	0.630	13.883	0.591	2.909
DPED (BlackBerry) [26]	15.170	0.621	16.193	0.691	15.781	0.642	17.042	0.677	18.035	0.678	16.444	0.662	2.518
DPED (Sony) [26]	16.398	0.672	17.679	0.707	17.378	0.697	17.997	0.685	18.685	0.700	17.627	0.692	2.740
DPE (HDR) [11]	14.399	0.572	15.219	0.573	15.091	0.593	15.692	0.581	16.640	0.626	15.408	0.589	2.417
DPE (U-FiveK) [11]	14.314	0.615	14.958	0.628	15.075	0.645	15.987	0.647	16.931	0.667	15.453	0.640	2.630
DPE (S-FiveK) [11]	14.786	0.638	15.519	0.649	15.625	0.668	16.586	0.664	17.661	0.684	16.035	0.661	2.621
HQEC [65] $*$	11.775	0.607	12.536	0.631	12.127	0.627	13.424	0.652	14.511	0.675	12.875	0.638	2.387
RetinexNet [58]	10.149	0.570	10.880	0.586	10.471	0.595	11.498	0.613	12.295	0.635	11.059	0.600	2.933
Deep UPE [56]	10.047	0.532	10.462	0.568	10.307	0.557	11.583	0.591	12.639	0.619	11.008	0.573	2.428
Zero-DCE [21]	10.116	0.503	10.767	0.502	10.395	0.514	11.471	0.522	12.354	0.557	11.0206	0.5196	2.774
Our method w/o $\mathcal{L}_{\text{adv}}$	**18.976**	**0.743**	**19.767**	**0.731**	**19.980**	**0.768**	**18.966**	**0.716**	**19.056**	**0.727**	**19.349**	**0.737**	2.189
Our method w/ $\mathcal{L}_{\text{adv}}$	18.874	0.738	19.569	0.718	19.788	0.760	18.823	0.705	18.936	0.719	19.198	0.728	**2.183**
-1 and -1.5 relative EVs (2,362 underexposed images)													
HE [19] $*$	16.158	0.683	16.293	0.669	16.517	0.692	16.632	0.665	17.280	0.684	16.576	0.679	2.486
CLAHE [69] $*$	16.310	0.619	17.140	0.646	16.779	0.621	15.955	0.613	15.568	0.608	16.350	0.621	2.387
WVM [17] $*$	17.686	0.728	19.787	**0.764**	18.670	0.728	18.568	0.729	18.362	0.724	18.615	0.735	2.525
LIME [22, 23] $*$	13.444	0.653	14.426	0.672	13.980	0.663	15.190	0.673	16.177	0.694	14.643	0.671	2.462
HDR CNN [15] w/ RHT [61]	14.547	0.456	14.347	0.427	14.068	0.441	13.025	0.398	11.957	0.379	13.589	0.420	5.072
HDR CNN [15] w/ PS [12]	17.324	0.692	18.992	0.714	18.047	0.696	18.377	0.689	19.593	0.701	18.467	0.698	**2.294**
DPED (iPhone) [26]	18.814	0.680	21.129	0.712	20.064	0.683	**19.711**	0.675	19.574	0.676	19.858	0.685	2.894
DPED (BlackBerry) [26]	19.519	0.673	**22.333**	0.745	20.342	0.669	19.611	0.683	18.489	0.653	**20.059**	0.685	2.633
DPED (Sony) [26]	18.952	0.679	20.072	0.691	18.982	0.662	17.450	0.629	15.857	0.601	18.263	0.652	2.905
DPE (HDR) [11]	17.625	0.675	18.542	0.705	18.127	0.677	16.831	0.665	15.891	0.643	17.403	0.673	2.340
DPE (U-FiveK) [11]	19.130	0.709	19.574	0.674	19.479	0.711	17.924	0.665	16.370	0.625	18.495	0.677	2.571
DPE (S-FiveK) [11]	**20.153**	0.738	20.973	0.697	**20.915**	0.738	19.050	0.688	17.510	0.648	19.720	0.702	2.564
HQEC [65] $*$	15.801	0.692	17.371	0.718	16.587	0.700	17.090	0.705	17.675	0.716	16.905	0.706	2.532
RetinexNet [58]	11.676	0.607	12.711	0.611	12.132	0.621	12.720	0.618	13.233	0.637	12.494	0.619	3.362
Deep UPE [56]	17.832	0.728	19.059	0.754	18.763	0.745	19.641	**0.737**	**20.237**	**0.740**	19.106	0.741	2.371
Zero-DCE [21]	13.935	0.585	15.239	0.593	14.552	0.589	15.202	0.587	15.893	0.614	14.9642	0.5936	3.001
Our method w/o $\mathcal{L}_{\text{adv}}$	19.432	0.750	20.590	0.739	20.542	**0.770**	18.989	0.723	18.874	0.727	19.685	**0.742**	2.344
Our method w/ $\mathcal{L}_{\text{adv}}$	19.475	**0.751**	20.546	0.730	20.518	0.768	18.935	0.715	18.756	0.719	19.646	0.737	2.342
Combined over and underexposed images (5,905 images)													
HE [19] $*$	16.148	0.685	16.283	0.671	16.525	0.696	16.639	0.668	17.305	0.688	16.580	0.682	2.405
CLAHE [69] $*$	14.884	0.589	15.669	0.610	15.383	0.599	15.452	0.601	15.737	0.610	15.425	0.602	2.317
WVM [17] $*$	14.488	0.665	15.803	0.699	15.117	0.678	15.863	0.693	16.469	0.704	15.548	0.688	2.415
LIME [22, 23]	11.154	0.591	11.828	0.610	11.517	0.607	12.638	0.628	13.613	0.653	12.150	0.618	2.432
HDR CNN [15] w/ RHT [61]	13.709	0.467	13.921	0.458	13.800	0.474	13.716	0.446	13.558	0.454	13.741	0.460	4.599
HDR CNN [15] w/ PS [12]	15.812	0.667	16.970	0.699	16.428	0.681	17.301	0.687	18.650	0.702	17.032	0.687	2.267
DPED (iPhone) [26]	15.134	0.609	16.505	0.636	15.907	0.622	16.571	0.627	17.251	0.649	16.274	0.629	2.903
DPED (BlackBerry) [26]	16.910	0.642	18.649	0.713	17.606	0.653	18.070	0.679	18.217	0.668	17.890	0.671	2.564
DPED (Sony) [26]	17.419	0.675	18.636	0.701	18.020	0.683	17.554	0.660	17.778	0.663	17.881	0.676	2.806
DPE (HDR) [11]	15.690	0.614	16.548	0.626	16.305	0.626	16.147	0.615	16.341	0.633	16.206	0.623	2.417
DPE (U-FiveK) [11]	16.240	0.653	16.805	0.646	16.837	0.671	16.762	0.654	16.707	0.650	16.670	0.655	2.606
DPE (S-FiveK) [11]	16.933	0.678	17.701	0.668	17.741	0.696	17.572	0.674	17.601	0.670	17.510	0.677	2.621
HQEC [65] $*$	13.385	0.641	14.470	0.666	13.911	0.656	14.891	0.674	15.777	0.692	14.487	0.666	2.445
RetinexNet [58]	10.759	0.585	11.613	0.596	11.135	0.605	11.987	0.615	12.671	0.636	11.633	0.607	3.105
Deep UPE [56]	13.161	0.610	13.901	0.642	13.689	0.632	14.806	0.649	15.678	0.667	14.247	0.640	2.405
Zero-DCE [21]	11.643	0.536	12.555	0.539	12.058	0.544	12.964	0.548	13.769	0.580	12.5978	0.5494	2.865
Our method w/o $\mathcal{L}_{\text{adv}}$	**19.158**	**0.746**	**20.096**	**0.734**	**20.205**	**0.769**	**18.975**	**0.719**	**18.983**	**0.727**	**19.483**	**0.739**	2.251
Our method w/ $\mathcal{L}_{\text{adv}}$	19.114	0.743	19.960	0.723	20.080	0.763	18.868	0.709	18.864	0.719	19.377	0.731	**2.247**

ting, our results (middle portion of table) are on par with the state-of-the-art methods. Finally, in contrast to most of the existing methods, the results in the bottom portion of the table show that our method can effectively deal with *both* types of exposure errors.

Generalization We further evaluate the generalization ability of our method on the following standard image datasets used by previous low-light image enhancement methods: (i) LIME (10 images) [23], (ii) NPE (75 images) [57], (iii) VV (24 images) [55], and DICM (44 images) [35]. Note that in these experiments, we report results of our model trained on our training set without further tuning or re-training on any of these datasets. Similar to previous methods, we use the NIQE perceptual score [45] for evaluation. Table 2 com-

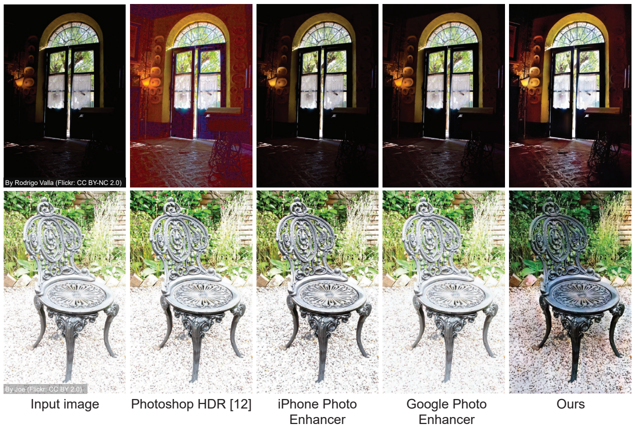


Figure 9: Comparisons with commercial software packages. The input images are taken from Flickr.

pares results by our method and the following methods: LIME [22, 23], WVM [17], RetinexNet (RNet) [58], "kindling the darkness" (KinD) [66], enlighten GAN (EGAN) [28], and deep bright-channel prior (BCP) [37]. As can be seen in Table 2, our method generally achieves perceptually superior results in correcting low-light 8-bit images of other datasets.

5.2. Qualitative Results

We compare our method qualitatively with a variety of previous methods. Note we show results using the model trained with the adversarial loss term, as it produces perceptually superior results (see the perceptual metric results in Tables 1 and 2).

Fig. 7 shows our results on different overexposed and underexposed images. As shown, our results are arguably visually superior to the other methods, even when input images have hard backlight conditions, as shown in the second row in Fig. 7 (right).

Generalization We also ran our model on several images from Flickr that are outside our introduced dataset, as shown in Figs. 1, 8, and 9. As with the images from our introduced dataset, our results on the Flickr images are arguably superior to the compared methods. Additional qualitative results and comparisons are provided in the supplementary materials.

5.3. Limitations

Our method may produce unsatisfactory results in regions that have insufficient semantic information, as shown in Fig. 10. For example, the input image shown in the first row in Fig. 10 is completely saturated and contains almost no details in the region of the man's face. We can see that our network cannot constrain the color inside the face region due to the lack of semantic information. In the supplementary materials, we provide a way to interactively control the output results by scaling each layer of the Laplacian pyramid before feeding them to the network. In that way, one can control the output results to reduce such color bleeding problems. It also can be observed that our method may introduce noise when the input image has extreme dark regions, as shown in the second example in Fig. 10. These challenging conditions prove difficult for other methods as well.

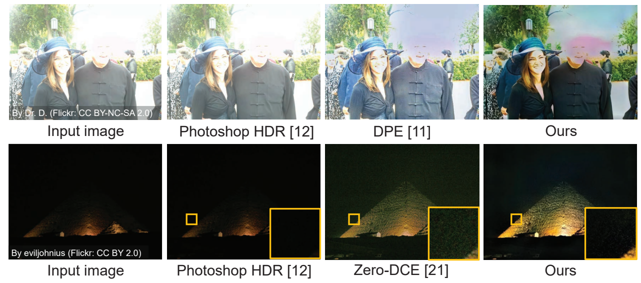


Figure 10: Failure examples of correcting (top) overexposed and (bottom) underexposed images. The input images are taken from Flickr.

Table 2: Perceptual quality evaluation. Summary of NIQE scores [45] on different *low-light* image datasets. In these datasets, there are no ground-truth images provided for full-reference quality metrics (e.g., PSNR). Highlights are in the same format as Table 1.

Method	LIME [23]	NPE [57]	VV [55]	DICM [35]	Avg.
NPE [57] *	3.91	3.95	2.52	3.76	3.54
LIME [23] *	4.16	4.26	2.49	3.85	3.69
WVM [17] *	3.79	3.99	2.85	3.90	3.63
RNet [58]	4.42	4.49	2.60	4.20	3.93
KinD [66]	**3.72**	3.88	-	-	3.80
EGAN [28]	**3.72**	4.11	2.58	-	3.50
DBCP [37]	3.78	**3.18**	-	3.57	3.48
Ours w/o $\mathcal{L}_{\text{adv}}$	3.76	3.20	**2.28**	2.55	2.95
Ours w/ $\mathcal{L}_{\text{adv}}$	3.76	**3.18**	**2.28**	**2.50**	**2.93**

6. Concluding Remarks

We proposed a single coarse-to-fine deep learning model for overexposed and underexposed image correction. We employed the Laplacian pyramid decomposition to process input images in different frequency bands. Our method is designed to sequentially correct each of the Laplacian pyramid levels in a multi-scale manner, starting with the global color in the image and progressively addressing the image details. Our method is enabled by a new dataset of over 24,000 images rendered with the broadest range of exposure errors to date. Each image in our introduced dataset has a reference image properly rendered by a well-trained photographer with well-exposure compensation. Through extensive evaluation, we showed that our method produces compelling results compared to available solutions for correcting images rendered with exposure errors and it generalizes well. We believe that our dataset will help future work on improving exposure correction for photographs.

References

[1] Adobe. Color and camera raw. https://helpx.adobe.com/ca/photoshop-elements/using/color-camera-raw.html. Accessed: 2020-11-12. 3

[2] Mahmoud Afifi, Brian Price, Scott Cohen, and Michael S Brown. When color constancy goes wrong: Correcting improperly white-balanced images. In *CVPR*, 2019. 3

[3] Yochai Blau, Roey Mechrez, Radu Timofte, Tomer Michaeli, and Lihi Zelnik-Manor. The 2018 PIRM challenge on perceptual image super-resolution. In *ECCV Workshops*, 2018. 6, 7

[4] Peter Burt and Edward Adelson. The Laplacian pyramid as a compact image code. *IEEE Transactions on Communications*, 31(4):532–540, 1983. 3

[5] Peter J Burt. Fast filter transform for image processing. *Computer Graphics and Image Processing*, 16(1):20–51, 1981. 3

[6] Vladimir Bychkovsky, Sylvain Paris, Eric Chan, and Frédo Durand. Learning photographic global tonal adjustment with a database of input / output image pairs. In *CVPR*, 2011. 2, 3, 5, 6, 7

[7] 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

[8] Turgay Celik and Tardi Tjahjadi. Contextual and variational contrast enhancement. *IEEE Transactions on Image Processing*, 20(12):3431–3441, 2011. 2

[9] Chen Chen, Qifeng Chen, Jia Xu, and Vladlen Koltun. Learning to see in the dark. In *CVPR*, 2018. 2

[10] Jiawen Chen, Andrew Adams, Neal Wadhwa, and Samuel W Hasinoff. Bilateral guided upsampling. *ACM Transactions on Graphics (TOG)*, 35(6):1–8, 2016. 5

[11] Yu-Sheng Chen, Yu-Ching Wang, Man-Hsin Kao, and Yung-Yu Chuang. Deep photo enhancer: Unpaired learning for image enhancement from photographs with GANs. In *CVPR*, 2018. 2, 6, 7

[12] Lisa DaNae Dayley and Brad Dayley. *Photoshop CS5 Bible*. John Wiley & Sons, 2010. 6, 7

[13] Paul E Debevec and Jitendra Malik. Recovering high dynamic range radiance maps from photographs. In *ACM SIGGRAPH*, 1997. 2

[14] Emily L Denton, Soumith Chintala, Arthur Szlam, and Rob Fergus. Deep generative image models using a Laplacian pyramid of adversarial networks. In *NeurIPS*, 2015. 4

[15] Gabriel Eilertsen, Joel Kronander, Gyorgy Denes, Rafa Mantiuk, and Jonas Unger. HDR image reconstruction from a single exposure using deep CNNs. *ACM Transactions on Graphics (TOG)*, 36(6):178:1–178:15, 2017. 2, 6, 7

[16] Yuki Endo, Yoshihiro Kanamori, and Jun Mitani. Deep reverse tone mapping. *ACM Transactions on Graphics (TOG)*, 36(6):177:1–177:10, 2017. 2

[17] Xueyang Fu, Delu Zeng, Yue Huang, Xiao-Ping Zhang, and Xinghao Ding. A weighted variational model for simultaneous reflectance and illumination estimation. In *CVPR*, 2016. 6, 7, 8

[18] 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):118:1–118:12, 2017. 2, 3

[19] Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing*. Addison-Wesley Longman Publishing Co., Inc., 2001. 2, 6, 7

[20] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NeurIPS*, 2014. 5

[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 *CVPR*, 2020. 2, 6, 7

[22] Xiaojie Guo. LIME: A method for low-light image enhancement. In *ACM MM*, 2016. 6, 7, 8

[23] Xiaojie Guo, Yu Li, and Haibin Ling. LIME: Low-light image enhancement via illumination map estimation. *IEEE Transactions on Image Processing*, 26(2):982–993, 2017. 2, 6, 7, 8

[24] Samuel W Hasinoff, Dillon Sharlet, Ryan Geiss, Andrew Adams, Jonathan T Barron, Florian Kainz, Jiawen Chen, and Marc Levoy. Burst photography for high dynamic range and low-light imaging on mobile cameras. *ACM Transactions on Graphics (TOG)*, 35(6):1–12, 2016. 2, 3

[25] Yuanming Hu, Hao He, Chenxi Xu, Baoyuan Wang, and Stephen Lin. Exposure: A white-box photo post-processing framework. *ACM Transactions on Graphics (TOG)*, 37(2):26:1–26:17, 2018. 2

[26] Andrey Ignatov, Nikolay Kobyshev, Radu Timofte, Kenneth Vanhoey, and Luc Van Gool. DSLR-quality photos on mobile devices with deep convolutional networks. In *ICCV*, 2017. 2, 6, 7

[27] Andrey Ignatov, Nikolay Kobyshev, Radu Timofte, Kenneth Vanhoey, and Luc Van Gool. WESPE: Weakly supervised photo enhancer for digital cameras. In *CVPR Workshops*, 2018. 2

[28] 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. *arXiv preprint arXiv:1906.06972*, 2019. 8

[29] Daniel J Jobson, Ziaur Rahman, and Glenn A Woodell. A multiscale Retinex for bridging the gap between color images and the human observation of scenes. *IEEE Transactions on Image Processing*, 6(7):965–976, 1997. 2

[30] Nima Khademi Kalantari and Ravi Ramamoorthi. Deep high dynamic range imaging of dynamic scenes. *ACM Transactions on Graphics (TOG)*, 36(4):144–1, 2017. 2

[31] Hakki Can Karaimer and Michael S Brown. A software platform for manipulating the camera imaging pipeline. In *ECCV*, 2016. 1

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

[33] Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. Deep Laplacian pyramid networks for fast and accurate super-resolution. In *CVPR*, 2017. 4

[34] Edwin H Land. The Retinex theory of color vision. *Scientific American*, 237(6):108–129, 1977. 2

[35] Chulwoo Lee, Chul Lee, and Chang-Su Kim. Contrast enhancement based on layered difference representation. In *ICIP*, 2012. 7, 8

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

[37] H. Lee, K. Sohn, and D. Min. Unsupervised low-light image enhancement using bright channel prior. *IEEE Signal Processing Letters*, 27:251–255, 2020. 8

[38] Orly Liba, Kiran Murthy, Yun-Ta Tsai, Tim Brooks, Tianfan Xue, Nikhil Karnad, Qiurui He, Jonathan T Barron, Dillon Sharlet, Ryan Geiss, Samuel W Hasinoff, Yael Pritch, and Marc Levoy. Handheld mobile photography in very low light. *ACM Transactions on Graphics (TOG)*, 38(6):1–16, 2019. 2

[39] Chao Ma, Chih-Yuan Yang, Xiaokang Yang, and Ming-Hsuan Yang. Learning a no-reference quality metric for single-image super-resolution. *Computer Vision and Image Understanding*, 158:1–16, 2017. 6

[40] Liqian Ma, Xu Jia, Qianru Sun, Bernt Schiele, Tinne Tuytelaars, and Luc Van Gool. Pose guided person image generation. In *NeurIPS*, 2017. 6

[41] Ruijun Ma, Haifeng Hu, Songlong Xing, and Zhengming Li. Efficient and fast real-world noisy image denoising by combining pyramid neural network and two-pathway unscented Kalman filter. *IEEE Transactions on Image Processing*, 29(1):3927–3940, 2020. 4

[42] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9:2579–2605, 2008. 2

[43] Tom Mertens, Jan Kautz, and Frank Van Reeth. Exposure fusion: A simple and practical alternative to high dynamic range photography. In *Computer Graphics Forum*, 2009. 2, 5

[44] Laurence Meylan and Sabine Susstrunk. High dynamic range image rendering with a Retinex-based adaptive filter. *IEEE Transactions on Image Processing*, 15(9):2820–2830, 2006. 2

[45] Anish Mittal, Rajiv Soundararajan, and Alan C Bovik. Making a "completely blind" image quality analyzer. *IEEE Signal Processing Letters*, 20(3):209–212, 2012. 6, 7, 8

[46] Sean Moran, Pierre Marza, Steven McDonagh, Sarah Parisot, and Gregory Slabaugh. DeepLPF: Deep local parametric filters for image enhancement. In *CVPR*, 2020. 2

[47] Kenta Moriwaki, Ryota Yoshihashi, Rei Kawakami, Shaodi You, and Takeshi Naemura. Hybrid loss for learning single-image-based HDR reconstruction. *arXiv preprint arXiv:1812.07134*, 2018. 2

[48] Jongchan Park, Joon-Young Lee, Donggeun Yoo, and In So Kweon. Distort-and-recover: Color enhancement using deep reinforcement learning. In *CVPR*, 2018. 2

[49] Bryan Peterson. *Understanding exposure: How to shoot great photographs with any camera*. AmPhoto Books, 2016. 1

[50] Stephen M Pizer, E Philip Amburn, John D Austin, Robert Cromartie, Ari Geselowitz, Trey Greer, Bart ter Haar Romeny, John B Zimmerman, and Karel Zuiderveld. Adaptive histogram equalization and its variations. *Computer Vision, Graphics, and Image Processing*, 39(3):355–368, 1987. 2

[51] Jonathan Ragan-Kelley, Connelly Barnes, Andrew Adams, Sylvain Paris, Frédo Durand, and Saman Amarasinghe. Halide: A language and compiler for optimizing parallelism, locality, and recomputation in image processing pipelines. In *ACM PLDI*, 2013. 5

[52] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015. 4

[53] Jeff Schewe and Bruce Fraser. *Real World Camera Raw with Adobe Photoshop CS5*. Pearson Education, 2010. 3

[54] Tamar Rott Shaham, Tali Dekel, and Tomer Michaeli. SinGAN: Learning a generative model from a single natural image. In *ICCV*, 2019. 4

[55] Vassilios Vonikakis. Busting image enhancement and tone-mapping algorithms. `https://sites.google.com/site/vonikakis/datasets`. Accessed: 2020-11-12. 7, 8

[56] Ruixing Wang, Qing Zhang, Chi-Wing Fu, Xiaoyong Shen, Wei-Shi Zheng, and Jiaya Jia. Underexposed photo enhancement using deep illumination estimation. In *CVPR*, 2019. 2, 6, 7

[57] Shuhang Wang, Jin Zheng, Hai-Miao Hu, and Bo Li. Naturalness preserved enhancement algorithm for non-uniform illumination images. *IEEE Transactions on Image Processing*, 22(9):3538–3548, 2013. 2, 7, 8

[58] Chen Wei, Wenjing Wang, Wenhan Yang, and Jiaying Liu. Deep Retinex decomposition for low-light enhancement. In *BMVC*, 2018. 2, 3, 6, 7, 8

[59] Ke Xu, Xin Yang, Baocai Yin, and Rynson WH Lau. Learning to restore low-light images via decomposition-and-enhancement. In *CVPR*, 2020. 2

[60] Wenhan Yang, Shiqi Wang, Yuming Fang, Yue Wang, and Jiaying Liu. From fidelity to perceptual quality: A semi-supervised approach for low-light image enhancement. In *CVPR*, 2020. 2

[61] Xin Yang, Ke Xu, Yibing Song, Qiang Zhang, Xiaopeng Wei, and Rynson WH Lau. Image correction via deep reciprocating HDR transformation. In *CVPR*, 2018. 2, 6, 7

[62] Runsheng Yu, Wenyu Liu, Yasen Zhang, Zhi Qu, Deli Zhao, and Bo Zhang. DeepExposure: Learning to expose photos with asynchronously reinforced adversarial learning. In *NeurIPS*, 2018. 2

[63] Lu Yuan and Jian Sun. Automatic exposure correction of consumer photographs. In *ECCV*, 2012. 2

[64] Qing Zhang, Yongwei Nie, and Wei-Shi Zheng. Dual illumination estimation for robust exposure correction. In *Computer Graphics Forum*, 2019. 2

[65] Qing Zhang, Ganzhao Yuan, Chunxia Xiao, Lei Zhu, and Wei-Shi Zheng. High-quality exposure correction of underexposed photos. In *ACM MM*, 2018. 2, 6, 7

[66] Yonghua Zhang, Jiawan Zhang, and Xiaojie Guo. Kindling the darkness: A practical low-light image enhancer. In *ACM MM*, 2019. 2, 8

[67] Zhou Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004. 6, 7

[68] Minfeng Zhu, Pingbo Pan, Wei Chen, and Yi Yang. EEMEFN: Low-light image enhancement via edge-enhanced multi-exposure fusion network. In *AAAI*, 2020. 2

[69] Karel Zuiderveld. Contrast limited adaptive histogram equalization. In *Graphics Gems IV*, pages 474–485, 1994. 2, 6, 7