Exposure-Aware Dynamic Weighted Learning for Single-Shot HDR Imaging

An Gia Vien[0000−0003−0067−0285] and Chul Lee[0000−0001−9329−7365]

Department of Multimedia Engineering, Dongguk University, Seoul, Korea
viengiaan@mme.dongguk.edu, chullee@dongguk.edu

Abstract. We propose a novel single-shot high dynamic range (HDR) imaging algorithm based on exposure-aware dynamic weighted learning, which reconstructs an HDR image from a spatially varying exposure (SVE) raw image. First, we recover poorly exposed pixels by developing a network that learns local dynamic filters to exploit local neighboring pixels across color channels. Second, we develop another network that combines only valid features in well-exposed regions by learning exposure-aware feature fusion. Third, we synthesize the raw radiance map by adaptively combining the outputs of the two networks that have different characteristics with complementary information. Finally, a full-color HDR image is obtained by interpolating missing color information. Experimental results show that the proposed algorithm significantly outperforms conventional algorithms on various datasets. The source codes and pretrained models are available at https://github.com/viengiaan/EDWL.

Keywords: HDR imaging, SVE image, exposure-aware fusion

1 Introduction

The luminance intensity range of real-world scenes is significantly higher than the range that conventional cameras can capture [33,37]. Therefore, conventional cameras typically acquire low dynamic range (LDR) images, which contain under- and/or over-exposed regions. To overcome the limitations of the conventional imaging systems, high dynamic range (HDR) imaging techniques have been developed to represent, store, and reproduce the full visible luminance range of real-world scenes. Due to its practical importance, various algorithms have been developed to acquire high-quality HDR images [20,26,34,36,48,51,52].

A common approach to HDR imaging is to merge a set of LDR images of a scene captured with different exposure times [7,28]. Whereas this approach works well with static scenes, camera or object motion across LDR images leads to ghosting artifacts in the synthesized HDR images. Although there has been much effort to develop deghosting algorithms for HDR image synthesis [13,16,48,51,52], developing an efficient, robust, and reliable algorithm that can handle complex motions remains a significant challenge. Another approach, called inverse tone mapping, attempts to reconstruct an HDR image from a single LDR image [8,9,17,20,25,36]. Although this approach can prevent ghosting artifacts, it

often fails to reconstruct texture details in large poorly exposed regions due to the lack of underlying information in the regions in a single LDR image.

Another effective approach to HDR imaging that does not result in ghosting artifacts is spatially varying exposure (SVE) [31]. SVE-based HDR imaging, also known as single-shot HDR imaging, algorithms capture a scene with pixel-wise varying exposures in a single image and then computationally synthesize an HDR image, which benefits from the multiple exposures of the single image. Because of this merit, various SVE-based HDR imaging algorithms have recently been developed to improve synthesis performance by recovering poorly exposed pixels by exploiting information from pixels with different exposures [4,5,6,10,38,40,44,50]. However, such algorithms are still susceptible to providing visible artifacts in the synthesized HDR images, since they fail to faithfully recover the missing pixels and texture information in poorly exposed regions.

To alleviate the aforementioned issues, we propose a novel single-shot HDR imaging algorithm that recovers missing information by learning weights to take advantage of the benefits of both neighboring pixels and learned deep features. The proposed algorithm is composed of the dynamic interpolation network (DINet), exposure-aware reconstruction network (ExRNet), and fusion network (FusionNet). DINet recovers poorly exposed pixels by learning local dynamic filters. ExRNet combines only valid features in well-exposed regions. Specifically, we develop the multi-exposure fusion (MEF) block for exposure-aware feature fusion that learns local and channel weights to exploit the complementarity between these features. Finally, FusionNet generates the reconstructed radiance map by adaptively merging the results from DINet and ExRNet. Experimental results demonstrate that the proposed algorithm outperforms the state-of-the-art single-shot HDR imaging algorithms [5,6,40,44,50] on various datasets.

The main contributions of this paper are as follow:

- We propose a learning-based single-shot HDR imaging algorithm that can recover poorly exposed pixels by exploiting both local neighboring pixels across color channels and exposure-aware feature fusion.
- We develop the MEF block, which learns adaptive local and channel weights to effectively fuse valid deep features by exploit the complementary information of the well-exposed regions.
- We experimentally show that the proposed algorithm significantly outperforms state-of-the-art single-shot HDR imaging algorithms on multiple datasets.

2 Related Work

2.1 SVE-based HDR Imaging

An approach to SVE-based HDR imaging is to use spatial light modulators in camera sensors to capture SVE images. Various such techniques have been developed, including learned optical coding [2,27,41], focus pixel sensors [47], and programmable sensors for per-pixel shutter [26,42]. However, these approaches are too complex and expensive for use in practical applications.

An alternative approach to capturing SVE images is to control the per-pixel exposure time or camera gain. Algorithms in this approach can be divided into two categories according to how they synthesize HDR images from captured SVE images. The first category of algorithms attempts to first reconstruct multiple images with different exposures from a single SVE image and then merges them to synthesize an HDR image. Interpolation [10], sparse representation [5], and deep learning [6,40] have been employed to recover the different exposures. However, artifacts in the reconstructed images remain in the synthesized HDR images and degrade their visual quality. The algorithms in the second category directly reconstruct HDR images from SVE images using neighboring pixels with different exposures. Interpolation-based algorithms [4,11] and learning-based algorithms [44,50] have been developed to exploit pixels with different exposures for synthesis. However, they may fail to produce textures in poorly exposed regions due to the spatial inconsistency among neighboring pixels with different exposures.

2.2 Image Inpainting with Partial Convolution

Both single-shot HDR imaging and image inpainting attempt to fill in missing regions of an image with visually plausible content. Recent learning-based approaches [18,32,46,55,56] have shown excellent inpainting performance. However, since these algorithms use convolutional layers, which apply identical filters to the entire image for feature extraction, they may extract invalid information in irregular missing regions. For better handling of those irregular missing regions, the partial convolutional (PConv) layer [19] was developed to ensure the use of only valid information during convolution through a binary mask, which is updated at each layer. In [49], the PConv was generalized by learnable masks for convolution. We also adopt the masks, but the outputs of the learnable mask convolutions are used as local weight maps that represent the relative importance of each value in features for HDR image synthesis.

3 Proposed Algorithm

3.1 Overview

In this work, we assume the SVE pattern in Fig. 1, which consists of row-wise varying exposures with two exposure times: a short exposure time Δt_S and a long exposure time Δt_L, in a single raw Bayer image with the 2×2 RGGB color filter array. This pattern has been commonly employed in single-shot HDR imaging [5,6,44,50]. Specifically, the input SVE image $\mathbf{Z}$ with a resolution of $W \times H$ and bit-depth of 8 is modeled as

$$\mathbf{Z} = \begin{cases} \mathbf{Z}_S, & \text{on } 4n+1 \text{ and } 4n+2\text{-th rows}, \\ \mathbf{Z}_L, & \text{on } 4n+3 \text{ and } 4n+4\text{-th rows}, \end{cases} \tag{1}$$

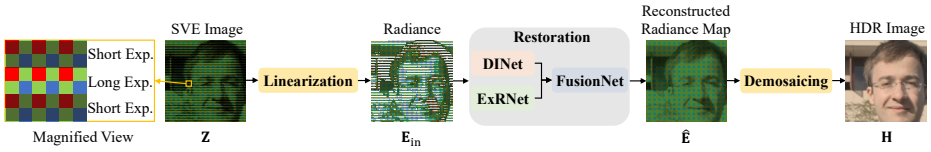


Fig. 1. An overview of the proposed single-shot HDR imaging algorithm. Given a linearized radiance map $\mathbf{E}_{\text{in}}$, DINet, ExRNet, and FusionNet jointly recover missing pixels to output the reconstructed radiance map $\widehat{\mathbf{E}}$. Next, a demosaicing algorithm synthesizes a full-color HDR image $\mathbf{H}$. Missing values in $\mathbf{E}_{\text{in}}$ are illustrated in white.

where $\mathbf{Z}_S$ and $\mathbf{Z}_L$ denote the short- and long-exposure subimages, respectively, and $n = 0, 1, \ldots, \frac{H}{4}$. We then linearize the input $\mathbf{Z}$ into the radiance map $\mathbf{E}_{\text{in}}$ using the camera response function (CRF) [7], which is known *a priori*. As $\mathbf{Z}$ contains poorly exposed pixels, $\mathbf{E}_{\text{in}}$ contains invalid values at the corresponding pixel locations, which are represented by white in Fig. 1.

We synthesize an HDR image by recovering missing information in $\mathbf{E}_{\text{in}}$ through two procedures: restoration and demosaicing. In restoration, missing information in $\mathbf{E}_{\text{in}}$ is recovered using DINet, ExRNet, and FusionNet. Then, given a reconstructed output $\widehat{\mathbf{E}}$, we obtain the full-color HDR image $\mathbf{H}$ using a demosaicing algorithm. We describe each stage subsequently.

3.2 Restoration

Fig. 2 shows the restoration procedure, which is composed of three networks: DINet, ExRNet, and FusionNet. DINet and ExRNet recover missing information by learning dynamic weights in the image and feature domains, respectively. Then, FusionNet fuses the restored results from DINet and ExRNet by exploiting their complementarity to generate the reconstructed radiance map $\widehat{\mathbf{E}}$.

DINet: Interpolation-based single-shot HDR imaging algorithms recover missing information from the neighboring pixels in each color channel using different weighting strategies, such as bicubic interpolation [10], bilateral filtering [4], and polynomial interpolation [11]. However, valid information may not be found in the neighboring pixels in each color channel, especially in large missing regions. To solve this issue, DINet exploits the neighboring pixels across color channels to consider inter-channel correlations for more accurate recovery.

We first rearrange the radiance map $\mathbf{E}_{\text{in}} \in \mathbb{R}^{W \times H}$ into a set of single-color subimages $\{\mathbf{E}^c_{\text{in}}\} \in \mathbb{R}^{\frac{W}{2} \times \frac{H}{2} \times 4}$, where $c \in \{R, G_1, G_2, B\}$ denotes color channels in an SVE image. It is easier to encode long-range dependencies across color channels in the subimages $\{\mathbf{E}^c_{\text{in}}\}$ than in $\mathbf{E}_{\text{in}}$, and the subimages contain structurally similar information. DINet consists of four dynamic filter networks (DFNs) [12] that generate local filters for each color channel. Each DFN[1] takes the four subimages $\{\mathbf{E}^c_{\text{in}}\}$ as input and generates local filter coefficients $\mathbf{k}^c \in \mathbb{R}^{3 \times 3 \times 4}$ dynamically for each color channel c to fuse the 3×3 local neighboring pixels in the

[1] The details of the DFN architecture are provided in the supplemental document.

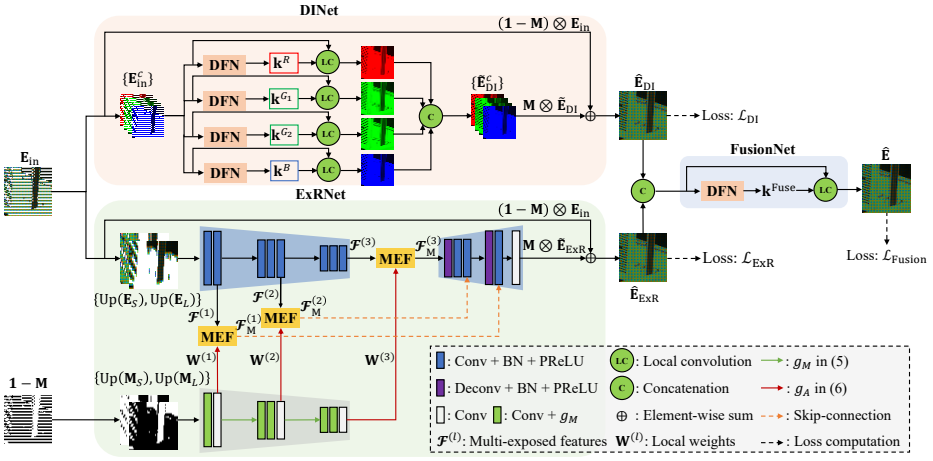


Fig. 2. An overview of the proposed restoration algorithm, consisting of DINet, ExRNet, and FusionNet. DINet learns dynamic local filters for restoration. ExRNet combines only valid features in well-exposed regions for restoration. FusionNet fuses the outputs of DINet and ExRNet to form the reconstructed radiance map $\widehat{\mathbf{E}}$.

four subimages. For each pixel (x, y) of the input $\{\mathbf{E}_{\text{in}}^c\}$, we obtain the filtered output for channel c via local convolution (LC) as

$$\widetilde{E}_{\text{DI}}^c(x,y) = \sum_{c'} \sum_{i=-1}^{1} \sum_{j=-1}^{1} k^c(i,j,c') E_{\text{in}}^{c'}(x+i, y+j), \tag{2}$$

where (i, j) are local coordinates around (x, y), and c' is the color channel index. The filter coefficients are normalized, $\sum_{c'} \sum_i \sum_j k^c(i,j,c') = 1$. Next, we rearrange the filtered outputs $\{\widetilde{\mathbf{E}}_{\text{DI}}^c\}$ into a single image $\widetilde{\mathbf{E}}_{\text{DI}} \in \mathbb{R}^{W \times H}$

Instead of the entire filtered radiance map $\widetilde{\mathbf{E}}_{\text{DI}}$, we use restored radiance values only on the poorly exposed regions and use those in $\mathbf{E}_{\text{in}}$ on well-exposed regions. To this end, we first define a soft mask $\mathbf{M}$ with values in the range of $[0, 1]$ to reveal poorly exposed regions as

$$\mathbf{M} = \min\left(\frac{\max(0, \mathbf{Z} - \tau) + \max(0, 255 - \tau - \mathbf{Z})}{255 - \tau}, 1\right), \tag{3}$$

where τ is a threshold to determine the over-exposure. We then obtain the reconstructed image $\widehat{\mathbf{E}}_{\text{DI}}$ in an exposure-aware manner as

$$\widehat{\mathbf{E}}_{\text{DI}} = \mathbf{M} \otimes \widetilde{\mathbf{E}}_{\text{DI}} + (\mathbf{1} - \mathbf{M}) \otimes \mathbf{E}_{\text{in}}, \tag{4}$$

where $\otimes$ is element-wise multiplication.

ExRNet: As the radiance map $\mathbf{E}_{\text{in}}$ is formed by interlacing two subimages $\{\mathbf{E}_S, \mathbf{E}_L\}$ for long and short exposures, respectively, poorly exposed regions in

$\mathbf{E}_{\text{in}}$ are irregular. Thus, previous approaches that do not take into account the spatial characteristic of poorly exposed regions in $\mathbf{E}_{\text{in}}$ may fail to faithfully restore missing information in an SVE image [6,40,44]. To solve this issue, we develop ExRNet, which combines only valid features in well-exposed regions so that missing pixels in $\mathbf{E}_{\text{in}}$ are restored more reliably and accurately.

Fig. 2 shows the architecture of ExRNet. We employ U-Net [35], which contains an encoder G_E and decoder G_D, as the baseline. The subimages $\{\mathbf{E}_S, \mathbf{E}_L\}$ are first upsampled vertically using linear interpolation $\text{Up}(\cdot)$ to the same resolution as $\mathbf{E}_{\text{in}}$. Then, the set of interpolated images $\{\text{Up}(\mathbf{E}_S), \text{Up}(\mathbf{E}_L)\} \in \mathbb{R}^{W\times H\times 1\times 2}$ is used as input to ExRNet. The encoder G_E extracts multi-exposure features $\boldsymbol{\mathcal{F}}^{(l)} = \left\{\boldsymbol{\mathcal{F}}_S^{(l)}, \boldsymbol{\mathcal{F}}_L^{(l)}\right\} = \left\{G_E^{(l)}(\text{Up}(\mathbf{E}_S)), G_E^{(l)}(\text{Up}(\mathbf{E}_L))\right\}$ at each downsampling level l. In the encoder of ExRNet, the convolution is applied to each subimage and its corresponding feature maps independently. However, note that the feature map $\boldsymbol{\mathcal{F}}^{(l)}$ contains invalid information due to poorly exposed regions in $\mathbf{E}_{\text{in}}$. Thus, we develop the MEF block as shown in Fig. 3, which enables the network to fuse two feature maps with different exposures by exploiting the information of the well-exposed regions in $\mathbf{E}_{\text{in}}$.

Because the two feature maps $\left\{\boldsymbol{\mathcal{F}}_S^{(l)}, \boldsymbol{\mathcal{F}}_L^{(l)}\right\}$ contain irregular missing regions, their fusion using convolution with local weights, which exploits only spatial information, may cause inaccurate restoration with large errors. In this scenario, global contexts across channels may contain meaningful information of a scene. Therefore, to exploit both spatial and global information, the MEF block fuses $\left\{\boldsymbol{\mathcal{F}}_S^{(l)}, \boldsymbol{\mathcal{F}}_L^{(l)}\right\}$ by learning local and channel weights for the spatial fusion and channel fusion, respectively.

First, for spatial fusion, we construct two local weight maps $\mathbf{W}^{(l)} = \{\mathbf{W}_S^{(l)}, \mathbf{W}_L^{(l)}\}$ by considering the information on poorly exposed regions. To this end, we use the encoder G_E with learnable mask convolution [49] to effectively exploit the well-exposed information. Specifically, as shown in Fig. 2, we first extract multi-exposure submasks $\{\mathbf{M}_S, \mathbf{M}_L\}$ from a mask $(\mathbf{1} - \mathbf{M})$ and then vertically upsample them to obtain $\{\text{Up}(\mathbf{M}_S), \text{Up}(\mathbf{M}_L)\}$, which are used as input to the encoder. After each convolution, to constrain each mask value in the range of $[0, 1]$, the mask-updating function g_M is used as an activation function, given by

$$g_M(x) = \big(\text{ReLU}(x)\big)^{\alpha}, \tag{5}$$

where $\alpha > 0$ is a hyper-parameter. At each downsampling level l, we obtain two adaptive local weight maps $\mathbf{W}^{(l)} = \{\mathbf{W}_S^{(l)}, \mathbf{W}_L^{(l)}\}$ by using the learnable attention function g_A [49] as an activation function as

$$g_A(x) = \begin{cases} a \cdot e^{-\gamma_l (x-\beta)^2}, & \text{if } x < \beta \\ 1 + (a-1) \cdot e^{-\gamma_r (x-\beta)^2}, & \text{otherwise,} \end{cases} \tag{6}$$

where a, β, γ_l, and γ_r are the learnable parameters. We then obtain the fused feature map $\boldsymbol{\mathcal{F}}_{\text{Sp}}^{(l)}$ by spatial fusion in an exposure-aware manner as

$$\boldsymbol{\mathcal{F}}_{\text{Sp}}^{(l)} = \frac{\boldsymbol{\mathcal{F}}_S^{(l)} \otimes \mathbf{W}_S^{(l)} + \boldsymbol{\mathcal{F}}_L^{(l)} \otimes \mathbf{W}_L^{(l)}}{\mathbf{W}_S^{(l)} + \mathbf{W}_L^{(l)}}, \tag{7}$$

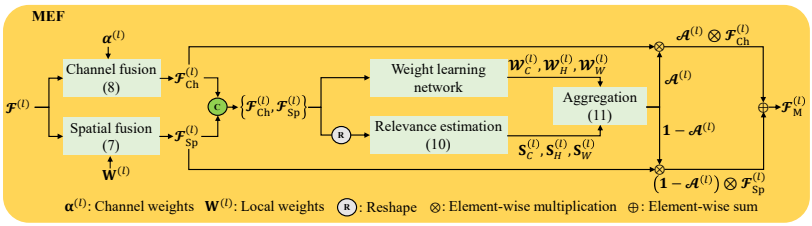


Fig. 3. Architecture of the MEF block.

where the division is component-wise.

Next, assuming that each channel of features represents different visual content, at each downsampling level l, the MEF block learns two channel weight maps $\boldsymbol{\alpha}^{(l)} = \{\boldsymbol{\alpha}_S^{(l)}, \boldsymbol{\alpha}_L^{(l)}\} \in \mathbb{R}^{C^{(l)}\times 1\times 1}$, where $C^{(l)}$ is the number of channels. Then, the fused feature map $\boldsymbol{\mathcal{F}}_{\text{Ch}}^{(l)}$ by channel fusion is obtained by

$$\boldsymbol{\mathcal{F}}_{\text{Ch}}^{(l)} = \frac{\boldsymbol{\mathcal{F}}_S^{(l)} \odot \boldsymbol{\alpha}_S^{(l)} + \boldsymbol{\mathcal{F}}_L^{(l)} \odot \boldsymbol{\alpha}_L^{(l)}}{\boldsymbol{\alpha}_S^{(l)} + \boldsymbol{\alpha}_L^{(l)}}, \tag{8}$$

where $\odot$ denotes channel-wise multiplication, and the division is channel-wise.

Although the two feature maps $\{\boldsymbol{\mathcal{F}}_{\text{Ch}}^{(l)}, \boldsymbol{\mathcal{F}}_{\text{Sp}}^{(l)}\}$ are obtained by fusing multi-exposure features by learning local and channel weights in (7) and (8), respectively, the feature representations in $\boldsymbol{\mathcal{F}}_{\text{Ch}}^{(l)}$ and $\boldsymbol{\mathcal{F}}_{\text{Sp}}^{(l)}$ may become inconsistent due to independent weight learning. Thus, a straightforward fusion of those feature maps using local convolutions may fail to convey essential information in the feature maps, because local convolutions can capture only local information from a small region. To address the limitation of local convolutions by capturing the long-range dependencies in an entire image, transformers [43] or a non-local module [45] have been employed [53,54] that can exploit the correlations between features. In this work, we develop an element-wise weighting scheme that can consider the relationship between $\boldsymbol{\mathcal{F}}_{\text{Ch}}^{(l)}$ and $\boldsymbol{\mathcal{F}}_{\text{Sp}}^{(l)}$ in both spatial and channel domains inspired by transformers and non-local module, as shown in Fig. 3. Specifically, the output of the MEF block $\boldsymbol{\mathcal{F}}_{\text{M}}^{(l)}$ is obtained as the element-wise weighted sum of the two features, given by

$$\boldsymbol{\mathcal{F}}_{\text{M}}^{(l)} = \boldsymbol{\mathcal{A}}^{(l)} \otimes \boldsymbol{\mathcal{F}}_{\text{Ch}} + (\mathbf{1} - \boldsymbol{\mathcal{A}}^{(l)}) \otimes \boldsymbol{\mathcal{F}}_{\text{Sp}}, \tag{9}$$

where $\boldsymbol{\mathcal{A}}^{(l)}$ is the learnable weight map.

In (9), the weight map $\boldsymbol{\mathcal{A}}^{(l)}$ is obtained so that the merged feature map carries the complementary information from the two feature maps. To this end, we employ a neural network[2] to learn three weight maps $\boldsymbol{\mathcal{W}}_W^{(l)}$, $\boldsymbol{\mathcal{W}}_H^{(l)}$, and $\boldsymbol{\mathcal{W}}_C^{(l)}$ of the size $C^{(l)} \times W^{(l)} \times H^{(l)}$ for each dimension of the feature maps. Next, the similarities (relevance) between the two fused features across spatial and channel

[2] The details of the network are provided in the supplemental document.

domains are computed for relevance embedding. Specifically, we first reshape $\boldsymbol{\mathcal{F}}_{\text{Ch}}^{(l)}$ and $\boldsymbol{\mathcal{F}}_{\text{Sp}}^{(l)}$ into matrices in $\mathbb{R}^{C^{(l)}\times W^{(l)}H^{(l)}}$, $\mathbb{R}^{H^{(l)}\times W^{(l)}C^{(l)}}$, and $\mathbb{R}^{W^{(l)}\times H^{(l)}C^{(l)}}$. Then, for each pair of reshaped feature maps, we compute the relevance map between the two feature maps; we thereby obtain three relevance maps: channel relevance map $\mathbf{S}_C^{(l)}$ to measure channel-wise similarities and height and width relevance maps $\mathbf{S}_W^{(l)}$ and $\mathbf{S}_H^{(l)}$ to measure width- and height-wise similarities. For example, let $\mathbf{R}_{\text{Ch}}^C$ and $\mathbf{R}_{\text{Sp}}^C$ denote the reshaped feature maps in the channel domain, then the channel relevance map $\mathbf{S}_C^{(l)}$ is obtained by

$$\mathbf{S}_C^{(l)} = \left(\frac{\mathbf{R}_{\text{Ch}}^C}{\|\mathbf{R}_{\text{Ch}}^C\|}\right)\left(\frac{\mathbf{R}_{\text{Sp}}^C}{\|\mathbf{R}_{\text{Sp}}^C\|}\right)^T. \tag{10}$$

Then, the weight map $\boldsymbol{\mathcal{A}}^{(l)}$ is obtain by aggregating the weight maps with the relevance maps as

$$\boldsymbol{\mathcal{A}}^{(l)}(i,j,c) = \frac{s_W\cdot\boldsymbol{\mathcal{W}}_W^{(l)}(i,j,c) + s_H\cdot\boldsymbol{\mathcal{W}}_H^{(l)}(i,j,c) + s_C\cdot\boldsymbol{\mathcal{W}}_C^{(l)}(i,j,c)}{\boldsymbol{\mathcal{W}}_W^{(l)}(i,j,c) + \boldsymbol{\mathcal{W}}_H^{(l)}(i,j,c) + \boldsymbol{\mathcal{W}}_C^{(l)}(i,j,c)}, \tag{11}$$

where $s_W = \sum_{k=1}^{W^{(l)}}\mathbf{S}_W^{(l)}(i,k)$, $s_H = \sum_{k=1}^{H^{(l)}}\mathbf{S}_H^{(l)}(j,k)$, and $s_C = \sum_{k=1}^{C^{(l)}}\mathbf{S}_C^{(l)}(c,k)$, and (i,j,c) are the indices of $W^{(l)}$, $H^{(l)}$, and $C^{(l)}$, respectively. For each channel c, the weights are normalized, $\sum_i\sum_j\boldsymbol{\mathcal{A}}^{(l)}(i,j,c) = 1$.

Finally, similarly to DINet, we reconstruct the output image $\widehat{\mathbf{E}}_{\text{ExR}}$ of ExRNet in an exposure-aware manner as

$$\widehat{\mathbf{E}}_{\text{ExR}} = \mathbf{M}\otimes\widetilde{\mathbf{E}}_{\text{ExR}} + (\mathbf{1}-\mathbf{M})\otimes\mathbf{E}_{\text{in}}. \tag{12}$$

FusionNet: In Fig. 2, FusionNet synthesizes an output image $\widehat{\mathbf{E}}$ by combining two reconstructed images, $\widehat{\mathbf{E}}_{\text{DI}}$ and $\widehat{\mathbf{E}}_{\text{ExR}}$, from DINet and ExRNet, respectively. Since the two images are reconstructed in the image and feature domains, respectively, they have different characteristics with complementary information, as will be discussed in Section 4.4. We adopt DFN in DINet as FusionNet, which learns local filters $\mathbf{k}^{\text{Fuse}} \in \mathbb{R}^{3\times3\times2}$ for combining the two images, $\widehat{\mathbf{E}}_{\text{DI}}$ and $\widehat{\mathbf{E}}_{\text{ExR}}$, and then obtains the filtered image $\widehat{\mathbf{E}}$ via the LC in (2).

3.3 Demosaicing

As mentioned above, the reconstructed image $\widehat{\mathbf{E}}$ is the Bayer pattern image, as shown in Fig. 1. It therefore requires the interpolation of missing color information to obtain a full-color HDR image $\mathbf{H}$. In this work, we employ the existing demosaicing algorithms [1,21,39,57]. The choice of the demosaicing algorithm affects the synthesis performance, as will be discussed in Section 4.4.

3.4 Loss Functions

To train DINet, ExRNet, and FusionNet, we define the DI loss $\mathcal{L}_{\text{DI}}$, ExR loss $\mathcal{L}_{\text{ExR}}$, and fusion loss $\mathcal{L}_{\text{Fusion}}$, respectively, as will be described subsequently.

DI loss: To train DINet, we define the DI loss $\mathcal{L}_{\text{DI}}$ as the weighted sum of the reconstruction loss $\mathcal{L}_r$ and the multi-scale contrast loss $\mathcal{L}_{\text{MC}}$ between a ground-truth radiance map $\mathbf{E}_{\text{gt}}$ and reconstructed radiance map $\widehat{\mathbf{E}}_{\text{DI}}$ as

$$\mathcal{L}_{\text{DI}} = \mathcal{L}_r(\mathbf{E}_{\text{gt}}, \widehat{\mathbf{E}}_{\text{DI}}) + \lambda_{\text{MC}}\mathcal{L}_{\text{MC}}(\mathbf{E}_{\text{gt}}, \widehat{\mathbf{E}}_{\text{DI}}), \tag{13}$$

where λ_{MC} is a hyper-parameter to balance the two losses. To define the losses, we compress the range of radiance values using the μ-law function $\mathcal{T}$ [13] as

$$\mathcal{T}(x) = \frac{\log(1+\mu x)}{\log(1+\mu)}, \tag{14}$$

where the parameter μ controls the amount of compression. We employ the ℓ_1-norm as the reconstruction loss $\mathcal{L}_r$ in poorly-exposed regions as

$$\mathcal{L}_r = \left\| \mathbf{M}_{\text{h}} \otimes \left(\mathcal{T}(\mathbf{E}_{\text{gt}}) - \mathcal{T}(\widehat{\mathbf{E}}_{\text{DI}}) \right) \right\|_1, \tag{15}$$

where $\mathbf{M}_{\text{h}}$ denotes a hard binary mask. A mask value of 1 indicates that the corresponding pixel in $\mathbf{Z}$ is poorly exposed, *i.e.*, $Z(x,y) < \tau$ or $Z(x,y) > 255 - \tau$ with the threshold τ. The multi-scale contrast loss [58] is defined as

$$\mathcal{L}_{\text{MC}} = 1 - \prod_{j=1}^{M} cs_j\big(\mathcal{T}(\mathbf{E}_{\text{gt}}), \mathcal{T}(\widehat{\mathbf{E}}_{\text{DI}})\big), \tag{16}$$

where M is the number of scales, and cs_j denotes the contrast and structure term at the j-th scale of SSIM.

ExR loss: We define the ExR loss $\mathcal{L}_{\text{ExR}}$ to train ExRNet as a weighted sum of the DI loss $\mathcal{L}_{\text{DI}}$ and the adversarial loss $\mathcal{L}_{\text{Adv}}$ between $\mathbf{E}_{\text{gt}}$ and a reconstructed map $\widehat{\mathbf{E}}_{\text{ExR}}$ as

$$\mathcal{L}_{\text{ExR}} = \mathcal{L}_{\text{DI}}(\mathbf{E}_{\text{gt}}, \widehat{\mathbf{E}}_{\text{ExR}}) + \lambda_{\text{Adv}}\mathcal{L}_{\text{Adv}}(\mathbf{E}_{\text{gt}}, \widehat{\mathbf{E}}_{\text{ExR}}), \tag{17}$$

where λ_{Adv} is a hyper-parameter that controls the relative impacts of the two losses. The adversarial loss $\mathcal{L}_{\text{Adv}}$ penalizes the semantic difference estimated by the discriminator network D,[3] which is defined as

$$\mathcal{L}_{\text{Adv}} = -\log D\big(\mathbf{M}_{\text{h}} \otimes \mathcal{T}(\widehat{\mathbf{E}}_{\text{ExR}})\big). \tag{18}$$

Fusion loss: To train FusionNet, we define the fusion loss $\mathcal{L}_{\text{Fusion}}$ between $\mathbf{E}_{\text{gt}}$ and $\widehat{\mathbf{E}}$ similarly to the DI loss $\mathcal{L}_{\text{DI}}$ in (13) but without $\mathbf{M}_{\text{h}}$ in $\mathcal{L}_r$.

4 Experiments

4.1 Datasets

Since there is no publicly available SVE image dataset with ground-truths, we evaluate the performance of the proposed algorithm on synthetic images

[3] The details of the network architecture is provided in the supplemental document.

from various datasets. Specifically, we define a set of exposure values as EV $= \{-1, +1\}$ for short and long exposures and then generate Bayer pattern images. The images in the datasets are either calibrated in units of cd/m^2 or non-calibrated. We multiply non-calibrated HDR images by a single constant to approximate luminance values, as done in [22,24,44].

Fairchild's dataset[4]**:** It contains 105 HDR images of the resolution 2848×4288; 36 images are calibrated and 69 images are non-calibrated.

Kalantari's dataset [13]: Its test set contains 12 non-calibrated HDR images of the resolution 1500×1000.

HDM-HDR[5]**:** It contains 10 non-calibrated videos of the resolution 1856×1024. We randomly selected 12 HDR frames for the test, and they are provided in the supplemental document.

HDR-Eye[6]**:** It contains 46 HDR images of the resolution 1920×1056, of which 16 are calibrated and 30 are non-calibrated.

HDRv [15]: It contains four calibrated HDR videos of HD (1280×720) resolution. For each video, we chose four different frames; thus, there are 16 calibrated HDR images in total.

4.2 Training

We first train DINet and ExRNet separately and then, after fixing them, train FusionNet. Next, we train the demosaicing networks with optimized DINet, ExRNet, and FusionNet in an end-to-end manner.

DINet, ExRNet, and FusionNet: We use the Adam optimizer [14] with $\beta_1 = 0.9$ and $\beta_2 = 0.999$ and a learning rate of 10^{-4} for 150 epochs. The threshold τ in (3) and (15) is set to 15, and the hyper-parameters α in (5), λ_{MC} in (13), λ_{Adv} in (17), and μ in (14) are fixed to 0.8, 0.75, 10^{-3}, and 5000, respectively.

Demosaicing: We retrain conventional demosaicing networks [57,39] using the robust loss in [44] with the same settings as those in DINet and ExRNet training.

Training dataset: We use only 36 calibrated images from the Fairchild's dataset in Section 4.1 for training. We augment the dataset by rotating and flipping images, and then we divided all training HDR images into non-overlapping patches with the size of 32×32.

4.3 Performance Comparison

We evaluate the synthesis performance of the proposed algorithm with those of conventional algorithms: Choi *et al.*'s [5], Suda *et al.*'s [40], Çoğalan and Akyüz's [6], Vien and Lee's [44], and Xu *et al.*'s [50]. We retrained the learning-based algorithms [6,40,44,50] with the parameter settings recommended by the

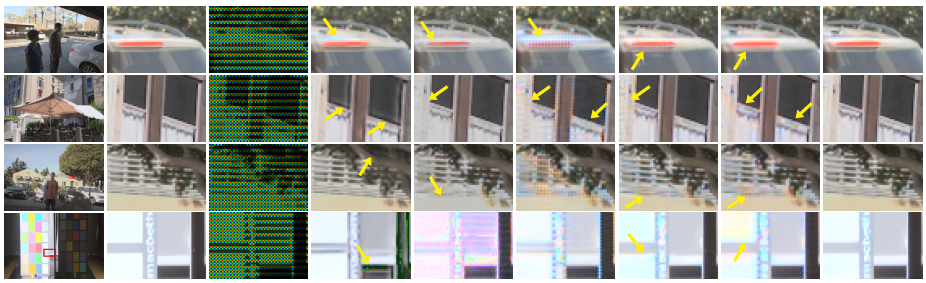
[4] http://markfairchild.org/HDRPS/HDRthumbs.html
[5] https://www.hdm-stuttgart.de/vmlab/hdm-hdr-2014
[6] https://mmspg.epfl.ch/hdr-eye

Table 1. Quantitative comparison of the proposed algorithm with the conventional algorithms on the test sets using six quality metrics. For each metric, the best result is **boldfaced**, while the second best is underlined.

	pu-MSSSIM	pu-PSNR	log-PSNR	HDR-VDP		HDR-VQM
				Q	P	
Kalantari's dataset						
Choi *et al.* [5]	0.9750	36.17	35.47	69.35	0.4559	0.9266
Suda *et al.* [40]	0.9833	37.19	36.27	71.48	0.5103	0.8826
Çoğalan and Akyüz [6]	0.9870	38.96	37.67	70.25	0.7694	0.9296
Vien and Lee [44]	0.9964	45.10	42.22	73.72	0.3930	0.9696
Xu *et al.* [50]	0.9957	44.62	42.01	73.00	0.5593	0.9700
Proposed	**0.9969**	**46.17**	**43.04**	**74.03**	**0.3889**	**0.9718**
HDM-HDR						
Choi *et al.* [5]	0.9540	33.71	27.20	66.13	0.4523	0.5677
Suda *et al.* [40]	0.9594	32.41	25.62	66.33	0.5989	0.4418
Çoğalan and Akyüz [6]	0.9399	35.61	27.15	67.07	0.6711	0.6246
Vien and Lee [44]	**0.9769**	38.58	29.52	68.34	0.4580	0.6520
Xu *et al.* [50]	0.9758	38.68	29.89	68.05	0.4878	0.6533
Proposed	0.9759	**39.44**	**30.87**	**68.70**	**0.4340**	**0.6948**
HDR-Eye						
Choi *et al.* [5]	0.9522	34.49	33.56	67.95	0.5334	0.8633
Suda *et al.* [40]	0.9823	39.78	37.23	71.80	0.5481	0.8452
Çoğalan and Akyüz [6]	0.9728	37.43	34.87	70.15	0.7952	0.8894
Vien and Lee [44]	0.9937	42.06	39.32	72.68	0.4902	0.9209
Xu *et al.* [50]	0.9933	41.28	38.28	72.42	0.6039	0.9149
Proposed	**0.9950**	**43.35**	**40.16**	**73.02**	**0.4053**	**0.9354**
HDRv						
Choi *et al.* [5]	0.9886	45.30	44.16	71.71	0.1809	0.9782
Suda *et al.* [40]	0.9954	47.87	44.36	74.27	0.3088	0.9731
Çoğalan and Akyüz [6]	0.9935	45.20	43.88	69.79	0.4352	0.9816
Vien and Lee [44]	0.9979	50.75	48.00	74.96	0.1290	0.9841
Xu *et al.* [50]	0.9976	50.99	47.49	74.13	0.3366	0.9835
Proposed	**0.9983**	**54.58**	**49.83**	**75.74**	**0.0978**	**0.9854**

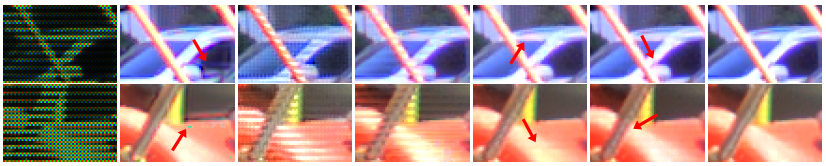
authors using the training dataset in Section 4.2. We use six quality metrics: pu-MSSSIM, pu-PSNR, log-PSNR [3], Q and P scores of HDR-VDP [23,29], and HDR-VQM [30]. The pu-MSSSIM and pu-/log-PSNR metrics are extensions of the MS-SSIM and PSNR, respectively, that consider human perception.

Table 1 compares the synthesis performances quantitatively on various datasets. First, the proposed algorithm outperforms the conventional algorithms in terms of pu-MSSSIM and pu-/log-PSNR in all cases by large margins, except for pu-MSSSIM on the HDM-HDR dataset, where the proposed algorithm achieves the second-best results. For example, the proposed algorithm achieves a 3.59 dB higher pu-PSNR and a 1.83 dB higher log-PSNR scores on HDRv dataset, and a 0.0013 higher pu-MSSSIM score on the HDR-Eye dataset in comparison with the second best, Vien and Lee's. Second, the proposed algorithm also provides the best results for perceptual quality metrics HDR-VDP and HDR-VQM, with no exception. In particular, on the HDRv dataset, the proposed algorithm yields a 0.79 higher HDR-VDP Q score than the second best, Vien and Lee's. These results indicate that the proposed algorithm synthesizes high-quality HDR im-



(a) (b) (c) (d) (e) (f) (g) (h) (i)

Fig. 4. Qualitative comparison of synthesized HDR images. (a) Ground-truths and the magnified parts for the red rectangles in (b) ground-truths, (c) synthetic SVE images, and synthesized images obtained by (d) Choi *et al.*'s [5], (e) Suda *et al.*'s [40], (f) Çoğalan and Akyüz's [6], (g) Vien and Lee's [44], (h) Xu *et al.*'s [50], and (i) the proposed algorithm.



(a) (b) (c) (d) (e) (f) (g)

Fig. 5. Synthesis results of the captured images. The magnified parts in (a) SVE images, and synthesized images obtained by (b) Choi *et al.*'s [5], (c) Suda *et al.*'s [40], (d) Çoğalan and Akyüz's [6], (e) Vien and Lee's [44], (f) Xu *et al.*'s [50], and (g) the proposed algorithm.

ages by recovering missing pixels accurately and in consideration of semantic information.

Fig. 4 qualitatively compares the synthesis results obtained by each algorithm. The conventional algorithms in Figs. 4(d)–(h) fail to synthesize textures and, thus, produce results with blurring, jaggy, and false-color artifacts in poorly exposed regions. In contrast, the proposed algorithm in Fig. 4(i) synthesizes high-quality HDR images without visible artifacts by restoring textures faithfully. For example, the conventional algorithms yield strong visible artifacts around the edges of the red light bar in the first row, which are effectively suppressed by the proposed algorithm. More qualitative comparisons are provided in the supplemental document.

Finally, we compare the synthesis results for the captured image dataset, provided in [44]. The synthesized results in Fig. 5 exhibit similar tendencies to those in Fig. 4. These results indicate that the proposed algorithm can effectively process real SVE images with real noise captured by real-world cameras, providing a superior generalization ability.

Table 2. Impacts of the multi-domain learning in the restoration algorithm on the synthesis performance.

	pu-MSSSIM	pu-PSNR	log-PSNR
DINet	0.9969	44.84	45.11
ExRNet	0.9965	46.96	47.01
FusionNet	**0.9973**	**47.72**	**47.75**

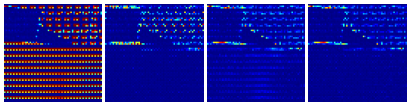


Fig. 6. Comparison of the error maps for different networks.

Table 3. Impacts of the mask **M** in DINet and ExRNet on the restoration performance.

	M	pu-MSSSIM	pu-PSNR	log-PSNR
DINet		0.9960	38.18	38.53
	✓	**0.9969**	**44.84**	**45.11**
ExRNet		0.9893	42.86	42.96
	✓	**0.9965**	**46.96**	**47.01**

4.4 Model Analysis

We analyze the contributions of key components in the proposed algorithm: multi-domain learning, exposure-aware reconstruction, and the MEF block. We also analyze the effects of the demosaicing algorithms on the synthesis performance. All experiments are performed using the Kalantari's dataset.

Multi-domain learning: To analyze the effects of DINet, ExRNet, and FusionNet in Fig. 2 on the synthesis performance, we train the proposed network with different settings. Table 2 compares the average scores. ExRNet yields significantly higher pu-PSNR and log-PSNR scores but a slightly worse pu-MSSSIM score than DINet. This indicates that, while ExRNet faithfully recovers missing pixels, its ability to preserve consistency between poorly and well-exposed regions is inferior to that of DINet. Finally, combining the results of DINet and ExRNet using FusionNet further improves the synthesis performance by exploiting complementary information from the two networks. In addition, Fig. 6 shows the error maps for each network, which indicates that DINet and ExRNet yield complementary results, and FusionNet combines the complementary information to improve the synthesis performance.

Exposure-aware reconstruction: We analyze the effectiveness of the exposure-aware reconstruction using a mask **M** in (4) and (12) in DINet and ExRNet, respectively. Table 3 compares the average scores of these settings for both networks. The exposure-aware reconstruction using **M** improves the performances of both DINet and ExRNet significantly by enabling the networks to recover only missing regions.

MEF block: To analyze the effectiveness of the proposed MEF block, we train ExRNet with different settings. Table 4 compares the results. ExRNet without the MEF block provides the worst performance because valid information in multi-exposed features cannot be fully exploited. Using either of the channel and spatial fusions improves the restoration performance by exploiting the exposure information, and the use of both fusion strategies further improves the

Table 4. Impacts of fusion schemes in the MEF block on the restoration performance.

Fusion			pu-MSSSIM	pu-PSNR	log-PSNR
Channel	Spatial	$\boldsymbol{\mathcal{A}}^{(l)}$			
			0.9962	46.05	46.24
✓			0.9963	46.67	46.77
	✓		0.9963	46.63	46.80
✓	✓		0.9964	46.85	46.88
✓	✓	✓	**0.9965**	**46.96**	**47.01**

Table 5. Impacts of different demosaicing algorithms on the synthesis performance.

Demosaicing	pu-MSSSIM	pu-PSNR	log-PSNR	HDR-VDP		HDR-VQM
				Q	P	
Adams [1]	0.9963	45.22	42.29	70.96	**0.2252**	0.9697
Malvar *et al.* [21]	0.9962	44.99	41.21	73.61	0.4707	0.9675
Sharif *et al.* [39]	0.9968	46.01	42.91	73.98	0.4627	0.9713
Zhang *et al.* [57]	**0.9969**	**46.17**	**43.04**	**74.03**	0.3889	**0.9718**

performance. Finally, the element-wise weighting scheme using the weight $\boldsymbol{\mathcal{A}}^{(l)}$ yields the best performance.

Demosaicing: To analyze the effects of demosaicing algorithms on the synthesis performance, we test four demosaicing algorithms: two model-based algorithms [1,21], which were employed in conventional algorithms [4,5,6], and two learning-based algorithms [39,57]. Table 5 compares the results. The choice of demosaicing algorithm significantly affects the synthesis performance. In particular, Zhang *et al.*'s [57] yields the best overall performance.

5 Conclusions

We proposed a learning-based single-shot HDR imaging algorithm that recovers poorly exposed regions via exposure-aware dynamic weighted learning. The proposed algorithm consists of three networks: DINet, ExRNet, and FusionNet. DINet recovers poorly exposed pixels by learning local dynamic filters. ExRNet combines only valid features in well-exposed regions. To achieve this, we developed the MEF block to learn local and channel weights for exposure-aware feature fusion. FusionNet aggregates the outputs from DINet and ExRNet to produce the reconstructed images. Extensive experiments demonstrated that the proposed algorithm outperforms conventional algorithms on various datasets.

Acknowledgements

This work was supported in part by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. 2020-0-00011, Video Coding for Machine) and in part by the National Research Foundation of Korea (NRF) grant funded MSIP (No. 2022R1F1A1074402).

References

1. Adams, J.E.: Design of practical color filter array interpolation algorithms for digital cameras. In: Proc. SPIE. pp. 117–125 (Feb 1997) 8, 14
2. Alghamdi, M., Fu, Q., Thabet, A., Heidrich, W.: Transfer deep learning for reconfigurable snapshot HDR imaging using coded masks. Comput. Graph. Forum **40**(6), 90–103 (Mar 2021) 2
3. Aydın, T.O., Mantiuk, R., Seidel, H.P.: Extending quality metrics to full dynamic range images. In: Proc. SPIE, Human Vision and Electronic Imaging XIII. pp. 6806–10 (Jan 2008) 11
4. Cho, H., Kim, S.J., Lee, S.: Single-shot high dynamic range imaging using coded electronic shutter. Comput. Graph. Forum **33**(7), 329–338 (Oct 2014) 2, 3, 4, 14
5. Choi, I., Baek, S.H., Kim, M.H.: Reconstructing interlaced high-dynamic-range video using joint learning. IEEE Trans. Image Process. **26**(11), 5353–5366 (Nov 2017) 2, 3, 10, 11, 12, 14
6. Çoğalan, U., Akyüz, A.O.: Deep joint deinterlacing and denoising for single shot dual-ISO HDR reconstruction. IEEE Trans. Image Process. **29**, 7511–7524 (Jun 2020) 2, 3, 6, 10, 11, 12, 14
7. Debevec, P., Malik, J.: Recovering high dynamic range radiance maps from photographs. In: Proc. ACM SIGGRAPH. pp. 369–378 (Aug 1997) 1, 4
8. Eilertsen, G., Kronander, J., Denes, G., Mantiuk, R.K., Unger, J.: HDR image reconstruction from a single exposure using deep CNNs. ACM Trans. Graph. **36**(6), 178:1–178:15 (Nov 2017) 1
9. Endo, Y., Kanamori, Y., Mitani, J.: Deep reverse tone mapping. ACM Trans. Graph. **36**(6), 177:1–177:10 (Nov 2017) 1
10. Gu, J., Hitomi, Y., Mitsunaga, T., Nayar, S.K.: Coded rolling shutter photography: Flexible space-time sampling. In: Proc. ICCP. pp. 1–8 (Mar 2010) 2, 3, 4
11. Hajisharif, S., Kronander, J., Unger, J.: HDR reconstruction for alternating gain (ISO) sensor readout. In: Eurograph. pp. 25–28 (Apr 2014) 3, 4
12. Jia, X., De Brabandere, B., Tuytelaars, T., Gool, L.V.: Dynamic filter networks. In: Proc. NeurIPS (Dec 2016) 4
13. Kalantari, N.K., Ramamoorthi, R.: Deep high dynamic range imaging of dynamic scenes. ACM Trans. Graph. **36**(4), 144:1–144:12 (Jul 2017) 1, 9, 10
14. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: Proc. ICLR (Sep 2015) 10
15. Kronander, J., Gustavson, S., Bonnet, G., Unger, J.: Unified HDR reconstruction from raw CFA data. In: Proc. ICCP. pp. 1–9 (Apr 2013) 10
16. Lee, C., Lam, E.Y.: Computationally efficient truncated nuclear norm minimization for high dynamic range imaging. IEEE Trans. Image Process. **25**(9), 4145–4157 (Sep 2016) 1
17. Lee, S., An, G.H., Kang, S.J.: Deep recursive HDRI: Inverse tone mapping using generative adversarial networks. In: Proc. ECCV. pp. 613–628 (Sep 2018) 1
18. Li, J., Wang, N., Zhang, L., Du, B., Tao, D.: Recurrent feature reasoning for image inpainting. In: Proc. CVPR. pp. 7757–7765 (Jun 2020) 3
19. Liu, G., Reda, F.A., Shih, K.J., Wang, T.C., Tao, A., Catanzaro, B.: Image inpainting for irregular holes using partial convolutions. In: Proc. ECCV. pp. 89–105 (Sep 2018) 3
20. Liu, Y.L., Lai, W.S., Chen, Y.S., Kao, Y.L., Yang, M.H., Chuang, Y.Y., Huang, J.B.: Single-image HDR reconstruction by learning to reverse the camera pipeline. In: Proc. CVPR. pp. 1651–1660 (Jun 2020) 1

21. Malvar, H.S., He, L.W., Cutler, R.: High-quality linear interpolation for demosaicing of bayer-patterned color images. In: Proc. ICASSP. pp. 2274–2282 (May 2004) 8, 14
22. Mantiuk, R., Efremov, A., Myszkowski, K., Seidel, H.P.: Backward compatible high dynamic range MPEG video compression. ACM Trans. Graph. **25**(3), 713–723 (Jul 2006) 10
23. Mantiuk, R., Kim, K.J., Rempel, A.G., Heidrich, W.: HDR-VDP-2: A calibrated visual metric for visibility and quality predictions in all luminance conditions. ACM Trans. Graph. **30**(4) (Jul 2011) 11
24. Mantiuk, R., Myszkowski, K., Seidel, H.P.: Lossy compression of high dynamic range images and video. In: Proc. SPIE, Human Vision and Electronic Imaging. pp. 6057–6057–10 (Feb 2006) 10
25. Marnerides, D., Bashford-Rogers, T., Hatchett, J., Debattista, K.: ExpandNet: A deep convolutional neural network for high dynamic range expansion from low dynamic range content. Comput. Graph. Forum **37**(2), 37–49 (May 2018) 1
26. Martel, J.N.P., Müller, L.K., Carey, S.J., Dudek, P., Wetzstein, G.: Neural sensors: Learning pixel exposures for HDR imaging and video compressive sensing with programmable sensors. IEEE Trans. Pattern Anal. Mach. Intell. **42**(7), 1642–1653 (Jul 2020) 1, 2
27. Metzler, C.A., Ikoma, H., Peng, Y., Wetzstein, G.: Deep optics for single-shot high-dynamic-range imaging. In: Proc. CVPR. pp. 1372–1382 (Jun 2020) 2
28. Mitsunaga, T., Nayar, S.K.: Radiometric self calibration. In: Proc. CVPR. pp. 374–380 (Aug 1999) 1
29. Narwaria, M., Mantiuk, R., Perreira Da Silva, M., Le Callet, P.: HDR-VDP-2.2: A calibrated method for objective quality prediction of high dynamic range and standard images. J. Electron. Imaging **24**(1) (Jan 2015) 11
30. Narwaria, M., Perreira Da Silva, M., Le Callet, P.: HDR-VQM: An objective quality measure for high dynamic range video. Signal Process. Image Commun. **35**, 46–60 (Jul 2015) 11
31. Nayar, S.K., Mitsunaga, T.: High dynamic range imaging: Spatially varying pixel exposures. In: Proc. CVPR. pp. 472–479 (Jun 2000) 2
32. Pathak, D., Krähenbühl, P., Donahue, J., Darrell, T., Efros, A.A.: Context encoders: Feature learning by inpainting. In: Proc. CVPR. pp. 2536–2544 (Jun 2016) 3
33. Reinhard, E., Ward, G., Pattanaik, S., Debevec, P., Heidrich, W., Myszkowski, K.: High Dynamic Range Imaging: Acquisition, Display, and Image-Based Lighting. Morgan Kaufmann Publishers, second edn. (2010) 1
34. Robidoux, N., Capel, L.E.G., Seo, D.e., Sharma, A., Ariza, F., Heide, F.: End-to-end high dynamic range camera pipeline optimization. In: Proc. CVPR. pp. 6297–6307 (Jun 2021) 1
35. Ronneberger, O., Fischer, P., Brox, T.: U-Net: Convolutional networks for biomedical image segmentation. In: Proc. Med. Imag. Comput. Computer-Assiated Intervention. pp. 234–241 (Nov 2015) 6
36. Santos, M.S., Ren, T.I., Kalantari, N.K.: Single image HDR reconstruction using a CNN with masked features and perceptual loss. ACM Trans. Graph. **39**(4), 80:1–80:10 (Jul 2020) 1
37. Sen, P., Aguerrebere, C.: Practical high dynamic range imaging of everyday scenes: Photographing the world as we see it with our own eyes. IEEE Signal Process. Mag. **33**(5), 36–44 (Sep 2016) 1

38. Serrano, A., Heide, F., Gutierrez, D., Wetzstein, G., Masia, B.: Convolutional sparse coding for high dynamic range imaging. In: Eurograph. p. 153–163 (May 2016) 2
39. Sharif, S.M.A., Ali Naqvi, R., Biswas, M.: Beyond joint demosaicking and denoising: An image processing pipeline for a pixel-bin image sensor. In: Proc. CVPRW. pp. 233–242 (Jun 2021) 8, 10, 14
40. Suda, T., Tanaka, M., Monno, Y., Okutomi, M.: Deep snapshot HDR imaging using multi-exposure color filter array. In: Proc. ACCV. pp. 353–370 (Nov 2020) 2, 3, 6, 10, 11, 12
41. Sun, Q., Tseng, E., Fu, Q., Heidrich, W., Heide, F.: Learning rank-1 diffractive optics for single-shot high dynamic range imaging. In: Proc. CVPR. pp. 1383–1393 (Jun 2020) 2
42. Vargas, E., Martel, J.N., Wetzstein, G., Arguello, H.: Time-multiplexed coded aperture imaging: Learned coded aperture and pixel exposures for compressive imaging systems. In: Proc. ICCV. pp. 2692–2702 (Oct 2021) 2
43. Vaswani, A., *et al.*: Attention is all you need. In: Proc. NeurIPS. pp. 6000–6010 (Dec 2017) 7
44. Vien, A.G., Lee, C.: Single-shot high dynamic range imaging via multiscale convolutional neural network. IEEE Access **9**, 70369–70381 (2021) 2, 3, 6, 10, 11, 12
45. Wang, X., Girshick, R., Gupta, A., He, K.: Non-local neural networks. In: Proc. CVPR. pp. 7794–7803 (Jun 2018) 7
46. Wang, Y., Tao, X., Qi, X., Shen, X., Jia, J.: Image inpainting via generative multi-column convolutional neural networks. In: Proc. NeurIPS. pp. 329–338 (Dec 2018) 3
47. Woo, S.M., Ryu, J.H., Kim, J.O.: Ghost-free deep high-dynamic-range imaging using focus pixels for complex motion scenes. IEEE Trans. Image Process. **30**, 5001–5016 (May 2021) 2
48. Wu, S., Xu, J., Tai, Y.W., Tang, C.K.: Deep high dynamic range imaging with large foreground motions. In: Proc. ECCV. pp. 120–135 (Sep 2018) 1
49. Xie, C., Liu, S., Li, C., Cheng, M.M., Zuo, W., Liu, X., Wen, S., Ding, E.: Image inpainting with learnable bidirectional attention maps. In: Proc. ICCV. pp. 8857–8866 (Oct 2019) 3, 6
50. Xu, Y., Liu, Z., Wu, X., Chen, W., Wen, C., Li, Z.: Deep joint demosaicing and high dynamic range imaging within a single shot. IEEE Trans. Circuits Syst. Video Technol. (2021) 2, 3, 10, 11, 12
51. Yan, Q., Gong, D., Shi, Q., van den Hengel, A., Shen, C., Reid, I., Zhang, Y.: Attention-guided network for ghost-free high dynamic range imaging. In: Proc. CVPR. pp. 1751–1760 (Jun 2019) 1
52. Yan, Q., Zhang, L., Liu, Y., Zhu, Y., Sun, J., Shi, Q., Zhang, Y.: Deep HDR imaging via a non-local network. IEEE Trans. Image Process. **29**, 4308–4322 (Feb 2020) 1
53. Yan, Q., Zhang, L., Liu, Y., Zhu, Y., Sun, J., Shi, Q., Zhang, Y.: Deep HDR imaging via a non-local network. IEEE Trans. Image Process. **29**, 4308–4322 (Feb 2020) 7
54. Yang, F., Yang, H., Fu, J., Lu, H., Guo, B.: Learning texture transformer network for image super-resolution. In: Proc. CVPR. pp. 5790–5799 (Jun 2020) 7
55. Yu, J., Lin, Z., Yang, J., Shen, X., Lu, X., Huang, T.: Free-form image inpainting with gated convolution. In: Proc. ICCV. pp. 4470–4479 (Oct 2019) 3
56. Yu, J., Lin, Z., Yang, J., Shen, X., Lu, X., Huang, T.S.: Generative image inpainting with contextual attention. In: Proc. CVPR. pp. 5505–5514 (Jun 2018) 3

57. Zhang, Y., Li, K., Li, K., Zhong, B., Fu, Y.: Residual non-local attention networks for image restoration. In: Proc. ICLR (May 2019) 8, 10, 14
58. Zhao, H., Gallo, O., Frosio, I., Kautz, J.: Loss functions for image restoration with neural networks. IEEE Trans. Comput. Imaging **3**(1), 47–57 (Mar 2017) 9