Deep Unrolled Low-Rank Tensor Completion for High Dynamic Range Imaging

Truong Thanh Nhat Mai, *Student Member, IEEE*, Edmund Y. Lam, *Fellow, IEEE*, and Chul Lee, *Member, IEEE*

***Abstract*— The major challenge in high dynamic range (HDR) imaging for dynamic scenes is suppressing ghosting artifacts caused by large object motions or poor exposures. Whereas recent deep learning-based approaches have shown significant synthesis performance, interpretation and analysis of their behaviors are difficult and their performance is affected by the diversity of training data. In contrast, traditional model-based approaches yield inferior synthesis performance to learning-based algorithms despite their theoretical thoroughness. In this paper, we propose an algorithm unrolling approach to ghost-free HDR image synthesis algorithm that unrolls an iterative low-rank tensor completion algorithm into deep neural networks to take advantage of the merits of both learning- and model-based approaches while overcoming their weaknesses. First, we formulate ghost-free HDR image synthesis as a low-rank tensor completion problem by assuming the low-rank structure of the tensor constructed from low dynamic range (LDR) images and linear dependency among LDR images. We also define two regularization functions to compensate for modeling inaccuracy by extracting hidden model information. Then, we solve the problem efficiently using an iterative optimization algorithm by reformulating it into a series of subproblems. Finally, we unroll the iterative algorithm into a series of blocks corresponding to each iteration, in which the optimization variables are updated by rigorous closed-form solutions and the regularizers are updated by learned deep neural networks. Experimental results on different datasets show that the proposed algorithm provides better HDR image synthesis performance with superior robustness compared with state-of-the-art algorithms, while using significantly fewer training samples.**

***Index Terms*— High dynamic range (HDR) imaging, low-rank tensor completion, algorithm unrolling.**

I. Introduction

ADVANCEMENTS in digital imaging technology have enabled the capture of high-quality images. However, the dynamic ranges of natural scenes often exceed those of existing cameras [1]; thus, captured images contain under- and over-exposed regions, which degrade the overall image quality. To overcome the limited dynamic ranges of cameras, high dynamic range (HDR) imaging technologies have been developed to capture and reproduce the full range of luminance in natural scenes to improve users' visual experience [2]. One such approach is developing specialized HDR camera systems [3], [4]; however, such systems are often complex and expensive, which limits their practical use. Instead, software-based techniques are more prevalent, in which algorithms are used to reproduce HDR images from low dynamic range (LDR) images captured by conventional cameras. A common approach, called multi-exposure fusion (MEF), merges multiple LDR images taken with different exposure times to synthesize a single HDR image. However, camera or object motions between different exposures generate ghosting artifacts, which degrade the synthesized image quality. Therefore, it is essential to develop an MEF algorithm that can produce ghost-free HDR images; thus, various algorithms have been proposed.

Early attempts for ghost-free HDR imaging are based on motion models and can be broadly categorized into two groups: motion-rejection-based and alignment-based algorithms. Algorithms in the first category attempt to detect motion regions in the input LDR images and reduce their contributions in the fusion process [5], [6], [7], [8], [9], [10]. However, because these algorithms assume sufficiently small motion regions, they may generate visible artifacts in regions with large motions caused by information loss in the input images. By contrast, alignment-based algorithms employ correspondence estimation to align motion regions in the input LDR images before fusion [11], [12], [13], [14], [15]. Despite their ability to handle large motions, these algorithms may yield artifacts in the region with significant information loss, for example, caused by poor exposure or occlusion.

Deep learning-based algorithms, which employ convolutional neural networks (CNNs) to establish end-to-end networks that output HDR images directly from input LDR images, have recently been actively developed [16], [17], [18], [19], [20], [21], [22], [23], [24], [25], [26]. Owing to the powerful ability of CNNs to learn visual features, deep learning-based algorithms can restore scene irradiance with higher quality than the aforementioned algorithms. These deep learning-based algorithms are based on a common strategy: CNNs are used to encode input LDR images to the feature domain, and different CNNs subsequently merge and decode the feature maps back to the image domain. However, the performance of deep learning-based algorithms is limited

Manuscript received 30 March 2022; revised 21 July 2022; accepted 12 August 2022. Date of publication 1 September 2022; date of current version 7 September 2022. This work was supported by the National Research Foundation of Korea (NRF) Grant funded by the Korean Government through the Ministry of Science and ICT (MSIT) under Grant NRF-2022R1F1A1074402. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Nikolaos Mitianoudis. *(Corresponding author: Chul Lee.)*

Truong Thanh Nhat Mai and Chul Lee are with the Department of Multimedia Engineering, Dongguk University, Seoul 04620, South Korea (e-mail: mtntruong@mme.dongguk.edu; chullee@dongguk.edu).

Edmund Y. Lam is with the Department of Electrical and Electronic Engineering, The University of Hong Kong, Hong Kong (e-mail: elam@eee.hku.hk).

Digital Object Identifier 10.1109/TIP.2022.3201708

by the lack of diversity in training data [27]. Furthermore, it is difficult to analyze the behaviors of CNNs because their architectures are often empirically designed, thereby leading to black box models. Note, however, that interpretability is an essential factor in several applications [28].

Recently, a novel technique, called algorithm unrolling [29], [30], was developed to overcome the weaknesses of both the aforementioned model- and learning-based approaches while leveraging their merits. Specifically, in algorithm unrolling, iterations of model-based iterative algorithms are implemented as a series of layers in deep networks, connecting model-based algorithms with deep networks. Hence, deep networks become interpretable and more theoretically sound as they inherit the benefit of mathematical models. Because of its advantages in performance and generalization ability, algorithm unrolling has recently been applied in various image processing tasks [30], [31], [32], [33], [34]. In particular, Mai *et al*. [21] adopted an algorithm unrolling strategy for the first time for HDR imaging based on low-rank matrix completion by building an explainable deep network. Their algorithm explicitly considers ghost regions during the fusion process by strictly following the constraints of the mathematical model, while lost information in ghost regions is restored by the learned CNNs. However, because it processes each color channel independently using matrices, it cannot fully exploit the correlations between color channels and exposures, degrading the synthesized image quality.

In this work, to mitigate the aforementioned limitations of the state-of-the-art algorithms and better exploit the correlations among color channels and exposures, we propose an algorithm unrolling approach to HDR imaging based on low-rank tensor completion, called LRT-HDR (low-rank tensor-based HDR). First, we formulate HDR image synthesis as a low-rank tensor completion task based on the assumption of the linear dependency of the background irradiance with respect to exposure times. A low-rank tensor model better exploits the correlations between color channels and underlying high-dimensional scene structures than matrix models and is more theoretically rigorous and robust [8], [9], [10], [15] than other model-based algorithms. Nevertheless, because the approximation rather than the precise determination of the underlying structures of the scene by the low-rank model may cause inaccuracies in the model; thus, we define two general regularization functions, which extract hidden model information, to compensate for these inaccuracies. Then, we solve the optimization iteratively by employing an unrolling approach; in each iteration, the optimization variables and the regularizers are updated by closed-form solutions and learned deep networks, respectively. By incorporating the theoretical foundation from the low-rank model with deep learning-based compensation for the modeling inaccuracies, the proposed algorithm is inherently interpretable and achieves better generalizability than fully deep learning-based approaches, whose networks are heavily dependent on learned features and difficult to analyze. More specifically, the proposed algorithm requires significantly fewer training samples, equivalent to the amount of information learned from training data, than fully deep learning-based algorithms, since the synthesized image is less dependent on learned features, as will be experimentally verified.

To summarize, we make the following contributions.

- We formulate the MEF-based HDR image synthesis as low-rank tensor completion by representing a set of irradiance maps from the input LDR images and the moving foreground objects as a low-rank tensor and a sparse tensor, respectively, assuming a underlying static background. In addition, we define two regularization functions to compensate for inaccurate modeling. We then iteratively solve the optimization problem using the augmented Lagrange multiplier (ALM) method.
- We develop an algorithm unrolling approach to the ALM-based iterative algorithm to solve the optimization problem. Specifically, we implement iterations of the algorithm as a series of blocks in which the optimization variables are updated via the closed-form solutions and the regularizers are updated by CNNs. Therefore, the proposed LRT-HDR inherits the low-rank model's theoretical foundation, thereby making it inherently interpretable.
- We experimentally demonstrate that the proposed LRT-HDR algorithm can synthesize HDR images with higher quality than state-of-the-art algorithms [10], [11], [12], [16], [17], [18], [19], [20], [21] by overcoming their limitations, while using significantly fewer training samples. Furthermore, we show that LRT-HDR exhibits a superior generalization ability, as it provides the best overall results for datasets with different properties from the training dataset.

Note that this work is an extension of our conference paper [21], in which preliminary results have been presented in part. In this paper, we develop a new low-rank tensor completion formulation, which can leverage the correlations between color channels and between exposures, and subsequently employ an unrolling approach to solve it. To the best of our knowledge, this is the first attempt to incorporate low-rank tensor completion and deep networks in MEF-based HDR imaging. We demonstrate that the proposed algorithm with new contributions outperforms our previous work by large margins. Furthermore, we present new experiments that reveal the effectiveness and generalization ability of the proposed algorithm through comparisons using additional datasets, comparisons with more algorithms, and more comprehensive ablation studies.

The rest of this paper is organized as follows: Section II reviews related work. Section III describes the proposed LRT-HDR algorithm. Section IV discusses the experimental results. Finally, Section V concludes the paper.

II. Related Work

A. Ghost-Free HDR Imaging

A traditional model-based approach to ghost-free HDR imaging is to detect ghost regions across the input exposures and then alleviate their contributions to the synthesized HDR image. For example, Gallo *et al*. [5] estimated the probability of ghosting artifacts occurring in pixels, based on the deviation from a reference. Heo *et al*. [6] employed the joint probability density between exposures and energy minimization to detect

ghost regions. In [8], [9], and [10], rank minimization was employed for ghost-region detection. However, removing pixels in the ghost regions may cause information loss in the input images, thereby resulting in a loss of detail or visible artifacts in HDR images.

Another model-based approach is to first align input LDR images and then merge them to synthesize an HDR image. Sen *et al*. [11] formulated HDR image synthesis as a single optimization problem that jointly solves patch-wise alignment and HDR synthesis. Hu *et al*. [12] aligned LDR images by enforcing radiance and texture consistencies across exposures. Oh *et al*. [15] also jointly performed LDR image alignment and HDR synthesis by employing rank minimization to exploit the linear dependency of LDR images. These algorithms tend to generate artifacts in the regions of the synthesized HDR image, where the alignment fails due to large motions or poor exposures.

Recently, deep learning-based approaches have been developed actively. They often employed the encoder-decoder architecture; the LDR images are encoded into feature maps, which are then merged and decoded back to the HDR image [16], [17], [22]. In [18], [24], and [25], attention mechanisms were employed to assess the reliability of each pixel in the fusion process, which reduced the contributions of ghost regions to the synthesized HDR images. More recently, in [19], feature-level alignment and attention maps were combined to leverage the strengths of both mechanisms. Furthermore, Niu *et al*. [20] employed a generative adversarial network to fill missing textures caused by saturation and occlusion. Despite their superior performance to model-based algorithms, their performance is significantly affected by the diversity of training data [27]. In addition, interpretation and analysis of their behaviors is usually difficult, as features are processed in considerably higher-dimensional feature spaces.

B. Algorithm Unrolling

Algorithm unrolling provides a systematic connection between iterative algorithms and deep neural networks [30]. Specifically, iterations of an iterative optimization algorithm are implemented as a series of layers in a deep neural network [29]. Since iterative algorithms are widely used in image processing, algorithm unrolling has recently been employed in various image processing and computer vision tasks [30].

In particular, iterative algorithms for sparse coding and compressive sensing have benefited from algorithm unrolling. For example, in [34], [35], and [36], iterative algorithms for compressive sensing reconstruction models were expanded, each iteration of which was implemented as deep neural networks. Algorithm unrolling has also been employed in filtering-based image restoration algorithms to learn the filters for specific applications from training data, which were traditionally iteratively updated, *e.g*., diffusion filters for image restoration [31], measurement matrices for clutter suppression in ultrasound imaging [32], and blur kernels for blind image deblurring [33]. Additionally, imaging tasks that rely on handcrafted model information and are solved iteratively were shown to benefit from algorithm unrolling. For example, the prior terms in image deblurring [37], sparsity transforms in lensless image reconstruction [38], and dictionaries for sparse coding in super-resolution [39] were learned using algorithm unrolling.

Recently, Mai *et al*. [21] employed an algorithm unrolling strategy for HDR imaging based on low-rank matrix completion. Although their algorithm explicitly considers ghost regions during the fusion process by conforming to the constraints of the mathematical model, it is limited to 2D matrices and processes each color channel independently, thereby breaking the correlations between color channels and exposures. To overcome these limitations, in this work, we develop an unrolled deep network for HDR imaging based on low-rank tensor completion, which better exploits the multidimensional dependencies across color channels and exposures.

III. Proposed LRT-HDR Algorithm

In this section, we first introduce the notations used throughout this paper. We then formulate HDR imaging as a low-rank tensor completion problem and derive an iterative solution. Finally, we unroll the derived iterative solution by constructing a deep network.

A. Notations

Hereafter, tensors are denoted by bold calligraphic letters (*e.g*., $\boldsymbol{\mathcal{A}}$), matrices by bold uppercase letters (*e.g*., $\mathbf{A}$), and scalars by italicized Latin or Greek letters (*e.g*., n, K, α). The element in tensor $\boldsymbol{\mathcal{A}}$ at location $(i_1, \ldots, i_M)$ is denoted by $\boldsymbol{\mathcal{A}}(i_1, \ldots, i_M)$. Given a third-order tensor $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times n_2 \times n_3}$, we express its ith horizontal, lateral, and frontal slices as $\boldsymbol{\mathcal{A}}(i, :, :)$, $\boldsymbol{\mathcal{A}}(:, i, :)$, and $\boldsymbol{\mathcal{A}}(:, :, i)$, respectively. For brevity, the ith frontal slice is denoted by $\boldsymbol{\mathcal{A}}^{(i)}$. The Fourier transform $\mathcal{F}$ of tensor $\boldsymbol{\mathcal{A}}$ along the third dimension is denoted by $\widehat{\boldsymbol{\mathcal{A}}}$, *i.e*.,

$$\widehat{\boldsymbol{\mathcal{A}}}(i, j, :) = \mathcal{F}\{\boldsymbol{\mathcal{A}}(i, j, :)\}. \tag{1}$$

The inner product $\langle \cdot, \cdot \rangle$ of two tensors $\boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}} \in \mathbb{R}^{n_1 \times \cdots \times n_M}$ is defined as

$$\langle \boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}} \rangle = \sum_{i_1, \ldots, i_M} \boldsymbol{\mathcal{A}}(i_1, \ldots, i_M) \boldsymbol{\mathcal{B}}(i_1, \ldots, i_M). \tag{2}$$

The ℓ_1-norm $\|\cdot\|_1$ and Frobenius norm $\|\cdot\|_F$ of an Mth-order tensor $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{n_1 \times \cdots \times n_M}$ are defined as

$$\|\boldsymbol{\mathcal{A}}\|_1 = \sum_{i_1, \ldots, i_M} |\boldsymbol{\mathcal{A}}(i_1, \ldots, i_M)|, \tag{3}$$

$$\|\boldsymbol{\mathcal{A}}\|_F = \sqrt{\sum_{i_1, \ldots, i_M} |\boldsymbol{\mathcal{A}}(i_1, \ldots, i_M)|^2}, \tag{4}$$

respectively. For a matrix $\mathbf{A} \in \mathbb{R}^{n_1 \times n_2}$, its truncated nuclear norm $\|\cdot\|_r$ is defined as

$$\|\mathbf{A}\|_r = \sum_{i=r+1}^{\min(n_1, n_2)} \sigma_i(\mathbf{A}), \tag{5}$$

where $\sigma_i(\mathbf{A})$ is the ith largest singular value of $\mathbf{A}$. Finally, the orthogonal projection of a tensor onto the subspace corresponding to the set of observed entries $\boldsymbol{\Omega} \in \mathbb{N}^{n_1 \times \cdots \times n_M}$ is defined as

$$[\mathcal{P}_{\boldsymbol{\Omega}}(\boldsymbol{\mathcal{A}})](i_1, \ldots, i_M) = \begin{cases} \boldsymbol{\mathcal{A}}(i_1, \ldots, i_M), & \text{if } (i_1, \ldots, i_M) \in \boldsymbol{\Omega}, \\ 0, & \text{otherwise.} \end{cases} \tag{6}$$

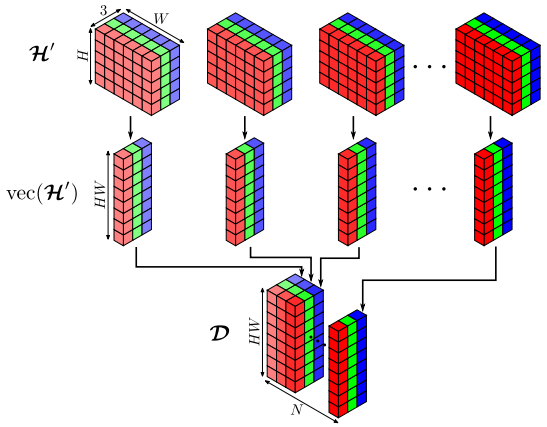


Fig. 1. Illustration of the creation of input data tensor $\mathcal{D}$.

B. Problem Formulation

Given $\mathcal{I} = \{\mathcal{I}_1, \ldots, \mathcal{I}_N\}$, a set of N LDR images of $H \times W \times 3$, where H and W are height and width, respectively, captured with different exposure times, we aim to synthesize an HDR image while suppressing ghosting artifacts caused by global and local motions. We first map the pixel values in the LDR images to the irradiance values $\mathcal{H}_i$ using the camera response function (CRF) as

$$\mathcal{H}_i = \frac{h^{-1}(\mathcal{I}_i)}{t_i}, \quad i = 1, \ldots, N, \tag{7}$$

where $h(\cdot)$ is the CRF applied elementwise, and t_i is the exposure time for the ith image with $t_i > t_{i-1}$ for $i \in [2, N]$. Then, we warp the irradiance maps to the reference image, which is at the middle of the sequence, using SIFT-Flow [40]; hence, a set of aligned irradiance maps $\mathcal{H}' = \{\mathcal{H}'_1, \ldots, \mathcal{H}'_N\}$ are obtained. Finally, the observed scene irradiance tensor $\mathcal{D} \in \mathbb{R}^{HW \times N \times 3}$ is constructed by stacking the vectorized frontal slices (color channels) of $\mathcal{H}'_i$, vec($\mathcal{H}'_i$), along the second dimension, *i.e.*,

$$\mathcal{D}(i, j, k) = [\text{vec}(\mathcal{H}'_j)](i, 1, k). \tag{8}$$

Fig. 1 illustrates the construction of $\mathcal{D}$ from $\mathcal{H}'$.

Given scene irradiance tensor $\mathcal{D}$, we formulate HDR image synthesis as a low-rank tensor completion problem. Specifically, we assume that $\mathcal{D}$ can be decomposed into two components: the desired irradiance background $\mathcal{X}$ and the alignment error $\mathcal{E}$. The lateral slices of $\mathcal{X}$ correspond to irradiance maps of the exposures, and they are linearly related to that of a static background; hence, $\mathcal{X}$ exhibits low-rankness. In addition, since the errors caused by misalignment and poor exposure occupy sufficiently small areas in the images, $\mathcal{E}$ is sparse, *i.e.*, most of its elements are zero. Moreover, due to the errors, only pixels in a set $\mathbf{\Omega}$ are observed. Therefore, restoration of the low-rank tensor $\mathcal{X}$ for ghost-free HDR image synthesis can be formulated as

$$\begin{aligned} \underset{\mathcal{X},\mathcal{E}}{\text{minimize}} \quad & \text{rank}(\mathcal{X}) + \lambda \|\mathcal{E}\|_0 \\ \text{subject to} \quad & \mathcal{P}_{\mathbf{\Omega}}(\mathcal{X} + \mathcal{E}) = \mathcal{P}_{\mathbf{\Omega}}(\mathcal{D}), \end{aligned} \tag{9}$$

where λ controls the balance between the tensor rank of $\mathcal{X}$ and the sparsity of $\mathcal{E}$, and $\mathcal{P}_{\mathbf{\Omega}}$ denotes the projection onto the set of reliable pixels $\mathbf{\Omega}$, as defined in (6).

We note that tensor ranks are not well defined due to their multidimensional structures. Various tensor rank definitions have been proposed [41], [42], [43], [44], and [45], which explore different aspects of low-rankness in tensor structures. In this work, we aim to minimize the rank of the frontal slices of $\mathcal{X}$ to enforce content consistency for the desired irradiance scene across all color channels, while preserving the correlations between channels. To this end, we adopt the tensor tubal multi-rank [46] to solve the rank minimization problem in (9), as this rank has concrete relationships with the low-rankness of frontal slices [47]. In this work, we consider three-channel color images; thus, the tubal multi-rank of $\mathcal{X}$ is defined as

$$\text{rank}^{\text{TMR}}(\mathcal{X}) = \left[\text{rank}(\widehat{\mathcal{X}}^{(1)}), \text{rank}(\widehat{\mathcal{X}}^{(2)}), \text{rank}(\widehat{\mathcal{X}}^{(3)})\right], \tag{10}$$

i.e., the matrix ranks of the frontal slices of the Fourier transform along the third dimension.

Furthermore, the exposures, *i.e.*, columns, of desired $\mathcal{X}^{(i)}$ are linearly dependent on the ith channel of a single static irradiance scene; consequently, $\mathcal{X}^{(i)}$ has rank 1. To enforce a target rank on $\mathcal{X}^{(i)}$, we employ the partial sum of the tubal nuclear norm (PSTNN) [48], a convex surrogate for the tubal multi-rank, which is defined as

$$\|\mathcal{X}\|_r^{\text{PSTNN}} = \sum_{i=1}^{3} \|\widehat{\mathcal{X}}^{(i)}\|_r, \tag{11}$$

where $\|\cdot\|_r$ is the truncated nuclear norm defined in (5). Because the PSTNN inherits the ability to enforce exact matrix ranks from the truncated nuclear norm, it can also constrain the tubal multi-rank to a pre-determined value r. Assuming rank-1 frontal slices, we set $r = 1$ in this work. Then, the optimization problem in (9) becomes

$$\begin{aligned} \underset{\mathcal{X},\mathcal{E}}{\text{minimize}} \quad & \|\mathcal{X}\|_{r=1}^{\text{PSTNN}} + \lambda \|\mathcal{E}\|_1 \\ \text{subject to} \quad & \mathcal{P}_{\mathbf{\Omega}}(\mathcal{X} + \mathcal{E}) = \mathcal{P}_{\mathbf{\Omega}}(\mathcal{D}). \end{aligned} \tag{12}$$

Note that, in (12), the models for $\mathcal{X}$ and $\mathcal{E}$, *i.e.*, the low-rankness of the scene and the sparse error due to misalignment, respectively, are based on certain assumptions. However, these models may fail to accurately represent all features of real scenes, such as the color distribution or underlying structures. To overcome the limitations of the aforementioned models, we add two general regularization functions, $f : \mathbb{R}^{HW \times N \times 3} \mapsto \mathbb{R}$ and $g : \mathbb{R}^{HW \times N \times 3} \mapsto \mathbb{R}$ for $\mathcal{X}$ and $\mathcal{E}$, respectively, to compensate for the modeling inaccuracy. We also consider noise in the observed pixels for real-world acquisition as similarly done in [10], *i.e.*, $\mathcal{P}_{\mathbf{\Omega}}(\mathcal{X} + \mathcal{E} + \mathcal{N}) = \mathcal{P}_{\mathbf{\Omega}}(\mathcal{D})$, where $\mathcal{N}$ is a noise tensor. Then, by introducing slack variables, the optimization problem in (12) can be rewritten as

$$\begin{aligned} \underset{\mathcal{X},\mathcal{E},\mathcal{T}}{\text{minimize}} \quad & \|\mathcal{X}\|_{r=1}^{\text{PSTNN}} + \lambda \|\mathcal{E}\|_1 + f(\mathcal{X}) + g(\mathcal{E}) \\ \text{subject to} \quad & \mathcal{X} + \mathcal{E} + \mathcal{T} = \mathcal{P}_{\mathbf{\Omega}}(\mathcal{D}), \\ & \|\mathcal{P}_{\mathbf{\Omega}}(\mathcal{T})\|_F \leq \delta, \end{aligned} \tag{13}$$

where $\mathcal{T}$ is a tensor of slack variables to compensate for unknown entries in $\mathcal{D}$, and $\delta \geq 0$ is the noise level.

C. Solution to the Optimization

We employ the ALM method [49] to iteratively solve the optimization problem in (13). To this end, we first reformulate (13) for variable splitting by introducing the auxiliary variables $\mathcal{P}$ and $\mathcal{Q}$ as

$$\begin{aligned}\underset{\mathcal{X},\mathcal{E},\mathcal{T},\mathcal{P},\mathcal{Q}}{\text{minimize}}\ & \|\mathcal{X}\|_{r=1}^{\text{PSTNN}}+\lambda\|\mathcal{E}\|_1+f(\mathcal{P})+g(\mathcal{Q})\\ \text{subject to}\ & \mathcal{P}=\mathcal{X},\ \mathcal{Q}=\mathcal{E},\\ & \mathcal{X}+\mathcal{E}+\mathcal{T}=\mathcal{P}_{\mathbf{\Omega}}(\mathcal{D}),\\ & \|\mathcal{P}_{\mathbf{\Omega}}(\mathcal{T})\|_F\le\delta.\end{aligned}\tag{14}$$

The ALM method solves a series of unconstrained subproblems derived from an original constrained optimization problem. Specifically, we define the augmented Lagrangian function $\mathcal{L}$ for problem (14) as

$$\begin{aligned}&\mathcal{L}(\mathcal{X},\mathcal{E},\mathcal{T},\mathcal{P},\mathcal{Q},\mathbf{\Lambda},\mathbf{\Gamma},\mathbf{\Phi})\\ &=\|\mathcal{X}\|_{r=1}^{\text{PSTNN}}+\lambda\|\mathcal{E}\|_1+\langle\mathbf{\Lambda},\mathcal{P}_{\mathbf{\Omega}}(\mathcal{D})-\mathcal{X}-\mathcal{E}-\mathcal{T}\rangle\\ &\quad+\frac{\mu}{2}\|\mathcal{P}_{\mathbf{\Omega}}(\mathcal{D})-\mathcal{X}-\mathcal{E}-\mathcal{T}\|_F^2\\ &\quad+f(\mathcal{P})+\frac{\alpha}{2}\|\mathcal{X}-\mathcal{P}\|_F^2+\langle\mathbf{\Gamma},\mathcal{X}-\mathcal{P}\rangle\\ &\quad+g(\mathcal{Q})+\frac{\beta}{2}\|\mathcal{E}-\mathcal{Q}\|_F^2+\langle\mathbf{\Phi},\mathcal{E}-\mathcal{Q}\rangle,\end{aligned}\tag{15}$$

where μ, α, and $\beta>0$ are penalty parameters; $\mathbf{\Lambda}$, $\mathbf{\Gamma}$, and $\mathbf{\Phi}\in\mathbb{R}^{HW\times N\times 3}$ are Lagrange multiplier tensors, and $\langle\cdot,\cdot\rangle$ denotes the inner product of two tensors defined in (2).

Solutions to the optimization problem in (14) can be obtained by minimizing $\mathcal{L}$ in (15), *i.e.*,

$$\begin{aligned}&(\mathcal{X}^*,\mathcal{E}^*,\mathcal{T}^*,\mathcal{P}^*,\mathcal{Q}^*)\\ &\quad=\underset{\mathcal{X},\mathcal{E},\mathcal{T},\mathcal{P},\mathcal{Q}}{\arg\min}\ \mathcal{L}(\mathcal{X},\mathcal{E},\mathcal{T},\mathcal{P},\mathcal{Q},\mathbf{\Lambda},\mathbf{\Gamma},\mathbf{\Phi}).\end{aligned}\tag{16}$$

However, joint optimization over the five variables in (16) is intractable in practice. Therefore, we employ the alternating direction method of multipliers [50], which splits the optimization over variables $\mathcal{X}$, $\mathcal{E}$, $\mathcal{T}$, $\mathcal{P}$, and $\mathcal{Q}$ and multipliers $\mathbf{\Lambda}$, $\mathbf{\Gamma}$, and $\mathbf{\Phi}$, and then solves them iteratively and individually. We describe how each subproblem is solved.

$\mathcal{X}$-subproblem: At the kth iteration, in the first step, we update $\mathcal{X}$ as

$$\begin{aligned}\mathcal{X}_{k+1}&=\underset{\mathcal{X}}{\arg\min}\ \mathcal{L}(\mathcal{X},\mathcal{E}_k,\mathcal{T}_k,\mathcal{P}_k,\mathcal{Q}_k,\mathbf{\Lambda}_k,\mathbf{\Gamma}_k,\mathbf{\Phi}_k)\\ &=\underset{\mathcal{X}}{\arg\min}\ \|\mathcal{X}\|_{r=1}^{\text{PSTNN}}+\langle\mathbf{\Lambda}_k,\mathcal{P}_{\mathbf{\Omega}}(\mathcal{D})-\mathcal{X}-\mathcal{E}_k-\mathcal{T}_k\rangle\\ &\quad+\frac{\mu_k}{2}\|\mathcal{P}_{\mathbf{\Omega}}(\mathcal{D})-\mathcal{X}-\mathcal{E}_k-\mathcal{T}_k\|_F^2\\ &\quad+\frac{\alpha_k}{2}\|\mathcal{X}-\mathcal{P}_k\|_F^2+\langle\mathbf{\Gamma}_k,\mathcal{X}-\mathcal{P}_k\rangle\\ &=\underset{\mathcal{X}}{\arg\min}\ \frac{1}{\mu_k+\alpha_k}\|\mathcal{X}\|_{r=1}^{\text{PSTNN}}+\frac{1}{2}\left\|\mathcal{X}-\mathbf{\Psi}_{\mathcal{X},k}\right\|_F^2,\end{aligned}\tag{17}$$

where $\mathbf{\Psi}_{\mathcal{X},k}=(\mu_k+\alpha_k)^{-1}(\mathbf{\Lambda}_k+\mu_k\mathcal{P}_{\mathbf{\Omega}}(\mathcal{D})-\mu_k\mathcal{E}_k-\mu_k\mathcal{T}_k+\alpha_k\mathcal{P}_k-\mathbf{\Gamma}_k)$. From the linearity of the Fourier transform and the definition of the PSTNN, the optimization in (17) can be separated into three matrix rank minimization problems on the frontal slices in the Fourier domain [48] as

$$\begin{aligned}\widehat{\mathcal{X}}_{k+1}^{(i)}&=\underset{\widehat{\mathcal{X}}^{(i)}}{\arg\min}\ \frac{1}{\mu_k+\alpha_k}\left\|\widehat{\mathcal{X}}^{(i)}\right\|_{r=1}\\ &\quad+\frac{1}{2}\left\|\widehat{\mathcal{X}}^{(i)}-\widehat{\mathbf{\Psi}}_{\mathcal{X},k}^{(i)}\right\|_F^2,\qquad\forall i\in\{1,2,3\}.\end{aligned}\tag{18}$$

We employ the partial singular value thresholding (PSVT) operator for complex matrices [48] to solve (18). Specifically, suppose that the singular value decomposition of a complex matrix $\mathbf{A}\in\mathbb{C}^{n_1\times n_2}$ is $\mathbf{A}=\mathbf{U}\Sigma\mathbf{V}^H$, where $\Sigma=\text{diag}(\sigma_1,\ldots,\sigma_{\min(n_1,n_2)})$. Then, the PSVT operator $\mathbb{P}_{r,\tau}$ is defined as

$$\mathbb{P}_{r,\tau}(\mathbf{A})=\mathbf{U}(\Sigma_1+\mathcal{S}_\tau(\Sigma_2))\mathbf{V}^H,\tag{19}$$

where $\Sigma_1=\text{diag}(\sigma_1,\ldots,\sigma_r,0,\ldots,0)$, $\Sigma_2=\text{diag}(0,\ldots,0,\sigma_{r+1},\ldots,\sigma_{\min(n_1,n_2)})$, and $\mathcal{S}_\tau(\cdot)$ is the element-wise soft-thresholding operator [51] with parameter $\tau>0$; that is, $[\mathcal{S}(\mathbf{A})](i,j)=\text{sign}(\mathbf{A}(i,j))\cdot\max\{|\mathbf{A}(i,j)|-\tau,0\}$. Then, the Fourier domain solutions for the frontal slices are given by

$$\widehat{\mathcal{X}}_{k+1}^{(i)}=\mathbb{P}_{r,\frac{1}{\mu_k+\alpha_k}}\left(\widehat{\mathbf{\Psi}}_{\mathcal{X},k}^{(i)}\right),\qquad\forall i\in\{1,2,3\}.\tag{20}$$

Finally, the closed-form solution to (17) is given by the inverse Fourier transform along the third dimension

$$\mathcal{X}_{k+1}(i,j,:)=\mathcal{F}^{-1}\left\{\widehat{\mathcal{X}}_{k+1}(i,j,:)\right\}.\tag{21}$$

$\mathcal{E}$-subproblem: Next, we estimate $\mathcal{E}$ by solving

$$\begin{aligned}\mathcal{E}_{k+1}&=\underset{\mathcal{E}}{\arg\min}\ \mathcal{L}(\mathcal{X}_{k+1},\mathcal{E},\mathcal{T}_k,\mathcal{P}_k,\mathcal{Q}_k,\mathbf{\Lambda}_k,\mathbf{\Gamma}_k,\mathbf{\Phi}_k)\\ &=\underset{\mathcal{E}}{\arg\min}\ \lambda_k\|\mathcal{E}\|_1+\langle\mathbf{\Lambda}_k,\mathcal{P}_{\mathbf{\Omega}}(\mathcal{D})-\mathcal{X}_{k+1}-\mathcal{E}-\mathcal{T}_k\rangle\\ &\quad+\frac{\mu_k}{2}\|\mathcal{P}_{\mathbf{\Omega}}(\mathcal{D})-\mathcal{X}_{k+1}-\mathcal{E}-\mathcal{T}_k\|_F^2\\ &\quad+\frac{\beta_k}{2}\|\mathcal{E}-\mathcal{Q}_k\|_{\text{F}}^2+\langle\mathbf{\Phi}_k,\mathcal{E}-\mathcal{Q}_k\rangle\\ &=\underset{\mathcal{E}}{\arg\min}\ \frac{\lambda_k}{\mu_k+\beta_k}\|\mathcal{E}\|_1+\frac{1}{2}\left\|\mathcal{E}-\mathbf{\Psi}_{\mathcal{E},k}\right\|_F^2,\end{aligned}\tag{22}$$

where $\mathbf{\Psi}_{\mathcal{E},k}=(\mu_k+\beta_k)^{-1}(\mathbf{\Lambda}_k+\mu_k\mathcal{P}_{\mathbf{\Omega}}(\mathcal{D})-\mu_k\mathcal{X}_{k+1}-\mu_k\mathcal{T}_k+\beta_k\mathcal{Q}_k-\mathbf{\Phi}_k)$, and λ_k is the value of λ in (14) at the kth iteration. The closed-form solution to (22) can be obtained using the soft-thresholding operator [51]

$$\mathcal{E}_{k+1}=\mathcal{S}_{\frac{\lambda_k}{\mu_k+\beta_k}}\left(\mathbf{\Psi}_{\mathcal{E},k}\right).\tag{23}$$

$\mathcal{T}$-subproblem: We update $\mathcal{T}$ by solving the following optimization problem.

$$\begin{aligned}\mathcal{T}_{k+1}&=\underset{\|\mathcal{P}_{\mathbf{\Omega}}(\mathcal{T})\|_F\le\delta_k}{\arg\min}\ \mathcal{L}(\mathcal{X}_{k+1},\mathcal{E}_{k+1},\mathcal{T},\mathcal{P}_k,\mathcal{Q}_k,\mathbf{\Lambda}_k,\mathbf{\Gamma}_k,\mathbf{\Phi}_k)\\ &=\underset{\|\mathcal{P}_{\mathbf{\Omega}}(\mathcal{T})\|_F\le\delta_k}{\arg\min}\ \langle\mathbf{\Lambda}_k,\mathcal{P}_{\mathbf{\Omega}}(\mathcal{D})-\mathcal{X}_{k+1}-\mathcal{E}_{k+1}-\mathcal{T}\rangle\\ &\quad+\frac{\mu_k}{2}\|\mathcal{P}_{\mathbf{\Omega}}(\mathcal{D})-\mathcal{X}_{k+1}-\mathcal{E}_{k+1}-\mathcal{T}\|_F^2\\ &=\underset{\|\mathcal{P}_{\mathbf{\Omega}}(\mathcal{T})\|_F\le\delta_k}{\arg\min}\ \|\mathcal{T}-\mathbf{\Psi}_{\mathcal{T},k}\|_F^2,\end{aligned}\tag{24}$$

where $\mathbf{\Psi}_{\mathcal{T},k}=\mathcal{P}_{\mathbf{\Omega}}(\mathcal{D})-\mathcal{X}_{k+1}-\mathcal{E}_{k+1}+\mu_k^{-1}\mathbf{\Lambda}_k$. We employ Theorem 1 of [10] to solve this problem. Specifically, the

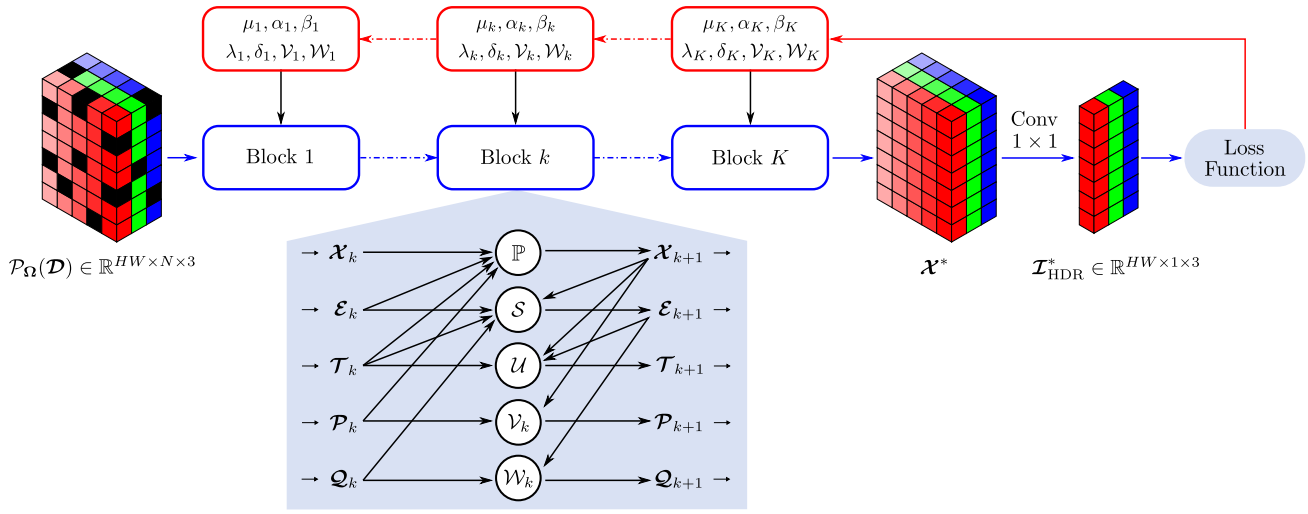


Fig. 2. LRT-HDR architecture. The network is composed of K blocks; each block represents an iteration of the iterative tensor completion algorithm. In each block, optimization variables $\boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{E}}$, and $\boldsymbol{\mathcal{T}}$ are updated by closed-form solutions, and $\boldsymbol{\mathcal{P}}$ and $\boldsymbol{\mathcal{Q}}$ are updated by CNNs. The black, blue, and red arrows indicate data flows, forward passing, and backpropagation, respectively. Finally, the HDR image is synthesized by applying a 1×1 convolutional filter to $\boldsymbol{\mathcal{X}}^*$.

closed-form solution is given by

$$\begin{aligned}\boldsymbol{\mathcal{T}}_{k+1} &= \mathcal{P}_{\boldsymbol{\Omega}^c}(\boldsymbol{\Psi}_{\boldsymbol{\mathcal{T}},k}) + \min\left\{\frac{\delta_k}{\|\mathcal{P}_{\boldsymbol{\Omega}}(\boldsymbol{\Psi}_{\boldsymbol{\mathcal{T}},k})\|_F}, 1\right\}\mathcal{P}_{\boldsymbol{\Omega}}(\boldsymbol{\Psi}_{\boldsymbol{\mathcal{T}},k}) \\ &= \mathcal{U}(\boldsymbol{\Psi}_{\boldsymbol{\mathcal{T}},k}), \end{aligned} \tag{25}$$

where $\mathcal{U}(\cdot)$ is used for a simpler notation.

$\boldsymbol{\mathcal{P}}$- and $\boldsymbol{\mathcal{Q}}$-subproblems: Then, $\boldsymbol{\mathcal{P}}$ and $\boldsymbol{\mathcal{Q}}$ are similarly updated as

$$\begin{aligned}\boldsymbol{\mathcal{P}}_{k+1} &= \arg\min_{\boldsymbol{\mathcal{P}}} \mathcal{L}(\boldsymbol{\mathcal{X}}_{k+1}, \boldsymbol{\mathcal{E}}_{k+1}, \boldsymbol{\mathcal{T}}_{k+1}, \boldsymbol{\mathcal{P}}, \boldsymbol{\mathcal{Q}}_k, \boldsymbol{\Lambda}_k, \boldsymbol{\Gamma}_k, \boldsymbol{\Phi}_k) \\ &= \arg\min_{\boldsymbol{\mathcal{P}}} f(\boldsymbol{\mathcal{P}}) + \frac{\alpha_k}{2}\|\boldsymbol{\mathcal{P}} - (\boldsymbol{\mathcal{X}}_{k+1} + \alpha_k^{-1}\boldsymbol{\Gamma}_k)\|_F^2 \\ &= \text{prox}_f(\boldsymbol{\mathcal{X}}_{k+1} + \alpha_k^{-1}\boldsymbol{\Gamma}_k), \end{aligned} \tag{26}$$

$$\begin{aligned}\boldsymbol{\mathcal{Q}}_{k+1} &= \arg\min_{\boldsymbol{\mathcal{Q}}} \mathcal{L}(\boldsymbol{\mathcal{X}}_{k+1}, \boldsymbol{\mathcal{E}}_{k+1}, \boldsymbol{\mathcal{T}}_{k+1}, \boldsymbol{\mathcal{P}}_{k+1}, \boldsymbol{\mathcal{Q}}, \boldsymbol{\Lambda}_k, \boldsymbol{\Gamma}_k, \boldsymbol{\Phi}_k) \\ &= \arg\min_{\boldsymbol{\mathcal{Q}}} g(\boldsymbol{\mathcal{Q}}) + \frac{\beta_k}{2}\|\boldsymbol{\mathcal{Q}} - (\boldsymbol{\mathcal{E}}_{k+1} + \beta_k^{-1}\boldsymbol{\Phi}_k)\|_F^2 \\ &= \text{prox}_g(\boldsymbol{\mathcal{E}}_{k+1} + \beta_k^{-1}\boldsymbol{\Phi}_k), \end{aligned} \tag{27}$$

where $\text{prox}_f(\cdot)$ and $\text{prox}_g(\cdot)$ are the proximal operators [52] corresponding to regularization functions $f(\cdot)$ and $g(\cdot)$, respectively. Traditionally, regularization functions have been determined by observing certain phenomena in applications, *e.g.*, fidelity of gradients [53], smoothness of illumination maps [54], sparsity [55], or low-rankness of non-local patches [56]. However, this approach may be inaccurate and limit the generalization ability of the model.

To overcome the limitations of traditional observation-based regularization functions, we design $f(\cdot)$ and $g(\cdot)$ such that they can express a wide variety of visual properties in real-world scenarios. To this end, we employ CNNs to implement proximal operators so that they can learn from training data and effectively reconstruct complex and diverse visual features. We denote the CNNs at the kth iteration as $\mathcal{V}_k$ and $\mathcal{W}_k$. Then, the closed-form solutions to the optimization problems in (26) and (27) can be written as

$$\boldsymbol{\mathcal{P}}_{k+1} = \mathcal{V}_k\left(\boldsymbol{\mathcal{X}}_{k+1} + \alpha_k^{-1}\boldsymbol{\Gamma}_k\right), \tag{28}$$

$$\boldsymbol{\mathcal{Q}}_{k+1} = \mathcal{W}_k\left(\boldsymbol{\mathcal{E}}_{k+1} + \beta_k^{-1}\boldsymbol{\Phi}_k\right), \tag{29}$$

respectively. During training, the network parameters of $\mathcal{V}_k$ and $\mathcal{W}_k$ are adjusted according to their input tensors $\boldsymbol{\mathcal{X}}_{k+1} + \alpha_k^{-1}\boldsymbol{\Gamma}_k$ and $\boldsymbol{\mathcal{E}}_{k+1} + \beta_k^{-1}\boldsymbol{\Phi}_k$, respectively, to produce the respective optimal solutions $\boldsymbol{\mathcal{P}}_{k+1}$ and $\boldsymbol{\mathcal{Q}}_{k+1}$.

Multipliers: Finally, the Lagrange multiplier tensors are updated following the strategy of the ALM method [49] as

$$\begin{aligned}\boldsymbol{\Lambda}_{k+1} &= \boldsymbol{\Lambda}_k + \mu_k(\mathcal{P}_{\boldsymbol{\Omega}}(\boldsymbol{\mathcal{D}}) - \boldsymbol{\mathcal{X}}_{k+1} - \boldsymbol{\mathcal{E}}_{k+1} - \boldsymbol{\mathcal{T}}_{k+1}), \\ \boldsymbol{\Gamma}_{k+1} &= \boldsymbol{\Gamma}_k + \alpha_k(\boldsymbol{\mathcal{X}}_{k+1} - \boldsymbol{\mathcal{P}}_{k+1}), \\ \boldsymbol{\Phi}_{k+1} &= \boldsymbol{\Phi}_k + \beta_k(\boldsymbol{\mathcal{E}}_{k+1} - \boldsymbol{\mathcal{Q}}_{k+1}). \end{aligned} \tag{30}$$

D. Unrolled Deep Network

Finally, we unroll iterations of the iterative tensor completion algorithm in the previous section into a series of blocks to establish an HDR imaging network. Fig. 2 illustrates the architecture of the proposed unrolled deep network for LRT-HDR. An observed tensor, $\boldsymbol{\mathcal{X}}_1 = \mathcal{P}_{\boldsymbol{\Omega}}(\boldsymbol{\mathcal{D}})$, is fed as input and all the other optimization variables are initialized to zeros. Then, the optimization variables are passed forward to K unrolled blocks. The operations in each block correspond to those in one iteration of the iterative algorithm in Section III-C, where $\boldsymbol{\mathcal{X}}, \boldsymbol{\mathcal{E}}$, and $\boldsymbol{\mathcal{T}}$ are updated by closed-form solutions, and $\boldsymbol{\mathcal{P}}$ and $\boldsymbol{\mathcal{Q}}$ are updated by CNNs. The output of the Kth block is the desired low-rank background irradiance map $\boldsymbol{\mathcal{X}}^*$. In addition to the network parameters for $\mathcal{V}_k$ and $\mathcal{W}_k$, the parameters μ_k and α_k in (17), β_k and λ_k in (22), and δ_k in (24) are implemented as learnable weights and updated by backpropagation during training. Note that the architecture strictly follows a mathematical model; hence, it is fully interpretable and allows analysis.

As $\mathcal{V}_k$ and $\mathcal{W}_k$ act as simple nonlinear mappings, they are structure-agnostic. Thus, any available CNNs can be employed for $\mathcal{V}_k$ and $\mathcal{W}_k$. For simplicity, in this work, we use a series

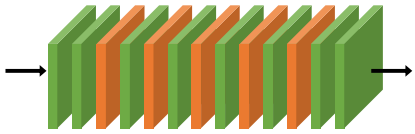


Fig. 3. Structure of $\mathcal{V}_k$ and $\mathcal{W}_k$ CNNs. The green and orange layers denote convolutional and ReLU layers, respectively.

of convolutional layers and rectified linear unit (ReLU) layers to construct both $\mathcal{V}_k$ and $\mathcal{W}_k$. Fig. 3 illustrates the structure of $\mathcal{V}_k$ and $\mathcal{W}_k$ CNNs. The convolutional layers have 128 filters of size 3×3, except for the last layer. The input and output of the first and last layer, respectively, have three channels. As shown in Fig. 3, both $\mathcal{V}_k$ and $\mathcal{W}_k$ have two convolutional layers at the beginning and end, with four convolutional layers and five ReLU layers alternately in the middle. The effects of different network structures will be discussed in Section IV-D.

The HDR image is synthesized by applying a 1×1 convolutional layer to $\boldsymbol{\mathcal{X}}^*$, which is equivalent to the weighted sum of elements in $\boldsymbol{\mathcal{X}}^*$ in the exposure dimension, *i.e.*,

$$\mathcal{I}^*_{\text{HDR}}(i, 1, k) = \sum_{j=1}^{N} w_j \boldsymbol{\mathcal{X}}^*(i, j, k), \tag{31}$$

where $\mathcal{I}^*_{\text{HDR}} \in \mathbb{R}^{HW \times 1 \times 3}$ and w_j are the weights for the 1×1 convolutional kernel without bias. We obtain the final HDR image by reshaping $\mathcal{I}^*_{\text{HDR}}$ to the original size of $H \times W \times 3$.

E. Training

To train the unrolled network, we define the loss function that incorporates both the fidelity of the reconstructed results and the rank constraint as

$$L_{\text{total}} = \omega L_{\text{recon}} + (1 - \omega) L_{\text{rank}}, \tag{32}$$

where L_{recon} and L_{rank} are the reconstruction loss and rank constraint loss, respectively, and ω is the weight that controls the contribution of each term. We compute the radiance reconstruction loss L_{recon} as the ℓ_1-norm of the difference between the ground-truth HDR image $\mathcal{I}_{\text{HDR}}$ and the synthesized HDR image $\mathcal{I}^*_{\text{HDR}}$, given by

$$L_{\text{recon}} = \|l(\mathcal{I}^*_{\text{HDR}}) - l(\mathcal{I}_{\text{HDR}})\|_1, \tag{33}$$

where $l(\cdot)$ denotes the conversion of the irradiance value to the perceptually uniform domain [57]. As the ground-truths $\boldsymbol{\mathcal{X}}_{\text{gt}}$ for $\boldsymbol{\mathcal{X}}^*$ are not readily available for the rank constraint loss, we construct $\boldsymbol{\mathcal{X}}_{\text{gt}}$ from $\mathcal{I}_{\text{HDR}}$. Specifically, because $\boldsymbol{\mathcal{X}}^{(i)}_{\text{gt}}$ has rank 1, the columns of $\boldsymbol{\mathcal{X}}^{(i)}_{\text{gt}}$ are identical and equal to $\mathcal{I}^{(i)}_{\text{HDR}}$. Thus, $\boldsymbol{\mathcal{X}}^{(i)}_{\text{gt}} \in \mathbb{R}^{HW \times N \times 3}$ can be constructed by replicating $\mathcal{I}^{(i)}_{\text{HDR}} \in \mathbb{R}^{HW \times 1 \times 3}$ by the number of exposures N, *i.e.*,

$$\boldsymbol{\mathcal{X}}^{(i)}_{\text{gt}} = \overbrace{\left[\mathcal{I}^{(i)}_{\text{HDR}}, \mathcal{I}^{(i)}_{\text{HDR}}, \ldots, \mathcal{I}^{(i)}_{\text{HDR}}\right]}^{N}, \quad \forall i \in \{1, 2, 3\}. \tag{34}$$

Then, the rank constraint loss is defined as

$$L_{\text{rank}} = \|l(\boldsymbol{\mathcal{X}}^*) - l(\boldsymbol{\mathcal{X}}_{\text{gt}})\|_1. \tag{35}$$

We use the Adam optimizer [58] with a batch size of 1 for 40 epochs. We begin with a learning rate of $\eta = 10^{-5}$ and decrease it by a factor of 0.1 at every tenth epoch. The training samples are constructed by randomly cropping 128×128 patches from images in the training dataset.

IV. Experimental Results

A. Settings

We evaluate the performance of the proposed LRT-HDR algorithm on two HDR video datasets—HDM-HDR [59] and HDRv [60]—and two non-reference real-scene datasets constructed by Sen *et al.* [11] and Tursun *et al.* [61], respectively. For HDM-HDR and HDRv, we randomly chose three consecutive frames from the videos and then generated multi-exposure images for each set using exposure biases of $\{-3, 0, +3\}$. Finally, we obtained 187 and 32 multi-exposure image sets from the HDM-HDR and HDRv datasets, respectively, with the corresponding ground-truth HDR images as the middle images. The image sets do not overlap, *i.e.*, the sets for training and testing are not from a single video. The source codes and datasets are available on our project website.[1]

We compare the performance of the proposed algorithm against those of Sen *et al.*'s algorithm [11], Hu *et al.*'s algorithm [12], TNNM-ALM [10], Kalantari and Ramamoorthi's algorithm [16], Wu *et al.*'s algorithm [17], AHDRNet [18], ADNet [19], HDR-GAN [20], and Mai *et al.*'s algorithm [21]. The first three are model-based algorithms, whereas the others are deep learning-based algorithms. The results of the conventional algorithms were obtained by executing the source codes provided by the respective authors with the recommended settings. We use a tone-mapping technique [62] to display synthesized HDR images.

We train deep learning-based algorithms, including the proposed LRT-HDR, using only 132 sets from the HDM-HDR dataset; the remaining 55 sets from this dataset are used for testing. Note that all the deep learning-based algorithms, excluding the proposed LRT-HDR algorithm, adopt data augmentation for performance improvement.

We define the observed region $\boldsymbol{\Omega} \in \mathbb{N}^{HW \times N \times 3}$ as a set of reliable pixel locations in terms of pixel value and structural information. First, we define the set of well-exposed pixel locations $\boldsymbol{\Omega}^e$ as

$$\boldsymbol{\Omega}^e = \{((k-1)H + j, i, l) | 0.01 \leq \mathcal{I}_i(j, k, l) \leq 0.99\}. \tag{36}$$

Next, we employ the structural similarity index (SSIM) [63] to consider the structural fidelity. Specifically, we define the set of well-warped pixel locations $\boldsymbol{\Omega}^w$ based on the SSIM score computed between the reference and warped images as

$$\boldsymbol{\Omega}^w = \{((k-1)H + j, i, l) \mid \text{SSIM}(\mathcal{H}'_i(j, k, l), \mathcal{H}'_{\text{ref}}(j, k, l)) \geq 0.90\}. \tag{37}$$

The pixel-wise SSIM scores are computed using an 11×11 window. Then, we define $\boldsymbol{\Omega} = \boldsymbol{\Omega}^e \cap \boldsymbol{\Omega}^w$. We set ω in (32) to 0.5, and the number of unrolled iterations to $K = 10$, unless otherwise specified. For the CRF, we employ the gamma correction function with a gamma value of 2.2 for consistency with previous works [16], [17], [18], [19], [20], [21].

[1] https://github.com/mtntruong/LRT-HDR

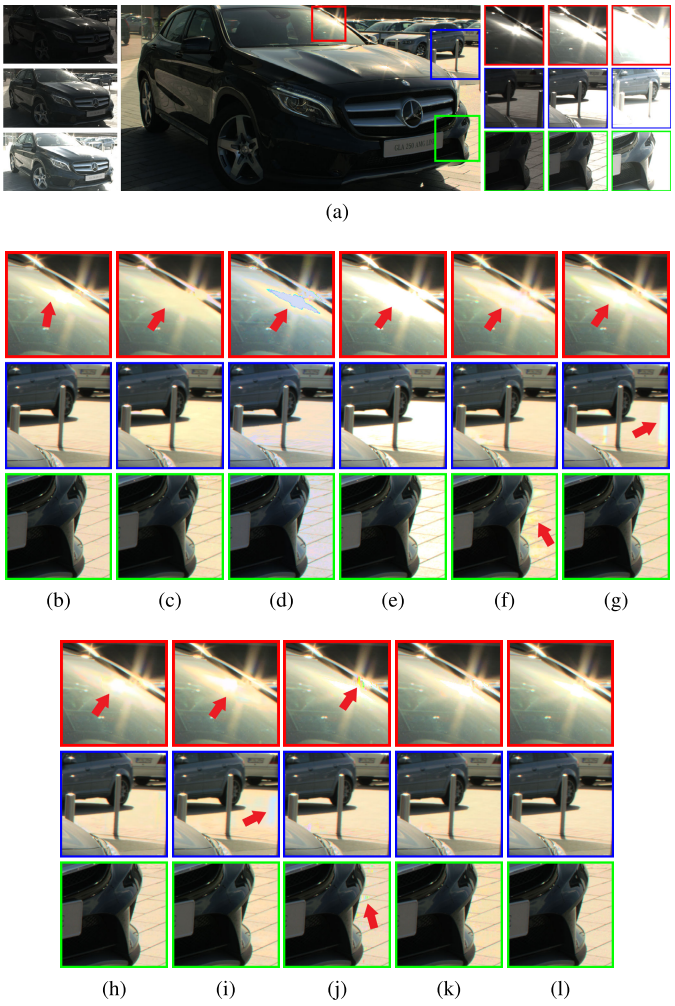


Fig. 4. Comparison of results for the 39th image of the HDM-HDR dataset. (a) LDR inputs and result of the proposed algorithm. The magnified regions marked by red, blue, and green rectangles of the synthesized results of (b) Sen *et al.* [11], (c) Hu *et al.* [12], (d) TNNM-ALM [10], (e) Kalantari and Ramamoorthi [16], (f) Wu *et al.* [17], (g) AHDRNet [18], (h) ADNet [19], (i) HDR-GAN [20], (j) Mai *et al.* [21], (k) the proposed algorithm, and (l) ground-truth.

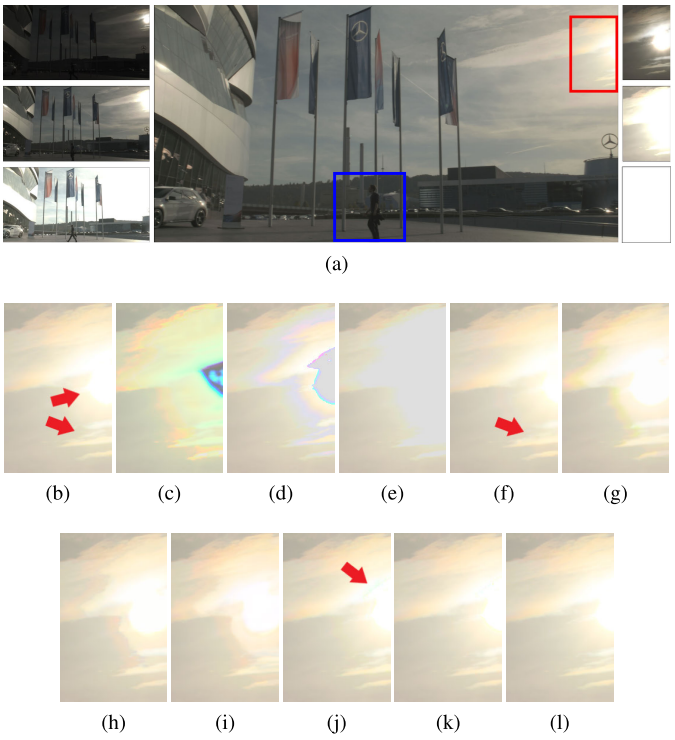


Fig. 5. Comparison of results for the 43rd image of the HDM-HDR dataset. (a) LDR inputs and result of the proposed algorithm. The magnified regions marked by red rectangle of the synthesized results of (b) Sen *et al.* [11], (c) Hu *et al.* [12], (d) TNNM-ALM [10], (e) Kalantari and Ramamoorthi [16], (f) Wu *et al.* [17], (g) AHDRNet [18], (h) ADNet [19], (i) HDR-GAN [20], (j) Mai *et al.* [21], (k) the proposed algorithm, and (l) ground-truth.

B. Subjective Assessment

Fig. 4 compares the synthesis results obtained by each algorithm for the 39th image set of the HDM-HDR dataset. The red arrows indicate the locations of interest where the artifacts are the most obvious. For example, the arrows in the red rectangles indicate artifacts in bright regions, whereas those in the blue and green rectangles point out ghosting artifacts and degraded textures, respectively. Sen *et al.*'s algorithm [11] and Hu *et al.*'s algorithm [12] in Figs. 4(b) and (c), respectively, inaccurately reconstruct bright regions when the corresponding regions in the reference LDR image are over-exposed. In Fig. 4(d), TNNM-ALM [10] fails to restore the information in the bright region, *i.e.*, red rectangle, which is saturated in all the input images. Since TNNM-ALM does not contain regularizers, it cannot compensate for information loss across all inputs, resulting in artifacts. Kalantari and Ramamoorthi's algorithm [16] in Fig. 4(e) fails to recover the bright regions faithfully, because it learns a simple weighting scheme that may fail if the information is improperly restored. Wu *et al.*'s algorithm [17] in Fig. 4(f) alters textures in the red rectangle and yields color artifacts in the green rectangle. AHDRNet [18], ADNet [19], and HDR-GAN [20] in Figs. 4(g)–(i) provide ghosting artifacts in the blue rectangles, which contain large over-exposed regions. This is because AHDRNet and ADNet incorrectly estimate attention maps, whereas HDR-GAN fails to compensate for global and local movements simultaneously. In Fig. 4(j), Mai *et al.*'s algorithm [21] exhibits color artifacts because of channel-wise matrix completion. On the contrary, LRT-HDR in Fig. 4(k) provides the synthesis result mostly identical to the ground-truth; this is due to the low-rank model and because the error term in the optimization effectively removes the ghost regions.

Fig. 5 compares the synthesis results of each algorithm for the image set, of which the under-exposed image in Fig. 5(a) contains large over-exposed regions. Sen *et al.*'s algorithm in Fig. 5(b) yields small ghosting artifacts due to misalignment on the clouds and the sun. Hu *et al.*'s algorithm, TNNM-ALM, and Kalantari and Ramamoorthi's algorithm in Fig. 5(c)–(e) yield severe color artifacts and fail to restore the textures in the bright regions of the sun. Wu *et al.*'s algorithm in Fig. 5(f) provides a better result; however, the cloud textures are altered because of the imprecise decoding process. AHDRNet, ADNet, and HDR-GAN in Figs. 5(g)–(i) yield severe visible artifacts because AHDRNet and ADNet fail to infer correct attention maps in the saturated regions of input images, and the learned features of HDR-GAN are less effective to restore those regions; therefore, the regions with less texture information in input images are fused in the result. In Fig. 5(j), Mai *et al.*'s algorithm provides greenish color artifacts in the

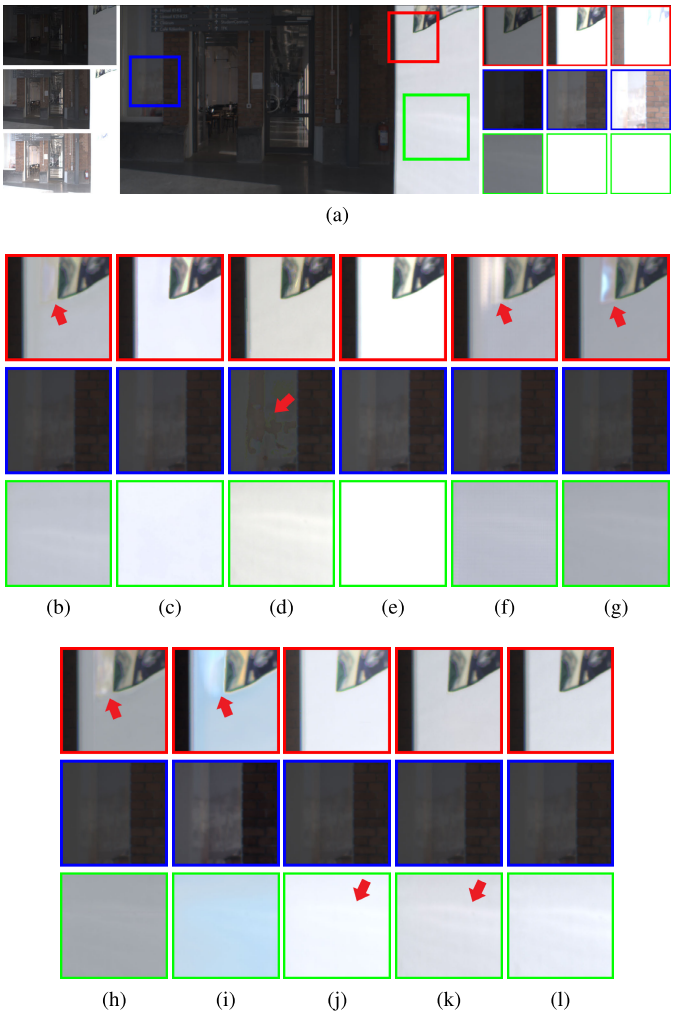


Fig. 6. Comparison of results for the 9th image of the HDRv dataset. (a) LDR inputs and result of the proposed algorithm. The magnified regions marked by red, blue, and green rectangles of the synthesized results of (b) Sen *et al.* [11], (c) Hu *et al.* [12], (d) TNNM-ALM [10], (e) Kalantari and Ramamoorthi [16], (f) Wu *et al.* [17], (g) AHDRNet [18], (h) ADNet [19], (i) HDR-GAN [20], (j) Mai *et al.* [21], (k) the proposed algorithm, and (l) ground-truth.

Fig. 7. Comparison of results for an image in the Tursun's dataset. (a) LDR inputs and synthesis result of the proposed algorithm. The synthesized results of (b) Sen *et al.* [11], (c) Hu *et al.* [12], (d) TNNM-ALM [10], (e) Kalantari and Ramamoorthi [16], (f) Wu *et al.* [17], (g) AHDRNet [18], (h) ADNet [19], (i) HDR-GAN [20], and (j) Mai *et al.* [21].

bright region because it cannot preserve relationships across different color channels. However, the proposed LRT-HDR algorithm in Fig. 5(k) yields a faithful restoration without noticeable artifacts by restoring information even in saturated regions.

Fig. 6 shows the synthesis results for the 9th image set from the HDRv dataset containing fast movement and strong illumination changes. In Figs. 6(b), (f)–(i), Sen *et al.*'s algorithm, Wu *et al.*'s algorithm, AHDRNet, ADNet, and HDR-GAN, respectively, yield strong ghosting artifacts in the red rectangle regions either because of incorrect alignment and attention map estimation or ineffective texture generation. Hu *et al.*'s and Kalantari and Ramamoorthi's algorithms in Figs. 6(c) and (e), respectively, again lose the textures in the bright regions in the green rectangles. TNNM-ALM and Mai *et al.*'s algorithm in Figs. 6(d) and (j) produce better results, but yield a small region of deterioration and detail loss, respectively, in the blue and green rectangles. In contrast, LRT-HDR in Fig. 6(k) provides the synthesis result without visible artifacts, and the fine textures in the bright region are faithfully preserved. Furthermore, the results in Fig. 6 for a dataset which was not used for training verify that the proposed LRT-HDR has superior generalization ability to other deep learning-based algorithms, because LRT-HDR primarily relies on low-rank formulation and is less dependent on learned features.

Finally, Figs. 7 and 8 compare the HDR synthesis results of each algorithm for non-reference real-scene images. Fig. 7 compares the synthesis results for the image set in the Tursun *et al.*'s dataset [61] that was captured using an exposure bias of $\{-2, 0, +2\}$. The proposed algorithm in Fig. 7(a) synthesizes the most natural result, which contains minimal visible artifacts. Fig. 8 shows synthesis results of the image set in the Sen *et al.*'s dataset [11], which was captured with an exposure bias of $\{-4, 0, +4\}$. The results are tone-mapped using darker settings for a better illustration of musical notes in the magnified regions. The proposed algorithm yields better texture reconstruction compared to the other state-of-the-art algorithms while effectively suppressing color artifacts.

C. Objective Assessment

To complement the subjective assessment, we objectively compare the HDR synthesis results of the proposed algorithm

TABLE I

QUANTITATIVE COMPARISON OF HDR SYNTHESIS PERFORMANCE ON THE HDM-HDR DATASET. THE ↑ AND ↓ SYMBOLS INDICATE "HIGHER IS BETTER" AND VICE VERSA, RESPECTIVELY. FOR EACH METRIC, THE **BOLDFACED** AND UNDERLINED NUMBERS DENOTE THE BEST AND THE SECOND-BEST RESULTS, RESPECTIVELY

	μ-PSNR (↑)	PU-PSNR (↑)	PU-MSSSIM (↑)	HDR-VDP (Q) (↑)	HDR-VDP (P) (↓)	# Training Samples	Augmentation
Sen *et al.* [11]	39.91	40.07	0.9849	61.21	0.2009	None	None
Hu *et al.* [12]	32.68	32.89	0.9743	57.90	0.5711	None	None
TNNM-ALM [10]	33.68	34.84	0.9640	57.28	0.3772	None	None
K. and R. [16]	38.21	37.96	0.9894	57.58	0.2211	2,000,000	Permute + flip + rotate
Wu *et al.* [17]	43.49	43.11	0.9950	66.36	0.2520	120,000	Flip + rotate
AHDRNet [18]	40.10	41.08	0.9886	64.18	0.4456	120,000	Flip + rotate
ADNet [19]	47.61	47.59	0.9979	66.90	0.0457	120,000	Flip + rotate
HDR-GAN [20]	34.41	35.33	0.9853	62.07	0.2916	120,000	Flip + rotate
Mai *et al.* [21]	48.94	49.01	0.9982	66.84	0.0339	66,000	Permute
Proposed	**49.54**	**49.65**	**0.9990**	**68.11**	**0.0273**	13,000	None

Fig. 8. Comparison of results for an image in the Sen *et al.*'s dataset. (a) LDR inputs and synthesis result of the proposed algorithm. The magnified regions marked by red rectangle of the synthesized results of (b) Sen *et al.* [11], (c) Hu *et al.* [12], (d) TNNM-ALM [10], (e) Kalantari and Ramamoorthi [16], (f) Wu *et al.* [17], (g) AHDRNet [18], (h) ADNet [19], (i) HDR-GAN [20], (j) Mai *et al.* [21], and (k) the proposed algorithm.

with those of conventional algorithms using five quality metrics: PSNR in the tone-mapped domain [16] (μ-PSNR), perceptually uniform extensions to PSNR (PU-PSNR) and multi-scale SSIM (PU-MSSSIM) [64], and HDR visible difference predictor (HDR-VDP) [65]. For HDR-VDP, we use the quality index score Q and the probability score P with the following parameters: a diagonal display size of 24 inches and a viewing distance of 0.5 meter, as in [17] and [20]. The metrics were averaged over 55 test image sets from the HDM-HDR dataset and 32 sets from the HDRv dataset.

Table I presents the objective assessment results for the algorithms on the HDM-HDR dataset. Note that the μ-PSNR and PU-PSNR metrics measure the fidelity of the reconstructed pixel values to the ground-truths, whereas PU-MSSSIM measures the structural similarities. The synthesized HDR images by the proposed algorithm are more faithful to the original HDR images than those by the conventional algorithms. Specifically, the μ-PSNR and PU-PSNR scores of the proposed algorithm exceed those of the other algorithms by large margins. The proposed algorithm also achieves the highest PU-MSSSIM score, indicating that the restored textures are the most similar to the ground-truths. Finally, the HDR-VDP (Q) index measures the perceptual differences between the reconstructed images and ground-truths, whereas the HDR-VDP (P) index indicates the probability of an average person noticing such differences. The proposed algorithm achieves the highest Q and lowest P, implying that the results are perceptually most faithful to the ground-truths.

Table I also reports the data augmentation procedures and approximate numbers of training samples used by the deep learning-based algorithms. Kalantari and Ramamoorthi (K. and R.)'s algorithm [16] uses the smallest patch size (40×40) while employing the largest number of augmentation combinations, thereby yielding a vast number of training samples. Wu *et al.*'s algorithm [17], AHDRNet [18], ADNet [19], and HDR-GAN [20] use the same patch sizes (256×256) and augmentation combinations, thereby constructing the same numbers of training samples. Mai *et al.*'s algorithm [21] uses 128×128 patches and employs color channel permutation to compensate for the limited effectiveness of applying matrix completion to color images. Notably, the proposed LRT-HDR algorithm uses 128×128 patches without data augmentation; in other words, it uses the fewest number of training samples. As the proposed algorithm relies on the theoretical foundation of the low-rank model, minimal learned information is required. Therefore, the proposed LRT-HDR can achieve better performance without augmentation. The effects of data augmentation on LRT-HDR will be discussed in Section IV-D.

Table II quantitatively compares the synthesis performance of the algorithms on the HDRv dataset, which was not used for training. The scores of the proposed algorithm remain the highest, proving its superior generalization ability. More specifically, the proposed algorithm yields significantly higher μ-PSNR and PU-PSNR scores than those of the conventional algorithms, indicating better reconstruction quality. ADNet achieves the highest PU-MSSSIM score; however, that of the proposed algorithm is comparable. This implies that the proposed algorithm and ADNet provide better restored textures than the other algorithms. Furthermore, the proposed algorithm

TABLE II
QUANTITATIVE COMPARISON OF HDR SYNTHESIS PERFORMANCE ON THE HDRV DATASET

	μ-PSNR (↑)	PU-PSNR (↑)	PU-MSSSIM (↑)	HDR-VDP (Q) (↑)	HDR-VDP (P) (↓)
Sen *et al.* [11]	53.50	55.45	0.9992	75.44	0.0226
Hu *et al.* [12]	38.45	38.76	0.9962	68.09	0.0809
TNNM-ALM [10]	44.58	46.59	0.9894	65.61	0.0819
Kalantari and Ramamoorth [16]	50.75	52.22	0.9987	73.37	0.0300
Wu *et al.* [17]	47.87	50.01	0.9978	67.74	0.1160
AHDRNet [18]	47.52	50.35	0.9941	69.84	0.1711
ADNet [19]	55.94	57.38	**0.9993**	74.05	0.0085
HDR-GAN [20]	39.98	40.83	0.9971	70.61	0.0899
Mai *et al.* [21]	54.40	55.50	0.9991	75.30	0.0069
Proposed	**56.52**	**57.97**	0.9992	**75.83**	**0.0049**

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

Fig. 9. Synthesized results at intermediate blocks. The magnified regions marked by blue rectangle in Fig. 5 of (a) LDR inputs, the synthesized results at blocks (b) $k = 2$, (c) $k = 4$, (d) $k = 6$, (e) $k = 8$, (f) $k = 10$ (final), and (g) ground-truth.

achieves the highest Q and lowest P. This implies greater similarities between the ground-truths and the HDR images synthesized by the proposed algorithm. To summarize, the proposed algorithm achieves the highest overall scores on the HDRv dataset as well while using the fewest training samples; this indicates that the theoretical foundation from the low-rank tensor completion model minimizes the dependency of the network on learned visual features, thereby increasing the generalization ability.

D. Model Analysis

We conduct ablation studies to analyze the interpretability of the deep networks, the effects of data augmentation, the network structures of $\mathcal{V}_k$ and $\mathcal{W}_k$, the number of unrolled iterations, and the loss functions on the synthesis performance.

1) Interpretability: As the proposed algorithm inherits the theoretical foundation of mathematical models and its iterative solution, the proposed network is fully interpretable. We analyze the interpretability of the network by visualizing intermediate results at selected unrolled blocks. Specifically, Fig. 9 shows the synthesized results of the region in the blue rectangle in Fig. 5 at every two blocks. As going through the blocks, equivalent to going through the iterations of the theoretical solution, ghosting artifacts are gradually removed while textures are gradually restored. The intermediate results show that the behavior of the proposed unrolled deep network follows that of the mathematical model, thereby providing interpretability and analyzability.

2) Augmentation: To analyze the effects of data augmentation on the synthesis performance, we train the proposed network with the three different augmentation strategies listed in Table I. Table III compares the performances on the HDM-HDR dataset. The data augmentation yields only a small performance improvement. In addition, the PU-MSSSIM scores are stable regardless of augmentation. For the other metrics, the differences between "None" and "Flip + rotate" and between "Permute" and "Permute + flip + rotate" are small, implying that geometric diversity is insignificant. This analysis shows that the proposed algorithm can achieve approximately peak performance without data augmentation while requiring the smallest training set, as the reconstruction of HDR images mostly relies on the low-rank model.

3) Network Structures: We analyze the effects of the network structures of $\mathcal{V}$ and $\mathcal{W}$ on the synthesis performance. To this end, we train the networks in Fig. 3 with different numbers of convolutional layers. More specifically, we add or remove pairs of convolutional and ReLU layers in the middle of the network. Table IV compares the synthesis performances of different settings on the HDM-HDR dataset. It is apparent that the network structures have a significant impact on the synthesis performances. As the number of convolutional layers increases from five to eight, the synthesis performance improves by increasing learned features. However, increasing the number of convolutional layers too much degrades the performance. This is because increasing the number of layers requires more parameters to be learned, making the learning difficult, thus yielding inaccurate synthesis.

4) Unrolled Iteration: Table V reports the effects of the number of unrolled iterations on the synthesis performance. When the algorithm is unrolled with five iterations, *i.e.*, $K = 5$, the number of operations and learned features are insufficient to ensure faithful synthesis, yielding the worst performance. As K increases, the synthesis performance improves, because the number of learned features increases. However, increasing K too much deteriorates the performance while increasing the computational and memory complexities. This is because the number of learnable parameters increases proportionally with an increase in K, thereby making it harder to converge and requiring more training samples. Therefore,

TABLE III
EFFECTS OF DATA AUGMENTATION ON THE SYNTHESIS PERFORMANCE

	# Training Samples	μ-PSNR (↑)	PU-PSNR (↑)	PU-MSSSIM (↑)	HDR-VDP (Q) (↑)	HDR-VDP (P) (↓)
None	13,000	49.54	49.65	0.9990	68.11	0.0273
Permute	78,000	49.64	49.84	0.9990	**68.14**	0.0259
Flip + rotate	120,000	49.57	49.67	0.9990	68.11	0.0273
Permute + flip + rotate	600,000	**49.66**	**49.85**	**0.9991**	**68.14**	**0.0258**

TABLE IV
EFFECTS OF NETWORK STRUCTURES ON THE SYNTHESIS PERFORMANCE

# conv.	μ-PSNR	PU-PSNR	PU-MSSSIM	HDR-VDP (Q)	HDR-VDP (P)
5	47.47	47.55	0.9982	66.29	0.0439
8	**49.54**	**49.65**	**0.9990**	**68.11**	**0.0273**
11	43.59	44.95	0.9951	64.65	0.0704

TABLE V
EFFECTS OF UNROLLED ITERATION NUMBERS ON THE SYNTHESIS PERFORMANCE

K	μ-PSNR	PU-PSNR	PU-MSSSIM	HDR-VDP (Q)	HDR-VDP (P)
5	47.47	47.82	0.9984	66.40	0.0412
10	**49.54**	**49.65**	**0.9990**	**68.11**	**0.0273**
15	49.34	49.41	0.9989	67.57	0.0289
20	48.55	48.75	0.9987	67.47	0.0292

TABLE VI
EFFECTS OF LOSSES ON THE SYNTHESIS PERFORMANCE

ω	μ-PSNR	PU-PSNR	PU-MSSSIM	HDR-VDP (Q)	HDR-VDP (P)
0	47.76	47.38	0.9979	66.87	0.0382
0.25	47.45	47.03	0.9979	66.50	0.0520
0.50	**49.54**	**49.65**	**0.9990**	**68.11**	**0.0273**
0.75	47.49	47.70	0.9980	66.34	0.0543
1	48.26	48.39	0.9986	66.85	0.0322

we chose $K = 10$ in this work to facilitate a desirable performance versus computation and memory trade-off.

5) Loss Functions: Finally, we discuss the effects of the losses on the synthesis performance by changing ω in (32). Table VI compares the results. When only the rank constraint or reconstruction fidelity is considered, *i.e.*, $\omega = 0$ or 1, the fidelity ($\omega = 1$) yields higher performance. However, when one of those components is considered more important that the other, *e.g.*, $\omega = 0.25$ or 0.75, the performance is worse than when $\omega = 0$ or 1. This is because the uneven contribution of the two losses hinders the convergence of the optimizer to satisfy the objective, making the training unstable. The best results are obtained when both the HDR reconstruction fidelity and the low-rankness of the output tensors contribute equally.

6) Limitation: Although the proposed LRT-HDR algorithm yields high-quality synthesized results without noticeable artifacts in most cases, it has a limitation that should be further addressed. Specifically, if the reference LDR image has saturated regions occluded in a shorter-exposed image, LRT-HDR fails to recover the textures in those regions faithfully. Fig. 10 shows an example of those artifacts, where the light bulbs in Fig. 10(b) are occluded in Fig. 10(a). Nevertheless,

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

Fig. 10. An example of failure cases of the proposed algorithm. (a)–(c) LDR inputs, the synthesized results of (d) Mai *et al.* [21], (e) the proposed algorithm, and (f) ground-truth.

TABLE VII
COMPARISON OF THE AVERAGE EXECUTION TIMES IN SECONDS AND THE NUMBER OF PARAMETERS

	Alignment	Main algorithm	# Param. (M)
Sen *et al.* [11]	-	70.97	-
Hu *et al.* [12]	-	223.14	-
TNNM-ALM [10]	26.77	21.71	-
K. and R. [16]	28.09	**0.20**	**0.38**
Wu *et al.* [17]	0.49	0.24	16.61
AHDRNet [18]	-	0.82	1.51
ADNet [19]	-	1.28	2.96
HDR-GAN [20]	-	0.36	2.56
Mai *et al.* [21]	26.77	63.44 (SVD: 57.72)	17.75
Proposed	26.77	18.23 (SVD: 17.61)	17.83

notice that LRT-HDR in Fig. 10(e) provides higher synthesis quality with less artifacts than the matrix completion-based approach, Mai *et al.*'s algorithm [21], in Fig. 10(d).

E. Computational Complexity

Table VII compares the average execution times of different algorithms over 55 test sets of resolution 1820×980 in HDM-HDR. In this test, we use a PC with a 3.6 GHz CPU and an Nvidia 2080Ti GPU. The proposed algorithm is the second most inefficient in terms of execution time. However, notice that the most time-consuming procedure in the main algorithm is the computation of the SVD, for which a PyTorch implementation is employed. Thus, if efficient techniques are employed for SVD [66], [67], the execution time would be significantly further reduced. Furthermore, efficient alignment techniques, *e.g.*, [68], can reduce the execution time, because different alignment algorithms do not impact the synthesis performance of the proposed algorithm significantly. Table VII also lists the number of parameters for learning-based algorithms. Although the structure of CNNs for LRT-HDR is simple as shown in Fig. 3, because a total of $2K$ CNNs are used, the number of parameters of the proposed algorithm is higher than those of conventional algorithms. However,

note that any available CNNs can be employed to reduce the number of parameters as mentioned previously.

V. Conclusion

We developed a ghost-free HDR image synthesis algorithm, called LRT-HDR, by unrolling a low rank tensor completion algorithm, which combines the advantages of both model- and learning-based approaches. By exploiting the low rank properties of tensors constructed from LDR images, we formulated HDR image synthesis as a low-rank tensor completion problem by defining learnable regularizers that extract hidden model information. Then, we iteratively solved the optimization problem using the ALM method. Finally, the iterative algorithm was unrolled for each iteration, wherein the optimization variables are updated by closed-form solutions and the regularizers by CNNs. Experimental results showed that the proposed LRT-HDR algorithm provides better HDR image synthesis performance than state-of-the-art algorithms while requiring significantly fewer training samples. Moreover, it was demonstrated that LRT-HDR exhibits better generalization ability than conventional end-to-end deep learning-based algorithms.

References

[1] C. A. Metzler, H. Ikoma, Y. Peng, and G. Wetzstein, "Deep optics for single-shot high-dynamic-range imaging," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 1372–1382.

[2] E. Reinhard, W. Heidrich, P. Debevec, S. Pattanaik, G. Ward, and K. Myszkowski, *High Dynamic Range Imaging: Acquisition, Display, and Image-based Lighting*, 2nd ed. San Mateo, CA, USA: Morgan Kaufmann, 2010.

[3] M. D. Tocci, C. Kiser, N. Tocci, and P. Sen, "A versatile HDR video production system," *ACM Trans. Graph.*, vol. 30, no. 4, p. 41, Jul. 2011.

[4] H. Zhao, B. Shi, C. Fernandez-Cull, S.-K. Yeung, and R. Raskar, "Unbounded high dynamic range photography using a modulo camera," in *Proc. ICCP*, Apr. 2015, pp. 1–10.

[5] O. Gallo, N. Gelfandz, M. Tico, W.-C. Chen, and K. Pulli, "Artifact-free high dynamic range imaging," in *Proc. IEEE ICCP*, Apr. 2009, pp. 1–7.

[6] Y. S. Heo, K. M. Lee, S. U. Lee, Y. Moon, and J. Cha, "Ghost-free high dynamic range imaging," in *Proc. Asian Conf. Comput. Vis.*, Nov. 2010, pp. 486–500.

[7] S. Raman and S. Chaudhuri, "Reconstruction of high contrast images for dynamic scenes," *Vis. Comput.*, vol. 27, no. 12, pp. 1099–1114, Dec. 2011.

[8] C. Lee, Y. Li, and V. Monga, "Ghost-free high dynamic range imaging via rank minimization," *IEEE Signal Process. Lett.*, vol. 21, no. 9, pp. 1045–1049, Sep. 2014.

[9] A. Bhardwaj and S. Raman, "Robust PCA-based solution to image composition using augmented Lagrange multiplier (ALM)," *Vis. Comput.*, vol. 32, no. 5, pp. 591–600, May 2016.

[10] C. Lee and E. Lam, "Computationally efficient truncated nuclear norm minimization for high dynamic range imaging," *IEEE Trans. Image Process.*, vol. 25, no. 9, pp. 4145–4157, Sep. 2016.

[11] P. Sen, N. K. Kalantari, M. Yaesoubi, S. Darabi, D. B. Goldman, and E. Shechtman, "Robust patch-based HDR reconstruction of dynamic scenes," *ACM Trans. Graph.*, vol. 31, no. 6, p. 203, Nov. 2012.

[12] J. Hu, O. Gallo, K. Pulli, and X. Sun, "HDR deghosting: How to deal with saturation?" in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 1163–1170.

[13] D. Hafner, O. Demetz, and J. Weickert, "Simultaneous HDR and optic flow computation," in *Proc. IEEE Int. Conf. Pattern Recognit.*, Aug. 2014, pp. 2065–2070.

[14] Y. Liu and Z. Wang, "Dense SIFT for ghost-free multi-exposure fusion," *J. Vis. Commun. Image Represent.*, vol. 31, pp. 208–224, Aug. 2015.

[15] T.-H. Oh, J.-Y. Lee, Y.-W. Tai, and I. S. Kweon, "Robust high dynamic range imaging by rank minimization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 6, pp. 1219–1232, Jun. 2015.

[16] N. K. Kalantari and R. Ramamoorthi, "Deep high dynamic range imaging of dynamic scenes," *ACM Trans. Graph.*, vol. 36, no. 4, p. 144, Jul. 2017.

[17] S. Wu, J. Xu, Y. W. Tai, and C. K. Tang, "Deep high dynamic range imaging with large foreground motions," in *Proc. Eur. Conf. Comput. Vis.*, Sep. 2018, pp. 120–135.

[18] Q. Yan *et al.*, "Attention-guided network for ghost-free high dynamic range imaging," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1751–1760.

[19] Z. Liu *et al.*, "ADNet: Attention-guided deformable convolutional network for high dynamic range imaging," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2021, pp. 463–470.

[20] Y. Niu, J. Wu, W. Liu, W. Guo, and R. W. H. Lau, "HDR-GAN: HDR image reconstruction from multi-exposed LDR images with large motions," *IEEE Trans. Image Process.*, vol. 30, pp. 3885–3896, 2021.

[21] T. T. N. Mai, E. Y. Lam, and C. Lee, "Ghost-free HDR imaging via unrolling low-rank matrix completion," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2021, pp. 2928–2932.

[22] Z. Pu, P. Guo, M. S. Asif, and Z. Ma, "Robust high dynamic range (HDR) imaging with complex motion and parallax," in *Proc. Asian Conf. Comput. Vis.*, Nov./Dec. 2020, pp. 134–149.

[23] S.-H. Lee, H. Chung, and N. I. Cho, "Exposure-structure blending network for high dynamic range imaging of dynamic scenes," *IEEE Access*, vol. 8, pp. 117428–117438, 2020.

[24] K. Metwaly and V. Monga, "Attention-mask dense merger (Attendense) deep HDR for ghost removal," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2020, pp. 2623–2627.

[25] S.-Y. Chen and Y.-Y. Chuang, "Deep exposure fusion with deghosting via homography estimation and attention learning," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2020, pp. 1464–1468.

[26] Z. Zheng, W. Ren, X. Cao, T. Wang, and X. Jia, "Ultra-high-definition image HDR reconstruction via collaborative bilateral learning," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 4449–4458.

[27] B. Kim, H. Kim, K. Kim, S. Kim, and J. Kim, "Learning not to learn: Training deep neural networks with biased data," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9004–9012.

[28] C. Rudin, "Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead," *Nature Mach. Intell.*, vol. 1, no. 5, pp. 206–215, May 2019.

[29] K. Gregor and Y. LeCun, "Learning fast approximations of sparse coding," in *Proc. 27th Int. Conf. Mach. Learn.*, 2010, pp. 399–406.

[30] V. Monga, Y. Li, and Y. C. Eldar, "Algorithm unrolling: Interpretable, efficient deep learning for signal and image processing," *IEEE Signal Process. Mag.*, vol. 38, no. 2, pp. 18–44, Feb. 2021.

[31] Y. Chen and T. Pock, "Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1256–1272, Jun. 2017.

[32] O. Solomon *et al.*, "Deep unfolded robust PCA with application to clutter suppression in ultrasound," *IEEE Trans. Med. Imag.*, vol. 39, no. 4, pp. 1051–1063, Apr. 2020.

[33] Y. Li, M. Tofighi, J. Geng, V. Monga, and Y. C. Eldar, "Efficient and interpretable deep blind image deblurring via algorithm unrolling," *IEEE Trans. Comput. Imag.*, vol. 6, pp. 666–681, 2020.

[34] J. T. Zhou, J. Du, H. Zhu, X. Peng, Y. Liu, and R. S. M. Goh, "AnomalyNet: An anomaly detection network for video surveillance," *IEEE Trans. Inf. Forensics Security*, vol. 14, no. 10, pp. 2537–2550, Oct. 2019.

[35] J. Zhang and B. Ghanem, "ISTA-Net: Interpretable optimization-inspired deep network for image compressive sensing," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1828–1837.

[36] S. Wu *et al.*, "Learning a compressed sensing measurement matrix via gradient unrolling," in *Proc. 36th Int. Conf. Mach. Learn.*, Jun. 2019, pp. 6828–6839.

[37] R. Liu, S. Cheng, Y. He, X. Fan, Z. Lin, and Z. Luo, "On the convergence of learning-based iterative methods for nonconvex inverse problems," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 12, pp. 3027–3039, Dec. 2020.

[38] K. Monakhova, J. Yurtsever, G. Kuo, N. Antipa, K. Yanny, and L. Waller, "Learned reconstructions for practical mask-based lensless imaging," *Opt. Exp.*, vol. 27, no. 20, pp. 28075–28090, Sep. 2019.

[39] Z. Wang, D. Liu, J. Yang, W. Han, and T. Huang, "Deep networks for image super-resolution with sparse prior," in *Proc. CVPR*, Dec. 2015, pp. 370–378.

[40] C. Liu, J. Yuen, and A. Torralba, "Sift flow: Dense correspondence across scenes and its applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 5, pp. 978–994, May 2011.

[41] F. L. Hitchcock, "The expression of a tensor or a polyadic as a sum of products," *J. Math. Phys.*, vol. 6, nos. 1–4, pp. 164–189, Apr. 1927.

[42] L. R. Tucker, "Some mathematical notes on three-mode factor analysis," *Psychometrika*, vol. 31, no. 3, pp. 279–311, Feb. 1966.

[43] I. V. Oseledets, "Tensor-train decomposition," *SIAM J. Sci. Comput.*, vol. 33, no. 5, pp. 2295–2317, Sep. 2011.

[44] M. E. Kilmer, K. Braman, N. Hao, and R. C. Hoover, "Third-order tensors as operators on matrices: A theoretical and computational framework with applications in imaging," *SIAM J. Matrix Anal. Appl.*, vol. 34, no. 1, pp. 148–172, Feb. 2013.

[45] Q. Zhao, G. Zhou, S. Xie, L. Zhang, and A. Cichocki, "Tensor ring decomposition," 2016, *arXiv:1606.05535*.

[46] O. Semerci, N. Hao, M. E. Kilmer, and E. L. Miller, "Tensor-based formulation and nuclear norm regularization for multienergy computed tomography," *IEEE Trans. Image Process.*, vol. 23, no. 4, pp. 1678–1693, Apr. 2014.

[47] C. Lu, J. Feng, Y. Chen, W. Liu, Z. Lin, and S. Yan, "Tensor robust principal component analysis: Exact recovery of corrupted low-rank tensors via convex optimization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 5249–5257.

[48] T.-X. Jiang, T.-Z. Huang, X.-L. Zhao, and L.-J. Deng, "Multi-dimensional imaging data recovery via minimizing the partial sum of tubal nuclear norm," *J. Comput. Appl. Math.*, vol. 372, Jul. 2020, Art. no. 112680.

[49] Z. Lin, M. Chen, L. Wu, and Y. Ma, "The augmented Lagrange multiplier method for exact recovery of corrupted low-rank matrices," Dept. Elect. Comput. Eng., Univ. Illinois, Urbana-Champaign, Champaign, IL, USA, Tech. Rep. UILU-ENG-09-2215, Nov. 2009.

[50] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, Jul. 2011.

[51] E. T. Hale, W. Yin, and Y. Zhang, "Fixed-point continuation for ℓ_1-minimization: Methodology and convergence," *SIAM J. Optim.*, vol. 19, no. 3, pp. 1107–1130, Oct. 2008.

[52] N. Parikh and S. Boyd, "Proximal algorithms," *Found. Trends Optim.*, vol. 1, no. 3, pp. 127–239, Jan. 2014.

[53] T. V. Nguyen, T. T. N. Mai, and C. Lee, "Single maritime image defogging based on illumination decomposition using texture and structure priors," *IEEE Access*, vol. 9, pp. 34590–34603, 2021.

[54] X. Ren, W. Yang, W.-H. Cheng, and J. Liu, "LR3M: Robust low-light enhancement via low-rank regularized retinex model," *IEEE Trans. Image Process.*, vol. 29, pp. 5862–5876, 2020.

[55] R. Giryes and M. Elad, "Sparsity-based Poisson denoising with dictionary learning," *IEEE Trans. Imag. Process.*, vol. 23, no. 12, pp. 5057–5069, Oct. 2014.

[56] W. Ren, X. Cao, J. Pan, X. Guo, W. Zuo, and M.-H. Yang, "Image deblurring via enhanced low-rank prior," *IEEE Trans. Image Process.*, vol. 25, no. 7, pp. 3426–3437, Jul. 2016.

[57] R. Mantiuk, K. Myszkowski, and H.-P. Seidel, "Lossy compression of high dynamic range images and video," *Proc. SPIE*, vol. 6057, pp. 311–320, Feb. 2006.

[58] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Represent.*, May 2015, pp. 1–15.

[59] J. Froehlich, S. Grandinetti, B. Eberhardt, S. Walter, A. Schilling, and H. Brendel, "Creating cinematic wide gamut HDR-video for the evaluation of tone mapping operators and HDR-displays," *Proc. SPIE*, vol. 9023, pp. 279–288, Mar. 2014.

[60] J. Kronander, S. Gustavson, G. Bonnet, A. Ynnerman, and J. Unger, "A unified framework for multi-sensor HDR video reconstruction," *Signal Process., Image Commun.*, vol. 29, no. 2, pp. 203–215, Feb. 2014.

[61] O. T. Tursun, A. O. Akyüz, A. Erdem, and E. Erdem, "An objective deghosting quality metric for HDR images," *Comput. Graph. Forum*, vol. 35, no. 2, pp. 139–152, May 2016.

[62] E. Reinhard, M. Stark, P. Shirley, and J. Ferwerda, "Photographic tone reproduction for digital images," *ACM Trans. Graph.*, vol. 21, no. 3, pp. 267–276, Jul. 2002.

[63] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

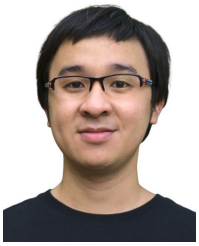
[64] T. O. Aydın, R. Mantiuk, and H.-P. Seidel, "Extending quality metrics to full luminance range images," in *Proc. SPIE*, Feb. 2008, pp. 109–118.

[65] R. Mantiuk, K. J. Kim, A. G. Rempel, and W. Heidrich, "HDR-VDP-2: A calibrated visual metric for visibility and quality predictions in all luminance conditions," *ACM Trans. Graph.*, vol. 30, no. 4, p. 40, 2011.

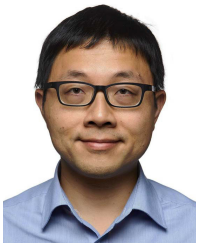
[66] Z. Allen-Zhu and Y. Li, "LazySVD: Even faster SVD decomposition yet without agonizing pain," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 29, Dec. 2016, pp. 1–9.

[67] T. Peken, S. Adiga, R. Tandon, and T. Bose, "Deep learning for SVD and hybrid beamforming," *IEEE Trans. Wireless Commun.*, vol. 19, no. 10, pp. 6621–6642, Oct. 2020.

[68] L. Kong, C. Shen, and J. Yang, "FastFlowNet: A lightweight network for fast optical flow estimation," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May/Jun. 2021, pp. 10310–10316.



Truong Thanh Nhat Mai (Student Member, IEEE) received the B.S. degree in mathematics and computer science from the Ho Chi Minh City University of Science, Ho Chi Minh City, Vietnam, in 2014, and the M.S. degree in electrical, electronic, and control engineering from Hankyong National University, Anseong, South Korea, in 2017. He is currently pursuing the Ph.D. degree with the Department of Multimedia Engineering, Dongguk University, Seoul, South Korea. His research interests include model-based deep learning for image processing, image restoration, and computational imaging.



Edmund Y. Lam (Fellow, IEEE) received the B.S., M.S., and Ph.D. degrees in electrical engineering from Stanford University. He was a Visiting Associate Professor with the Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology. He is currently a Professor of electrical and electronic engineering at The University of Hong Kong. He also serves as the Computer Engineering Program Director and a Research Program Coordinator with the AI Chip Center for Emerging Smart Systems. His research interests include computational imaging algorithms, systems, and applications. He is a fellow of Optica, SPIE, IS&T, and HKIE; and a Founding Member of the Hong Kong Young Academy of Sciences.

Chul Lee (Member, IEEE) received the B.S., M.S., and Ph.D. degrees in electrical engineering from Korea University, Seoul, South Korea, in 2003, 2008, and 2013, respectively.

From 2002 to 2006, he was with Biospace Inc., Seoul, where he involved in the development of medical equipment. From 2013 to 2014, he was a Postdoctoral Scholar with the Department of Electrical Engineering, Pennsylvania State University, University Park, PA, USA. From 2014 to 2015, he was a Research Scientist with the Department of Electrical and Electronic Engineering, The University of Hong Kong, Hong Kong. From 2015 to 2019, he was an Assistant Professor with the Department of Computer Engineering, Pukyong National University, Busan, South Korea. In March 2019, he joined the Department of Multimedia Engineering, Dongguk University, Seoul, where he is currently an Associate Professor. His current research interests include image processing and computational imaging with an emphasis on restoration and high dynamic range imaging.

Dr. Lee received the Best Paper Award from the *Journal of Visual Communication and Image Representation* in 2014. He is also an Editorial Board Member of the *Journal of Visual Communication and Image Representation*.