
Wenhui Wu\(^1\)  Jian Weng\(^2\)  Pingping Zhang\(^3\)  Xu Wang\(^2\)\(^*\)  Wenhan Yang\(^4\)  Jianmin Jiang\(^2\)
\(^1\)College of Electronics and Information Engineering, Shenzhen University
\(^2\)College of Computer Science and Software Engineering, Shenzhen University
\(^3\)Department of Computer Science, City University of Hong Kong
\(^4\)School of Electrical and Electronic Engineering, Nanyang Technological University

Abstract

Retinex model-based methods have shown to be effective in layer-wise manipulation with well-designed priors for low-light image enhancement. However, the commonly used hand-crafted priors and optimization-driven solutions lead to the absence of adaptivity and efficiency. To address these issues, in this paper, we propose a Retinex-based deep unfolding network (URetinex-Net), which unfolds an optimization problem into a learnable network to decompose a low-light image into reflectance and illumination layers. By formulating the decomposition problem as an implicit priors regularized model, three learning-based modules are carefully designed, responsible for data-dependent initialization, high-efficient unfolding optimization, and user-specified illumination enhancement, respectively. Particularly, the proposed unfolding optimization module, introducing two networks to adaptively fit implicit priors in data-driven manner, can realize noise suppression and details preservation for the final decomposition results. Extensive experiments on real-world low-light images qualitatively and quantitatively demonstrate the effectiveness and superiority of the proposed method over state-of-the-art methods. The code is available at https://github.com/AndersonYong/URetinex-Net.

1. Introduction

Images captured in a poor light environment always suffer from low contrast and low visibility, which pose challenges for both human visualization and numerous high-level vision tasks such as object detection [21, 23, 35]. To uncover the buried details in the low-light image and avoid degenerated performance of the subsequent vision tasks, researchers have made great efforts on contrast enhancement, texture recovering and noise removal for the low-light image. Especially for the low-light image enhancement (LLIE), many methods have been proposed, including histogram equalization [29], unsharp masking algorithms [7], Retinex-based methods [12, 27], multiple exposure fusion [5], and deep learning-based methods [11, 34].

Since Retinex theory well models color perception of human vision, LLIE methods based on Retinex theory have attracted much attention. As stated in Retinex theory, an image can be decomposed into two components, i.e., reflectance and illumination. Mathematically, the observed image \(I\) can be expressed by

\[ I = R \cdot L, \]  

where \(R\), \(L\) and \(\cdot\) denote reflectance, illumination and element-wise multiplication, respectively. In some early Retinex-based methods [16, 17], illumination is first estimated, and then reflectance is treated as the final enhanced results. Although details can be largely recovered from the input, it often leads to an unnatural and over-exposed look. Afterward, a number of model-based methods which have
good interpretability are proposed to solve the ill-posed decomposition problem in Eq. (1), where various hand-crafted priors are designed as the regularization terms introduced into models [12,13,22,27,30]. Then, to reveal the low-light image, illumination is further brightened up by Gamma correction. Designing explicit prior to fit data is the key to making models well-work, but it is challenging for model-based methods to be adaptive enough in various scenes. Furthermore, most model-based methods adopting conventional iteration optimization schemes are costly for a single image adjustment, which will hinder their development in real-time applications.

Due to these limitations existing in model-based methods, researchers take advantage of deep networks [3,11,15,25,34,42] to restore low-light images in a data-driven manner. Among these learning-based methods, Retinex-based ones [34,39,41] use deep networks to estimate reflectance and illumination, and brighten up illumination. However, most of these methods perform denoising operations on reflectance after decomposition, resulting in the loss of details. Furthermore, learning-based methods suffer from a lack of interpretability and flexibility, which brings difficulties in analyzing the potential limitations of the designed networks.

To this end, we propose a Retinex-based deep unfolding network (URetinex-Net) to reveal low-light images in RGB color space. To integrate the strengths from model-based and learning-based methods, we formulate Retinex-based decomposition problem as an implicit priors regularized model, where robust regularization terms are inferred by learnable networks instead of using hand-crafted priors. Specifically, the energy function of the formulated model is split into four univariate subproblems via alternating half-quadratic splitting algorithm, and this optimization problem can be solved by iteratively minimizing four subproblems. Then, we unfold the optimization scheme into a deep network. For subproblems regarding to prior terms, two networks are introduced to adaptively fit implicit priors, while the others regarding to the fidelity term are solved by corresponding close-form solutions. During unfolding optimization, the decomposed reflectance and illumination step-wise get rid of degradation (see Fig. 1). Meanwhile, the formulated model avoids designing explicit prior terms. Furthermore, considering the important effect of initialization on optimization, we propose an initialization module to favor the optimization. Finally, we design an illumination adjustment module to flexible brighten up illumination map according to user-specified light level.

In summary, the contributions of this paper lie threefold:

- Based on traditional model-based methods, we propose a novel deep unfolding network for LLIE (URetinex-Net), consisting of three functionally clear modules corresponding to initialization, optimization, and illumination adjustment, respectively, which inherits the flexibility and interpretability from model-based methods.
- The optimization module in our proposed URetinex-Net unfolds optimization procedure into a deep network, which leverages the powerful model ability of learning-based methods to adaptively fit data-dependent priors.
- Extensive experiments on real-world datasets are conducted to demonstrate high efficiency and superiority of our URetinex-Net, which can realize noise suppression and details preservation for the final enhanced results.

2. Related Work

2.1. Retinex Based LLIE Methods

**Model-Based Methods:** Classic Retinex theory models the Human Visual System (HVS), which assumes that the observed color depends on the intrinsic components of the object itself and the extrinsic non-uniform light source falls onto the object. Naturally, the image can be decomposed into reflectance and illumination as denoted in Eq. (1).

Several Retinex decomposition models [9,18,27] have been proposed under variational frameworks. Then adjusting the estimated illumination, the target low-light image is restored. Afterward, several model-based methods whose energy functions are under Maximum a posteriori (MAP) framework are proposed [12,13,22,30]. Guo et al. [12] proposed a structure-aware regularization model to refine the illumination map based on the initial one. In order to model degradation caused by noise, Li et al. [22] further introduced a noise term into the objective function to help remove noise while amplifying the details. Hao et al. [13] use Gaussian total variation as the regularization term to build decomposition model. In general, conventional model-based methods most rely on carefully designed hand-craft priors or certain statistical models. However, such priors are limited by model capacity when applied in various scenes.

**Deep Learning-Based Methods:** In the past decade, deep learning-based methods have provided promising results for LLIE problems [20]. Inspired by Retinex theory, Wei et al. [34] proposed an end-to-end trainable network named Retinex-Net, which includes a decomposition module and an illumination adjustment module. To realize denoising operation in Retinex-Net, BM3D [6] is used as a post-processing layer for reflectance, which implies that Retinex-Net can not handle heavy noise lying in extremely low-light images. Following [34], KinD [41] adopts a trainable denoising module for reflectance restoration. Moreover, a learnable mapping function is designed in the illumination adjustment module, in which images can be flexibly
restored under a user-specific light level. More recently, inspired by Retinex theory combined with maximum entropy, Zhang et al. [39] proposed a self-supervised framework utilizing histogram equalization operator to impose the constraint on reflectance.

Although these methods have shown remarkable performance on LLIE, they lack interpretability which will hinder their development on LLIE. Besides, based on the theory that reflectance depicting intrinsic components should be consistent under different light environments, most Retinex-based deep learning methods restore reflectance after decomposition, which will result in the loss of details [22].

2.2. Deep Unfolding Methods

Model-based LLIE methods are highly interpretable and flexible, while learning-based LLIE methods show superiority in learning complicated mapping in a data-driven manner. In addition, deep neural networks perform fast during inference, which is particularly computationally efficient. The unfolding (or unrolling) algorithm leveraging the strengths lying in model-based and learning-based methods has attracted much attention in the past decade.

Gregor and Lecun [10] first designed a time-unfolded algorithm to solve the iterative shrinkage and thresholding algorithm in the optimization of sparse coding, such that the proposed algorithm produces competitive performance within fewer iterations. Inspired by such optimization scheme, deep unfolding algorithms have made great impact on many significant image processing problems, such as super-resolution [33] [36], image denoising [3] [38], clutter suppression [31], and rain removal [8].

Recently, Liu et al. [24] proposed an unfolding framework for both illumination estimation and noise removal in an unsupervised manner, while the mutual connection between reflectance and illumination will be ignored in this way. Our method is different from it in two main aspects: (1) we tend to simultaneously estimate reflectance and illumination of the input in a unified framework; (2) our network can flexibly enhance illumination via a user-defined ratio.

3. The Proposed Unfolding Network

In this section, we first introduce the formulation of our proposed method and then present the details of the framework.

3.1. Problem Formulation

Classic Retinex-based model assumes that image can be decomposed into reflectance and illumination via Eq. (1), and various hand-crafted priors are developed to solve this ill-posed decomposition problem under MAP framework. Therefore, the reflectance and illumination can be obtained by minimizing the following regularized energy function:

$$E(R, L) = \| I - R \cdot L \|^2_F + \alpha \Phi(R) + \beta \Psi(L), \quad (2)$$
where $\| \cdot \|_F$ denotes Frobenius norm, $\| I - R \cdot L \|_F^2$ is the fidelity term derived from Eq. (1), $\Phi(R)$ and $\Psi(L)$ are regularization terms denoting imposed priors over $R$ and $L$, respectively, and $\alpha$ and $\beta$ are trade-off parameters.

Generally, to facilitate optimization, the fidelity term and the regularization terms are handled separately, such that we introduce two auxiliary variables $P$ and $Q$ to approximate $R$ and $L$, respectively. Accordingly, this leads to the following minimization problem:

$$
\min_{P,Q,R,L} \| I - P \cdot Q \|_F^2 + \alpha \Phi(R) + \beta \Psi(L)
$$

$$
s.t. \quad P = R, \quad Q = L.
$$

To deal with the equality constraints, two quadratic penalty terms are introduced, and problem is rewritten as

$$
\min_{P,Q,R,L} \| I - P \cdot Q \|_F^2 + \alpha \Phi(R) + \beta \Psi(L) + \gamma \| P - R \|_F^2 + \lambda \| Q - L \|_F^2,
$$

where $\gamma$ and $\lambda$ are penalty parameters.

To solve the problem in Eq. (4), the value of one variable is alternatively updated with those of the others fixed. Therefore, we partition the problem into four univariate subproblems, which can be optimized by the following alternating scheme:

$$
P_k = \arg \min_P \| I - P \cdot Q_{k-1} \|_F^2 + \gamma \| P - R_{k-1} \|_F^2, \quad \tag{5}
$$

$$
R_k = \arg \min_R \alpha \Phi(R) + \gamma \| P_k - R \|_F^2, \quad \tag{6}
$$

$$
Q_k = \arg \min_Q \| I - P_k \cdot Q \|_F^2 + \lambda \| Q - L_{k-1} \|_F^2, \quad \tag{7}
$$

$$
L_k = \arg \min_L \beta \Psi(L) + \lambda \| Q_k - L \|_F^2, \quad \tag{8}
$$

where $k$ denotes the iteration index.

### 3.2. URetinex-Net Framework

Since it is difficult to design specific regularization terms $\Phi(R)$ and $\Psi(L)$, we take advantage of deep networks to adaptively fit physical priors of $R$ and $L$. Therefore, based on the above-mentioned optimization scheme, we map the update steps to a deep unfolding network architecture, and propose a novel framework for LLIE. As shown in Fig. 2, the proposed URetinex-Net includes three modules, i.e., initialization module, unfolding optimization module, and illumination adjustment module.

#### 3.2.1 Initialization Module

Initialization plays an important role during optimization. Random or all-zero initializations are widely used in the conventional optimization scheme, such as ADMM [1]. Considering that a reliable initialization is beneficial for performance, we hope to obtain an initialized illumination and reflectance with richer information instead of random values or all zeros.

Intuitively, to preserve the overall structure of $I$, initial illumination $L_0$ can be initialized by seeking the maximum value of three color channels [12], and initial reflectance $R_0$ can be accordingly derived by $R_0 = \frac{I}{L_0}$, where $\frac{1}{L_0}$ denotes element-wise division. However, initialization in such a rigid manner will cause color distortion. As illustrated in Fig. 3(b), statistical characteristics of intensity for three channels (e.g., $\{R, G, B\}$) are changed.

Therefore, in order to reveal coarse details but avoid raising distortion, we propose a data-dependent initialization module named $D$, which uses a fully convolutional (Conv) network to adaptively and simultaneously learn $R_0$ and $L_0$. The initialization module consists of three Conv+LeakyReLU layers, followed by a convolutional layer and ReLU layer. Kernel size is set to be $3 \times 3$ across the whole convolutional layers. For initializing two components of low-light images, the loss function is designed as follows:

$$
\min_{R_0, L_0} \| I - R_0 \cdot L_0 \|_1 + \mu \| L_0 - \max_{c \in \{R, G, B\}} I^{(c)} \|_F^2, \quad \tag{9}
$$

where $\| \cdot \|_1$ denotes $l_1$ norm, $\mu$ is the hyper-parameter, and $c \in \{R, G, B\}$ denotes the RGB channels. The first term is the reconstruction loss, and the second term aims to encourage the initialized illumination to preserve the overall structure of $I$.

Besides, due to the absence of ground-truth reflectance, normal-light images are utilized to generate clear reflectance, which should be close to that of low-light images. Therefore, the reflectance of the normal-light image is used as a reference in the following unfolding optimization module. Based on the network architecture of the initialization module, we further impose structure-aware smooth constraints.
constraint [34] on the illumination of the normal-light image, and then loss function for decomposing a normal-light image is as follows:

$$\min_{R,L} \| \hat{I} - \hat{R} \hat{L} \|_1 + \mu (\| \hat{L} - \max_{c \in \{R,G,B\}} \hat{I}(c)^2 + \| c^{-\gamma} \hat{I} \|_1), \quad (10)$$

where $\hat{I}$, $\hat{R}$, and $\hat{L}$ represent the normal-light image, reflectance of $\hat{I}$ and illumination of $\hat{I}$, respectively. $\gamma$ and $\mu$ are hyper-parameters, and $\nabla(\cdot)$ denotes gradient operation.

In the third term, the total variation of the illumination map is weighted by the gradient of the image, such that illumination can be spatially smoothed in a structure-aware manner.

### 3.2.2 Unfolding Optimization Module

The unfolding optimization module aims to iteratively solve four univariate subproblems to update corresponding variables within $T$ iterations. Through mapping the updating steps to a deep neural network architecture, the inference is unfolded into $T$ stages, each of which corresponds to one iteration where $P$, $Q$, $L$, and $R$ are updated in an alternative manner. In the following, we sequentially present updating rules in the proposed module.

**Updating rules for $P$ and $Q$:** Apparently, P-subproblem in Eq. (5) is a classic least square problem, whose close-form solution can be obtained by differentiating Eq. (5) with respect to $P$ and setting the derivative to 0. Hence, given the initialized reflectance and the closed-form solution to P-subproblem, the updating formula concerning $P$ is as follows:

$$P_k = F_P(I, R_{k-1}, Q_{k-1}, \gamma)$$

$$= \begin{cases} R_0, & k = 1, \\ \gamma R_{k-1} + \hat{I} \cdot Q_{k-1} - Q_{k-1} \cdot I + \gamma I, & \text{else}, \end{cases} \quad (11)$$

where $I$ denotes all-ones matrix.

Similarly, updating $Q$ can be done via solving $Q$-subproblem in Eq. (7). As restoring low-light images in RGB space, reflectance layers in three channels share the same illumination layer, such that illumination is assumed to be grayscale. Therefore, Eq. (7) is rewritten as

$$Q_k = \arg \min_Q \sum_{c \in \{R,G,B\}} \| I(c) - P_k(c) Q_k \|_F^2 + \lambda \| Q - L_{k-1} \|_F^2,\quad (12)$$

whose closed-form solution can be found easily. Considering the initialized illumination, the updating formula for $Q$ is obtained as

$$Q_k = F_Q(I, L_{k-1}, P_k, \lambda)$$

$$= \begin{cases} L_0, & k = 1, \\ \lambda L_{k-1} + \sum_{c \in \{R,G,B\}} \hat{I}(c) \cdot P_k(c) - Q_k \cdot I + \lambda I, & \text{else}. \end{cases} \quad (13)$$

### Updating rules for $L$ and $R$:** For L- and R-subproblems in Eqs. (8) and (6), instead of introducing hand-crafted priors to manually design specific loss functions, we develop learning-based methods to explore implicit priors from real-world data. In other words, two networks denoted as $G_L$ and $G_R$ are introduced to conduct updating for $L$ and $R$, respectively.

Specifically, the network which is employed to fit the physical prior over $L$ is expressed as

$$L_k = G_L(Q_k; \theta_L), \quad (14)$$

where $Q_k$ is taken as the input of $G_L$, and $\theta_L$ denotes learnable parameters. We adopt a simple fully convolutional network with five Conv layers followed by ReLU activation to learn implicit priors over $L$, thereby prior can be learned from training data while avoiding to design a complicated regularization term.

Then, by passing the degraded $P_k$ through a learnable denoising network $G_R$ in a similar way, a cleaner reflectance can be obtained. However, the level of distortion that appears in reflectance is highly correlated to the luminance of an illumination layer, where darker regions accompany by heavier degradation. Therefore, $Q_k$ is aggregated with $P_k$ as input fed to $G_R$ for the purpose of guiding reflectance restoration. Thus, the network which used for performing updating of $R$ is expressed as

$$R_k = G_R(P_k, Q_k; \theta_R), \quad (15)$$

where $\theta_R$ denotes learnable parameters in $G_R$. In order to fuse $P_k$ and $Q_k$ to update $R_k$, the squeeze-and-excitation (SE) [14] block is employed. Details of $G_R$ are demonstrated in Fig. 2(c).

The unfolding optimization module is trained in an end-to-end manner, where parameters and network architectures of $G_L$ and $G_R$ are shared across different stages. Normal-light reflectance $\hat{R}$ generated by our initialization module is used as the reference during the optimization of unfolding networks. With regarding to the loss function, we adopt the summation of loss functions for reflectance and illumination, which includes the mean squared error (MSE) loss between $P_k$ and $R_k$ in each stage, the MSE loss, the structural similarity loss, and the perceptual loss between $R$ and the final restored reflectance $R_T$, the MSE loss between $Q_k$ and $L_k$ in each stage, and the total variation loss for $L_k$ in each stage. The loss function for the unfolding optimization module is as follows:

$$\min_{R_T, L_T} \sum_{k=1}^T (\gamma_k \| P_k - R_k \|_F^2 + \lambda_k \| Q_k - L_k \|_F^2)$$

$$+ \beta \| \nabla L_T \|_1 + \alpha (\| \hat{R}(P_k) - \phi(R_T) \|_1)$$

$$+ \| \hat{R} - R_T \|_F^2 + (1 - \text{SSIM}(\hat{R}, R_T))), \quad (16)$$

5905
where $T$ denotes the total number of stages, $\phi(\cdot)$ denotes the high-level feature extractor which is pre-trained on ImageNet by VGG19 network, SSIM(\cdot) represents the structural similarity loss, and $\gamma_k$, $\lambda_k$, $\alpha$, and $\beta$ are trade-off parameters, respectively.

Apparently, even in the deep neural network architecture, the proposed unfolding optimization module has good interpretability, where $\mathcal{F}_P$, $\mathcal{F}_Q$, $\mathcal{G}_L$, and $\mathcal{G}_R$ all have clear meanings. What’s more, it avoids explicit regularization designing and adaptively restores illumination and reflectance in the deep-learning manner.

### 3.2.3 Illumination Adjustment Module

In practice, there is no ground-truth light level for image enhancement, and flexibly adjusting illumination is necessary for fitting different practical requirements. For LLIE, Gamma correction is widely used to brighten up the illumination map [9, 22, 30], i.e., $\tilde{L} = L^\gamma$, where $\tilde{L}$ represents adjusted map and the changeable factor $\gamma$ is empirically set to 1/2.2. However, as suggested in [41], compared with gamma correction, illumination adjustment in the learning manner is more corroborative with actual situations. To this end, we propose an illumination adjustment module, which takes low-light illumination $L$ and user-specific enhancement ratio $\omega$ as input, expressed as

$$\tilde{L} = A(L, \omega; \theta_A),$$

(17)

where $\theta_A$ denotes learnable parameters of $A$. Note that $\omega > 1$ denotes the brighten-up case.

As the distribution of the illumination map is non-uniform, we expand $\omega$ to a map, whose size is the same as that of $L$. Thus, the adjusted illumination $\tilde{L}$ is then obtained by passing the concatenation of $\omega$ and $L$ through $A$. In the concern of computational efficiency, we adopt the same lightweight network structure as the initialization module. To maintain the consistency and adjust illumination smoothly, the convolutional kernel size in this module is enlarged to $5 \times 5$.

The loss function for illumination adjustment module is as follows:

$$\min_{\tilde{L}} \| \nabla \tilde{L} - \nabla L \|_1 + \| R : \tilde{L} - \hat{I} \|_F^2 + (1 - \text{SSIM}(R : \tilde{L}, \hat{I})),$$

(18)

where $\| \nabla \tilde{L} - \nabla L \|_1$ aims to enforce the refined illumination map $\tilde{L}$ to maintain the consistency with $L$, and the other two terms are both reconstruction losses to guarantee the fidelity constraints for learning $\tilde{L}$. During training, the value of $\omega$ is defined by $\frac{\text{mean}(\tilde{L})}{\text{mean}(L)}$. While for inference, the enhancement ratio $\omega$ is specified by user.

Although KinD [41] also develops a flexible illumination adjustment net, it depends on the decomposed illumination map of the normal-light image, which may destroy the local structure of the input $L$. Instead, the proposed module aims to maintain structural consistency with low-light illumination. Moreover, the last two terms in loss function integrate reflectance component, which aims to constrain the refined illumination can naturally reconstruct the normal-light images.

### 4. Experiments

#### 4.1. Implementation Details and Data sets

To evaluate the performance of our proposed method, we train and test our model on LOL dataset [34], which contains 500 low/normal-light image pairs, and is captured at various exposure times from the real world. We follow the setting of training data as [41]. For a more convincing comparison, we extend our model to SICE dataset [2], which contains 589 natural scenes with multi-level exposure, and randomly select 108 under/normal exposure image pairs from it. Furthermore, we adopt MEF dataset [19] for visual comparison to demonstrate the efficiency of our proposed method.

URetinex-Net is trained separately for each module, where the batch size is set to be 4. We use a small patch size $(48 \times 48)$ for training our unfolding module in the concern of efficiency. Through the ablation experiment, we find that $T = 3$ has already achieved promising results, such that $T$ is empirically set to 3 in the following experiments. Each module is trained using Adam optimizer with a learning rate of 0.0001 and decaying by 10 after 30K iterations. Heuristically, $\mu$, $\mu_e$, $\alpha$, $\beta$ and $\epsilon$ are set to 0.1, 0.1, 1, 20 and 10 respectively. According to [36], penalty parameters $\lambda$ and $\gamma$ are expected to iteratively increase for stable convergence. Here, we initially set $\lambda$ and $\gamma$ as 0.5 and 0.1 respectively, both of which are increased by 0.05 in each stage. All experiments are conducted on an NVIDIA Tesla V100 GPU under PyTorch [28] framework.

#### 4.2. Comparison with the state-of-the-art

We qualitatively and quantitatively compare with five traditional Retinex-based methods, including NPE [32], SRIE [9], LIME [12], RRIM [22], and LR3M [30]. Furthermore, to verify the efficiency of our model in learning implicit prior from data, we compare with state-of-the-art (SOTA) learning-based methods including RetinexNet [34], KinD [41], Zero-DCE [11], RUAS [24], AGLL-Net [26] and KinD++ [40]. Retinex-Net, KinD, KinD++ and RUAS have been trained on LOL dataset, such that we utilize their provided models for evaluation. Otherwise, for Zero-DCE, we reorganize the training data with the cropped images in LOL training dataset and fine-tuned it via the expanded dataset following their default settings. To measure the differences in color, structure, and high-level feature...
Net and Zero-DCE introduce and even amplify noise after retraining or fine-tuning. Results are reported in Table 1. Quantitative comparison on LOL and SICE datasets. The best and the second best results are boldfaced and underlined.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Method</th>
<th>MAE</th>
<th>PSNR</th>
<th>SSIM</th>
<th>LPIPS</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOL</td>
<td>NPE [32]</td>
<td>0.1290</td>
<td>16.9697</td>
<td>0.4818</td>
<td>2.00607</td>
</tr>
<tr>
<td></td>
<td>SRIE [39]</td>
<td>0.2058</td>
<td>14.0580</td>
<td>0.6420</td>
<td>1.2492</td>
</tr>
<tr>
<td></td>
<td>LIME [12]</td>
<td>0.1472</td>
<td>15.4572</td>
<td>0.7001</td>
<td>1.3935</td>
</tr>
<tr>
<td></td>
<td>RRM [22]</td>
<td>0.1601</td>
<td>15.9489</td>
<td>0.6892</td>
<td>1.5084</td>
</tr>
<tr>
<td></td>
<td>LR3M [30]</td>
<td>0.1940</td>
<td>14.2086</td>
<td>0.6073</td>
<td>1.9482</td>
</tr>
<tr>
<td></td>
<td>Retinex-Net  [34]</td>
<td>0.1382</td>
<td>15.9080</td>
<td>0.7052</td>
<td>1.5377</td>
</tr>
<tr>
<td></td>
<td>KinD++ [40]</td>
<td>0.1151</td>
<td>17.7330</td>
<td>0.7588</td>
<td>1.6684</td>
</tr>
<tr>
<td></td>
<td>Zero-DCE [11]</td>
<td>0.1330</td>
<td>16.9090</td>
<td>0.7554</td>
<td>1.2613</td>
</tr>
<tr>
<td></td>
<td>RUAS [24]</td>
<td>0.2628</td>
<td>10.9878</td>
<td>0.5451</td>
<td>2.5565</td>
</tr>
<tr>
<td></td>
<td>AGLLNet [26]</td>
<td>0.1463</td>
<td>16.0006</td>
<td>0.7336</td>
<td>1.5196</td>
</tr>
<tr>
<td></td>
<td>URetinex-Net</td>
<td>0.1068</td>
<td>18.9467</td>
<td>0.7808</td>
<td>1.2744</td>
</tr>
<tr>
<td>SICE</td>
<td>NPE [32]</td>
<td>0.1486</td>
<td>16.1973</td>
<td>0.7177</td>
<td>1.2848</td>
</tr>
<tr>
<td></td>
<td>SRIE [39]</td>
<td>0.2058</td>
<td>14.0580</td>
<td>0.6420</td>
<td>1.2492</td>
</tr>
<tr>
<td></td>
<td>LIME [12]</td>
<td>0.1472</td>
<td>15.4572</td>
<td>0.7001</td>
<td>1.3935</td>
</tr>
<tr>
<td></td>
<td>RRM [22]</td>
<td>0.1601</td>
<td>15.9489</td>
<td>0.6892</td>
<td>1.5084</td>
</tr>
<tr>
<td></td>
<td>LR3M [30]</td>
<td>0.1940</td>
<td>14.2086</td>
<td>0.6073</td>
<td>1.9482</td>
</tr>
<tr>
<td></td>
<td>Retinex-Net  [34]</td>
<td>0.1382</td>
<td>15.9080</td>
<td>0.7052</td>
<td>1.5377</td>
</tr>
<tr>
<td></td>
<td>KinD++ [40]</td>
<td>0.1151</td>
<td>17.7330</td>
<td>0.7588</td>
<td>1.6684</td>
</tr>
<tr>
<td></td>
<td>Zero-DCE [11]</td>
<td>0.1330</td>
<td>16.9090</td>
<td>0.7554</td>
<td>1.2613</td>
</tr>
<tr>
<td></td>
<td>RUAS [24]</td>
<td>0.2628</td>
<td>10.9878</td>
<td>0.5451</td>
<td>2.5565</td>
</tr>
<tr>
<td></td>
<td>AGLLNet [26]</td>
<td>0.1463</td>
<td>16.0006</td>
<td>0.7336</td>
<td>1.5196</td>
</tr>
<tr>
<td></td>
<td>URetinex-Net</td>
<td>0.1068</td>
<td>18.9467</td>
<td>0.7808</td>
<td>1.2744</td>
</tr>
</tbody>
</table>

Moreover, we give an extensive visual comparison on MEF dataset. We report the top-5 best methods in terms of performance and speed in Fig. 5. Although traditional model-based methods can somehow restore low-light images, they are time-consuming in the iterative optimization procedure. In contrast, our method saves more time during inference. Furthermore, Zero-DCE as a learning-based LLIE method has a fast processing speed, but it has a limited capacity to achieve noise suppression and reach a satisfactory effect. KinD++ and RUAS remove noise in a post-processing manner which may bring other problems such as losing details, blurring or even worse image quality. In comparison to all of these approaches, our model is capable of noise suppression and detail preservation while sufficiently revealing low-light images. Demonstrating that, compared to meticulously hand-crafted planned priors, our unfolding module can impose a more robust implicit prior. In particular, our model shows the unique advantage of details recovering, which illustrates the superiority of our unfolding optimization. More comparison results and analysis can be found in the supplementary materials.

4.3 Ablation Study

The results of the ablation study are reported in Table 2. We first analyze the effectiveness of our initialization module by comparing with a rigid initialization manner (e.g., \( L_0 = \max_{c \in \{R, G, B\}} I^{(c)} \)). Then, to investigate the effectiveness of fusing illumination layer for the guidance of learning regularization term on reflectance, we remove \( Q_k \) from the input of network \( \hat{u}_R \), while the rest of the setups are as same as URetinex-Net. In order to illustrate the effect-

Table 1. Quantitative comparison on LOL and SICE datasets. The best and the second best results are boldfaced and underlined.
Table 2. Quantitative results of ablation study in terms of PSNR, SSIM, LPIPS and inference time on LOL dataset, where IM, UOM and IG are abbreviations for the proposed initialization module, unfolding optimization module and illumination guiding, respectively. Noted that time is represented in millisecond (ms).

<table>
<thead>
<tr>
<th>Method</th>
<th>IM</th>
<th>UOM</th>
<th>IG</th>
<th>T</th>
<th>PSNR↑</th>
<th>SSIM↑</th>
<th>LPIPS↓</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>w/o IM</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>3</td>
<td>20.0992</td>
<td>0.7304</td>
<td>1.9847</td>
<td>34.6</td>
</tr>
<tr>
<td>w/o UOM</td>
<td>✓</td>
<td>✗</td>
<td>✓</td>
<td>5</td>
<td>19.6109</td>
<td>0.8173</td>
<td>1.3264</td>
<td>35.8</td>
</tr>
<tr>
<td>w/o IG</td>
<td>✓</td>
<td>✓</td>
<td>✗</td>
<td>1</td>
<td>21.1222</td>
<td>0.8273</td>
<td>1.3165</td>
<td><strong>14.4</strong></td>
</tr>
<tr>
<td>Ours*</td>
<td>✓</td>
<td>✗</td>
<td>✓</td>
<td>3</td>
<td>21.3282</td>
<td>0.8348</td>
<td>1.2234</td>
<td>36.7</td>
</tr>
<tr>
<td>Ours</td>
<td>✓</td>
<td>✓</td>
<td>✗</td>
<td>5</td>
<td>21.5767</td>
<td>0.8334</td>
<td>1.2412</td>
<td>58.7</td>
</tr>
<tr>
<td>Ours*</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>7</td>
<td>21.4345</td>
<td>0.8332</td>
<td>1.2332</td>
<td>80.8</td>
</tr>
</tbody>
</table>

This work was supported in part by the National Natural Science Foundation of China (Grants 61871270 and 62006158), and in part by the Shenzhen Natural Science Foundation under Grants JCYJ20200109110410133, 20200812110350001 and 20200810150732001.
References


