LIME: Low-light Image Enhancement via Illumination Map Estimation

Xiaojie Guo, *Member, IEEE,* Yu Li, *Member, IEEE*, and Haibin Ling, *Member, IEEE*

Abstract—When one captures images in low-light conditions, the images often suffer from low visibility. Besides degrading the visual aesthetics of images, this poor quality may also significantly degenerate the performance of many computer vision and multimedia algorithms that are primarily designed for high-quality inputs. In this paper, we propose a simple yet effective low-light image enhancement (LIME) method. More concretely, the illumination of each pixel is first estimated individually by finding the maximum value in R, G and B channels. Further, we refine the initial illumination map by imposing a structure prior on it, as the final illumination map. Having the well-constructed illumination map, the enhancement can be achieved accordingly. Experiments on a number of challenging low-light images are present to reveal the efficacy of our LIME and show its superiority over several state-of-the-arts in terms of enhancement quality and efficiency.

Index Terms—Illumination Estimation, Illumination (Light) Transmission, Low-light Image Enhancement

I. INTRODUCTION

U NDOUBTEDLY, high-visibility images reflect clear details of target scenes, which are critical to many vision-based techniques, such as object detection [1] and tracking [2]. But, images captured in low-light conditions are often of low visibility. The visual quality of images captured under low-light conditions, for one thing, is barely satisfactory. For another thing, it very likely hurts the performance of algorithms that are primarily designed for high-visibility inputs. Figure 1 provides several such examples, from which, we can see that many details, such as the paintings on the wall in the first case, the distant field on the bottom-left corner in the third case and the reflection on the floor in the last one, have almost been "buried" in the dark. To make the buried information visible, low-light image enhancement is definitely demanded.

Directly amplifying the low-light image is probably the most intuitive and simplest way to recall the visibility of dark regions. But, this operation gives birth to another problem, say relatively bright regions might be saturated and thus loss corresponding details. Histogram equalization (HE) strategies

X. Guo is with State Key Laboratory Of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China (e-mail: xj.max.guo@gmail.com).

Y. Li is with the Advanced Digital Sciences Center, Singapore (e-mail: li.yu@adsc.com.sg).

H. Ling is with Department of Computer and Information Sciences, Temple University, PhiladelphiaPA 19122, USA (e-mail: hbling@temple.edu).

[3], [4], [5] can avoid the above problem by somehow forcing the output image to fall in the range $[0, 1]$. Further, variational methods aim to improve the HE performance by imposing different regularization terms on the histogram. For instance, contextual and variational contrast enhancement (CVC) [6] tries to find a histogram mapping that pays attention on large gray-level differences, while the work [7] achieves improvement by seeking a layered difference representation of 2D histograms (LDR). However, in nature, they focus on contrast enhancement instead of exploiting real illumination causes, having the risk of over- and under-enhancement. Another solution is Gamma correction that is a nonlinear operation on images. The main drawback is that the nonlinear operation of Gamma correction is carried out on each pixel individually without considering the relationship of a certain pixel with its neighbors, and thus may make enhanced results vulnerable and visually inconsistent with real scenes.

In Retinex theory [8], the dominant assumption is that the (color) image can be decomposed into two factors, say reflectance and illumination. Early attempts based on Retinex, such as single-scale Retinex (SSR) [9] and multi-scale Retinex (MSR) [10], treat the reflectance as the final enhanced result, which often looks unnatural and frequently appears to be over-enhanced. The method proposed in [11] tries to enhance contrast while preserving naturalness of illumination. Although it prevents results from over-enhancement, in our experiments, it performs less impressive than our method in terms of both efficiency and visual quality. Fu *et al.* proposed a method to adjust the illumination by fusing multiple derivations of the initially estimated illumination map (MF) [12]. The performance of MF is mostly promising. But, due to the blindness of illumination structure, MF may lose the realism of regions with rich textures. The most recent work of [13] proposed a weighted variational model for simultaneous reflectance and illumination estimation (SRIE). With the estimated reflectance and illumination, the target image can be enhanced by manipulating the illumination. As noticed in [14], inverted low-light images look like haze images, as shown in Fig. 2. Based on this observation, the authors of [14] alternatively resorted to dehaze the inverted low-light images. After dehazing, the obtained unrealistic images are inverted again as the final enhanced results. Recently, Li *et al.* followed this technical line and further improved the visual quality by first over-segmenting the input image and then adaptively denoising different segments [15]. Even though the above dehazing-like methods can provide reasonable results, the basic model they rely on is lacking in physical explanation. By contrast, our method has clear physical intuition.
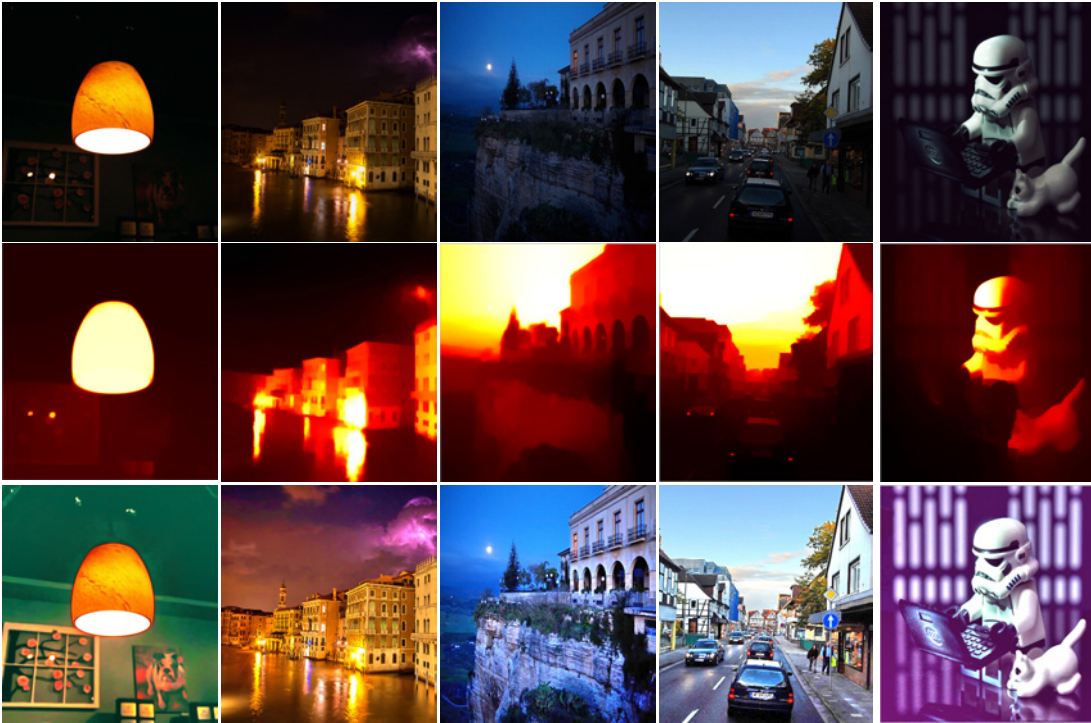
Fig. 1: **Top Row:** Natural low-light images. **Middle Row:** The illumination maps estimated by our method. **Bottom Row:** The results enhanced by our method.
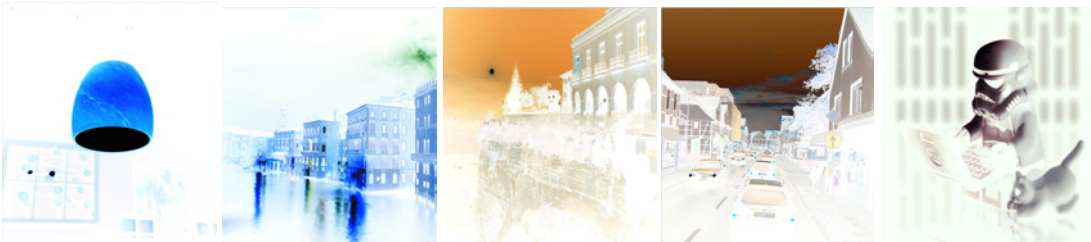

Fig. 2: The inverted versions (unrealistic images) of images shown in the top row of Fig. 1.

Contribution Our method belongs to the Retinex-based category, which intends to enhance a low-light image by estimating its illumination map. It is worth noting that, different from the traditional Retinex-based methods like [13] that decompose an image into the reflectance and the illumination components, our method only estimates one factor, say the illumination, which shrinks the solution space and reduces the computational cost to reach the desired result. The illumination map is first constructed by finding the maximum intensity of each pixel in R, G and B channels. Then, we exploit the structure of the illumination to refine the illumination map. An Augmented Lagrangian Multiplier (ALM) based algorithm is given to exactly solve the refinement problem, while another sped-up solver is designed to intensively reduce the computational load. Experiments on a number of challenging images are conducted to reveal the advantages of our method in comparison with other state-of-the-art methods.

II. METHODOLOGY

Our method is built upon the following (Retinex) model, which explains the formation of a low-light image:

$$\mathbf{L} = \mathbf{R} \circ \mathbf{T}, \tag{1}$$

where $\mathbf{L}$ and $\mathbf{R}$ are the captured image and the desired recovery, respectively. Furthermore, $\mathbf{T}$ represents the illumination map, and the operator $\circ$ means element-wise multiplication. In this paper, we assume that, for color images, three channels share the same illumination map. With slight abuse of notations, we use $\mathbf{T}$ ($\hat{\mathbf{T}}$) to represent one-channel and three-channel illumination maps interchangeably. The model (1) is with clear physical meaning, say the observed image can be decomposed into the product of the desired light-enhanced scene and the illumination map.

The model of our problem is similar with that of the intrinsic image decomposition [16], [17], [18], [19], which attempts
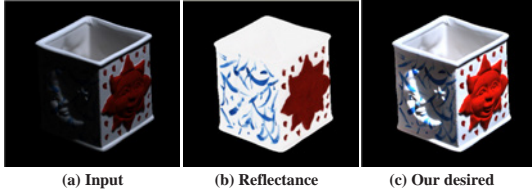
(a) Input   (b) Reflectance   (c) Our desired

Fig. 3: Different purposes of intrinsic image decomposition and low-light image enhancement.

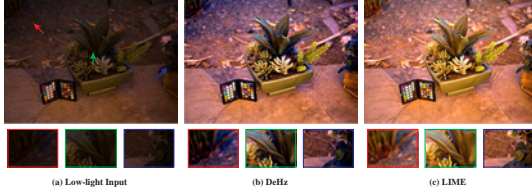(a) Low-light Input   (b) DeHz   (c) LIME

Fig. 4: Comparison of (6) and (3) with the same illumination map. The atmospheric light a estimated by [20] is larger than 0.95. Even though, the difference is still noticeable.

to decompose the input into two components[1]. However, the goal of the intrinsic image decomposition is to recover the reflectance component and the shading one from the given image. As shown in Fig. 3 (b), the reflectance loses the shape of the box (the ground truth reflectance is from [16]), which does not satisfy the purpose of low-light image enhancement. The expectation of our work is to recall the visual content of dark regions as well as keep the visual realism, as shown in Fig. 3 (c). Some researchers noticed the unrealism of using the reflectance as the enhanced result, for example [9], [10], and tried to project the modified illumination back to the reflectance [13] by $\hat{\mathbf{R}} \circ f(\hat{\mathbf{T}})$, where $\hat{\mathbf{R}}$ and $\hat{\mathbf{T}}$ are the recovered reflectance and illumination respectively, and $f(\cdot)$ stands for a manipulation operator such as Gamma correction. We can see that the desired result of enhancement is obtained by somehow combining the decomposed components again. Further, due to the ill-posedness of the decomposition problem, more priors are required to help constrain the space of solution. But if the task is just to lighten low-light images, which is this paper concentrates on, it is not necessary to decompose the input image into two components. Because, by slightly transforming (1), we have $\mathbf{R} = \mathbf{L}/\mathbf{T}$, where the division is element-wise. It is apparent that the estimation of $\mathbf{T}$ is key to the recovery of $\mathbf{R}$. In this way, the problem is simplified, only demanding the estimation of $\mathbf{T}$. Please notice that $\mathbf{L}/\hat{\mathbf{T}}$ can directly act as the light-enhanced result.

A. Illumination Map Estimation

As one of the first color constancy methods, Max-RGB [8] tries to estimate the illumination by seeking the maximum value of three color channels, say R, G and B. But this

estimation can only boost the global illumination. In this paper, to handle non-uniform illuminations, we alternatively adopt the following initial estimation:

$$\hat{\mathbf{T}}(x) \leftarrow \max_{c \in \{R,G,B\}} \mathbf{L}^c(x), \qquad (2)$$

for each individual pixel x. The principle behind the above operation is that the illumination is at least the maximal value of three channels at a certain location. The obtained $\hat{\mathbf{T}}(x)$ guarantees that the recovery will not be saturated, because of

$$\mathbf{R}(x) = \mathbf{L}(x)/(\max_c \mathbf{L}^c(x) + \epsilon), \qquad (3)$$

where ϵ is a very small constant to avoid the zero denominator. We point out that the goal of this work is to non-uniformly enhance the illumination of low-light images, instead of eliminating the color shift caused by light sources.

As mentioned, another widely used model is based on the observation that inverted low-light images $1 - \mathbf{L}$ look similar to haze images, which is thus expressed as [20], [21], [22]:

$$1 - \mathbf{L} = (1 - \mathbf{R}) \circ \tilde{\mathbf{T}} + a(1 - \tilde{\mathbf{T}}), \qquad (4)$$

where a represents the global atmospheric light. Although the visual effect of inverted low-light images $1 - \mathbf{L}$ is intuitively similar to haze images, compared to the model (1), the physical meaning of the above remains vague. Below we intend to show the relation between (4) and (1).

Let us here recall the dark channel prior, a commonly used prior to estimate the transmission map for dehazing [20], on $1 - \mathbf{L}$ as follows:

$$\tilde{\mathbf{T}}(x) \leftarrow 1 - \min_c \frac{1 - \mathbf{L}^c(x)}{a} = 1 - \frac{1}{a} + \max_c \frac{\mathbf{L}^c(x)}{a}. \qquad (5)$$

Accordingly, substituting (5) into (4) yields:

$$\mathbf{R}(x) = \frac{\mathbf{L}(x) - 1 + a}{(1 - \frac{1}{a} + \max_c \frac{\mathbf{L}^c(x)}{a} + \epsilon)} + (1 - a). \qquad (6)$$

We can see that when $a = 1$, both (3) and (6) reach the same result. But, if a gets away from 1, the equivalence between the model (6) [14] and (3) breaks. As can be seen from Fig. 4, even though the atmospheric light is greater than 0.95, the visual difference between using (6) and using (3) is still conspicuous. The dark regions in Fig. 4 (b) are less enhanced than those in Fig. 4 (c), please see the zoomed-in patches for details. In this work, we rely on the model (3) without involving the atmospheric light a.

In this work, we employ (2) to initially estimate illumination map $\hat{\mathbf{T}}$, due to its simplicity, although various approaches (*e.g.* [23], [24], [25]) have been developed to improve the accuracy in past decades. Most of these improvements essentially consider the local consistency of illumination by taking into account neighboring pixels within a small region around the target pixel. Two representative ways are:

$$\hat{\mathbf{T}}(x) \leftarrow \max_{y \in \Omega(x)} \max_{c \in \{R,G,B\}} \mathbf{L}^c(y);$$
$$\hat{\mathbf{T}}(x) \leftarrow \operatorname*{mean}_{y \in \Omega(x)} \max_{c \in \{R,G,B\}} \mathbf{L}^c(y), \qquad (7)$$

where $\Omega(x)$ is a region centered at pixel x, and y is the location index within the region. These schemes can somewhat

---

[1]The intrinsic image decomposition originally involves three factors including Lambertian shading ($\mathbf{T}$), reflectance ($\mathbf{R}$) and specularities ($\mathbf{C}$), formally expressed as $\mathbf{L} = \mathbf{R} \circ \mathbf{T} + \mathbf{C}$. As pointed out in [16], the simplified model with the component $\mathbf{C}$ discarded can also work well. And many works, such as [17], [18] and [19], are based on this simplified model.

enhance the local consistency, but they are structure-blind. In the following, we provide a more powerful scheme to better achieve this goal.

A "good" solution should simultaneously preserve the overall structure and smooth the textural details. To address this issue, based on the initial illumination map $\hat{\mathbf{T}}$, we propose to solve the following optimization problem:

$$\min_{\mathbf{T}} \|\hat{\mathbf{T}} - \mathbf{T}\|_F^2 + \alpha\|\mathbf{W} \circ \nabla\mathbf{T}\|_1, \qquad (8)$$

where α is the coefficient to balance the involved two terms and, $\|\cdot\|_F$ and $\|\cdot\|_1$ designate the Frobenious and ℓ_1 norms, respectively. Further, $\mathbf{W}$ is the weight matrix, and $\nabla\mathbf{T}$ is the first order derivative filter. In this work, it only contains $\nabla_h\mathbf{T}$ (horizontal) and $\nabla_v\mathbf{T}$ (vertical). In the objective (8), the first term takes care of the fidelity between the initial map $\hat{\mathbf{T}}$ and the refined one $\mathbf{T}$, while the second term considers the (structure-aware) smoothness. Prior to discussing possible strategies of constructing $\mathbf{W}$, we give two solvers in the next two sub-sections to resolve problem (8).

B. Exact Solver to Problem (8)

Traditionally, the problem (8) can be effectively solved via the alternating direction minimization technique. As can be seen from the objective in (8), both two terms, say ℓ_2 and ℓ_1 terms, involve $\mathbf{T}$. An auxiliary variable $\mathbf{G}$ is introduced to replace $\nabla\mathbf{T}$ for making the problem separable and thus easy to solve. Accordingly, $\nabla\mathbf{T} = \mathbf{G}$ is added as a constraint. As a result, we have the following equivalent optimization problem:

$$\min_{\mathbf{T},\mathbf{G}} \|\hat{\mathbf{T}} - \mathbf{T}\|_F^2 + \alpha\|\mathbf{W} \circ \mathbf{G}\|_1 \quad \text{s.t.} \quad \nabla\mathbf{T} = \mathbf{G}. \qquad (9)$$

The augmented Lagrangian function of (9) can be naturally written in the following shape:

$$\mathcal{L} = \|\hat{\mathbf{T}} - \mathbf{T}\|_F^2 + \alpha\|\mathbf{W} \circ \mathbf{G}\|_1 + \Phi(\mathbf{Z}, \nabla\mathbf{T} - \mathbf{G}), \quad (10)$$

with the definition $\Phi(\mathbf{Z}, \nabla\mathbf{T} - \mathbf{G}) = \frac{\mu}{2}\|\nabla\mathbf{T} - \mathbf{G}\|_F^2 + \langle\mathbf{Z}, \nabla\mathbf{T} - \mathbf{G}\rangle$, where $\langle\cdot,\cdot\rangle$ represents matrix inner product, μ is a positive penalty scalar, and $\mathbf{Z}$ is the Lagrangian multiplier. There are three variables, including $\mathbf{T}$, $\mathbf{G}$ and $\mathbf{Z}$ to solve. The ALM technique is a common choice to solve the problem (8). The solver iteratively updates one variable at a time by fixing the others, and each step has a simple closed-form solution. For conveniently analyzing and comparing the exact solver and the sped-up one (proposed later), we provide the solutions of the sub-problems below:

T sub-problem: Collecting the $\mathbf{T}$ involved terms from Eq. (10) gives the problem as follows:

$$\mathbf{T}^{(t+1)} \leftarrow \operatorname*{argmin}_{\mathbf{T}} \|\hat{\mathbf{T}} - \mathbf{T}\|_F^2 + \Phi(\mathbf{Z}^{(t)}, \nabla\mathbf{T} - \mathbf{G}^{(t)}). \quad (11)$$

As can be seen from Eq. (11), it is a classic least squares problem. Thus, the solution can be computed through differentiating (11) with respect to $\mathbf{T}$ and setting it to 0:

$$2(\mathbf{T} - \hat{\mathbf{T}}) + \mu^{(t)}\mathbf{D}^T(\mathbf{D}\mathbf{T} - \mathbf{G}^{(t)}) + \mathbf{D}^T\mathbf{Z}^{(t)} = 0$$
$$\Rightarrow (2\mathbf{I} + \mu^{(t)}\mathbf{D}^T\mathbf{D})\mathbf{T} = 2\hat{\mathbf{T}} + \mu^{(t)}\mathbf{D}^T(\mathbf{G}^{(t)} - \frac{\mathbf{Z}^{(t)}}{\mu^{(t)}}), \quad (12)$$

where $\mathbf{I}$ is the identity matrix with proper size. And $\mathbf{D}$ contains $\mathbf{D}_h$ and $\mathbf{D}_v$, which are the Toeplitz matrices from the discrete gradient operators with forward difference. We note that, for convenience, the operations $\mathbf{D}\mathbf{X}$ and $\mathbf{D}^T\mathbf{X}$ represent reshape($\mathbf{D}\mathbf{x}$) and reshape($\mathbf{D}^T\mathbf{x}$), respectively, where $\mathbf{x}$ is the vectorized $\mathbf{X}$, and reshape($\cdot$) stands for the operation of reshaping vectors back to their matrix forms. Directly calculating the inverse of $2\mathbf{I} + \mu^{(t)}\mathbf{D}^T\mathbf{D}$ is an intuitive way to accomplish this job. However, the matrix inverse is computationally expensive, especially for large matrices like $\mathbf{D}^T\mathbf{D}$. Fortunately, by assuming circular boundary conditions, we can apply 2D FFT techniques on the above problem, which enables us to compute the solution fast. Consequently, we have

$$\mathbf{T}^{t+1} \leftarrow \mathcal{F}^{-1}\left(\frac{\mathcal{F}\big(2\hat{\mathbf{T}} + \mu^{(t)}\mathbf{D}^T(\mathbf{G}^{(t)} - \frac{\mathbf{Z}^{(t)}}{\mu^{(t)}})\big)}{\mathbf{2} + \mu^{(t)}\sum_{d\in\{h,v\}}\overline{\mathcal{F}(\mathbf{D}_d)} \circ \mathcal{F}(\mathbf{D}_d)}\right) \quad (13)$$

where $\mathcal{F}(\cdot)$ is the 2D FFT operator, while $\mathcal{F}^{-1}(\cdot)$ and $\overline{\mathcal{F}(\cdot)}$ stand for the 2D inverse FFT and the complex conjugate of $\mathcal{F}(\cdot)$, respectively. The division performs element-wise. In addition, $\mathbf{2}$ is the matrix with proper size, all the entries of which are 2.

G sub-problem: Dropping the terms unrelated to $\mathbf{G}$ leads to the following optimization problem:

$$\mathbf{G}^{(t+1)} \leftarrow \operatorname*{argmin}_{\mathbf{G}} \alpha\|\mathbf{W} \circ \mathbf{G}\|_1 + \Phi(\mathbf{Z}^{(t)}, \nabla\mathbf{T}^{(t+1)} - \mathbf{G}). \quad (14)$$

The closed form solution of (14) can be easily obtained by performing the shrinkage operation like:

$$\mathbf{G}^{(t+1)} = \mathcal{S}_{\frac{\alpha\mathbf{W}}{\mu^{(t)}}}\left[\nabla\mathbf{T}^{(t+1)} + \frac{\mathbf{Z}^{(t)}}{\mu^{(t)}}\right]. \quad (15)$$

$\mathcal{S}_{\varepsilon>0}[\cdot]$ represents the shrinkage operator, the definition of which on scalars is: $\mathcal{S}_\varepsilon[x] = \operatorname{sgn}(x)\max(|x| - \varepsilon, 0)$. The extension of the shrinkage operator to vectors and matrices is to simply process data element-wise, say $\mathcal{S}_\mathbf{A}[\mathbf{X}]$ performs the shrinkage on the elements of $\mathbf{X}$ with thresholds given by corresponding entries of $\mathbf{A}$.

Z and μ: The updating of $\mathbf{Z}$ and μ can be done via:

$$\begin{aligned}\mathbf{Z}^{(t+1)} &\leftarrow \mathbf{Z}^{(t)} + \mu^{(t)}(\nabla\mathbf{T}^{(t+1)} - \mathbf{G}^{(t+1)}); \\ \mu^{(t+1)} &\leftarrow \mu^{(t)}\rho, \rho > 1.\end{aligned} \quad (16)$$

For clarity, the entire procedure of exact solver to problem (8) is summarized in Algorithm 1. The iteration is stopped when $\|\nabla\mathbf{T}^{(t+1)} - \mathbf{G}^{(t+1)}\|_F \leq \delta\|\hat{\mathbf{T}}\|_F$ with $\delta = 10^{-5}$ or the maximal number of iterations is reached. Please refer to Algorithm 1 for other details that we can not cover in the text.

Remark 1 (Convergence and Optimality) As aforementioned, the problem (9) is equivalent to (8). We can observe that every term appears in the objective function of (9) is convex, and the constraint is affine. The proposed Algorithm 1 follows the framework of Augmented Lagrangian Multiplier with Alternating Direction Minimizing (ALM-ADM), the theoretical guarantee of which has been well established for two-block convex cases [26], [27]. In other words, our proposed exact solver converges to a global optimal solution to the problem (8), and thus to the original (9).

Algorithm 1: Exact Solver to Problem (8)

Input: The positive coefficient α, and the initial illumination map $\hat{\mathbf{T}} \in \mathbb{R}^{m \times n}$.
Initialization: $\mathbf{T}^{(0)} = \mathbf{0} \in \mathbb{R}^{m \times n}$,
  $\mathbf{G}^{(0)} = \mathbf{Z}^{(0)} = \mathbf{0} \in \mathbb{R}^{2m \times n}$, $t = 0$, $\mu^{(0)} > 0$, $\rho > 1$.
while *not converged* **do**
  Update $\mathbf{T}^{(t+1)}$ via Eq. (13);
  Update $\mathbf{G}^{(t+1)}$ via Eq. (15));
  Update $\mathbf{Z}^{(t+1)}$ and $\mu^{(t+1)}$ via Eq. (16);
  $t = t + 1$;
end
Output: Optimal solution $\mathbf{T}^* = \mathbf{T}^{(t)}$.
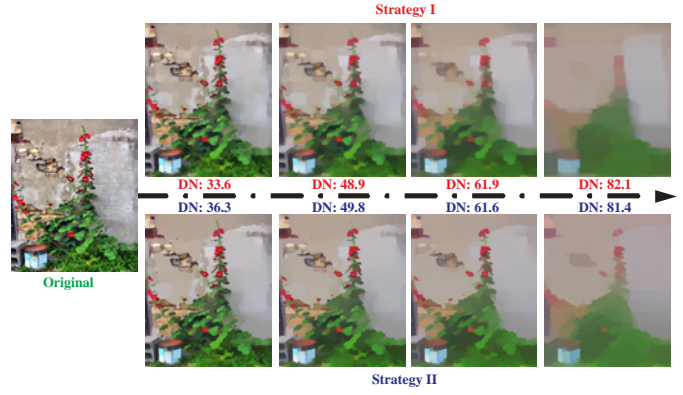

Fig. 5: Difference between the weighting strategy I and II with varying parameter α. For the first row (Strategy I), we use $\alpha \in \{0.2, 0.5, 1.0, 3.0\}$ corresponding to the four results, respectively. While for the second row (Strategy II), we use $\alpha \in 0.15 \times \{0.2, 0.5, 1.0, 3.0\}$ to do the test. The choice of the coefficient 0.15 is based on the observation that the difference norms (DN), defined as $\|\mathbf{T} - \hat{\mathbf{T}}\|_F$, of the two cases are close, so that the comparison is fair.
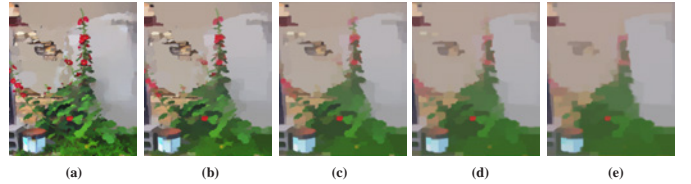

Fig. 6: Effect of parameter σ for weighting strategy III. **From (a) to (e):** the results by setting σ to 10^{-5}, 1, 2, 3 and 4 respectively with fixed $\alpha = 0.15 \times 0.2$.

Remark 2 (Computational Complexity) Each iteration of Algorithm 1 involves three sub-problems. Regarding the $\mathbf{T}$ sub-problem, it requires $\mathcal{O}(N \log N)$ to finish the computation, where N is the total amount of pixels. Its dominant cost comes from 2D FFT and inverse FFT operations. As for the $\mathbf{G}$ and $\mathbf{Z}$ sub-problems, they both are linear with respect to N, say $\mathcal{O}(N)$. Hence, each iteration takes $\mathcal{O}(N \log N)$. Based on the above, it is ready to conclude that the complexity of Algorithm 1 is $\mathcal{O}(tN \log N)$, where t is the number of iterations required to converge.

C. Sped-up Solver to Problem (8)

Although the complexity of Algorithm 1 is reasonably low, we want to further reduce it. Let us take a closer look at the problem (8). The origin bringing the iterative procedure is the sparse weighted gradient term, *i.e.* $\|\mathbf{W} \circ \nabla \mathbf{T}\|_1$. The ℓ_1 norm together with the gradient operation on $\mathbf{T}$ makes it somewhat complex. Clearly, the relationship below holds true:

$$\lim_{\epsilon \to 0^+} \sum_x \sum_{d \in \{h,v\}} \frac{\mathbf{W}_d(x)(\nabla_d \mathbf{T}(x))^2}{|\nabla_d \mathbf{T}(x)| + \epsilon} = \|\mathbf{W} \circ \nabla \mathbf{T}\|_1. \quad (17)$$

Based on the above, we use $\sum_x \sum_{d \in \{h,v\}} \frac{\mathbf{W}_d(x)(\nabla_d \mathbf{T}(x))^2}{|\nabla_d \hat{\mathbf{T}}(x)| + \epsilon}$ to approximate $\|\mathbf{W} \circ \nabla \mathbf{T}\|_1$. As a result, the approximate problem to (8) can be written as follows:

$$\min_{\mathbf{T}} \|\hat{\mathbf{T}} - \mathbf{T}\|_F^2 + \alpha \sum_x \sum_{d \in \{h,v\}} \frac{\mathbf{W}_d(x)(\nabla_d \mathbf{T}(x))^2}{|\nabla_d \hat{\mathbf{T}}(x)| + \epsilon}. \quad (18)$$

Although the objective function changes, compared to the original, the goal of extracting the structure of illumination from the initial illumination estimate $\hat{\mathbf{T}}$ is consistent with the original. More specifically, when $|\nabla_d \hat{\mathbf{T}}(x)|$ is small, $|\nabla_d \mathbf{T}(x)|$ is about to be suppressed, so is the value $\frac{(\nabla_d \mathbf{T}(x))^2}{|\nabla_d \hat{\mathbf{T}}(x)| + \epsilon}$. In other words, the target $\mathbf{T}$ is constrained to avoid creating gradients where the initially estimated illumination map has small magnitudes of gradient. In contrary, if $|\nabla_d \hat{\mathbf{T}}(x)|$ is strong, the above suppression alleviates, because this location is more likely on structure boundary than on regular texture.

As can be observed, the problem (18) only involves quadratic terms. Thus, the problem can be directly obtained by solving the following:

$$\left(\mathbf{I} + \sum_{d \in \{u,v\}} \mathbf{D}_d^T \operatorname{Diag}(\tilde{\mathbf{w}}_d) \mathbf{D}_d\right) \mathbf{t} = \hat{\mathbf{t}}, \quad (19)$$

where $\tilde{\mathbf{w}}_d$ is the vectorized version of $\tilde{\mathbf{W}}_d$ with $\tilde{\mathbf{W}}_d(x) \leftarrow \frac{\mathbf{W}_d(x)}{|\nabla_d \hat{\mathbf{T}}(x)| + \epsilon}$. Further, the operator $\operatorname{Diag}(\mathbf{x})$ is to construct a diagonal matrix using vector $\mathbf{x}$. Since $(\mathbf{I} + \sum_{d \in \{u,v\}} \mathbf{D}_d^T \operatorname{Diag}(\tilde{\mathbf{w}}_d) \mathbf{D}_d)$ is a symmetric positive definite Laplacian matrix, there are many techniques available for solving it, for example, [28], [29], [30], [31], [32].

Remark 3 (Computational Complexity) Solvers such as the multi-resolution preconditioned conjugate gradient can reach $\mathcal{O}(N)$ complexity. Compared to the complexity of Algorithm 1, the sped-up solver eliminates the iteration requirement t as well as reduces $N \log N$ to N.

D. Possible Weighting Strategies

For the structure-aware refinement on the initial illumination map, the key is the design of $\mathbf{W}$. In this part, we discuss three possible weighing strategies as follows.

Strategy I: It can be seen that setting the weight matrix as

$$\mathbf{W}_h(x) \leftarrow 1; \quad \mathbf{W}_v(x) \leftarrow 1, \quad (20)$$

leads (8) to a classic ℓ_2 loss total variation minimization problem [33].

Strategy II: As discussed in Sec. II-C, it is reasonable to use the gradient of the initial illumination map as the weight.
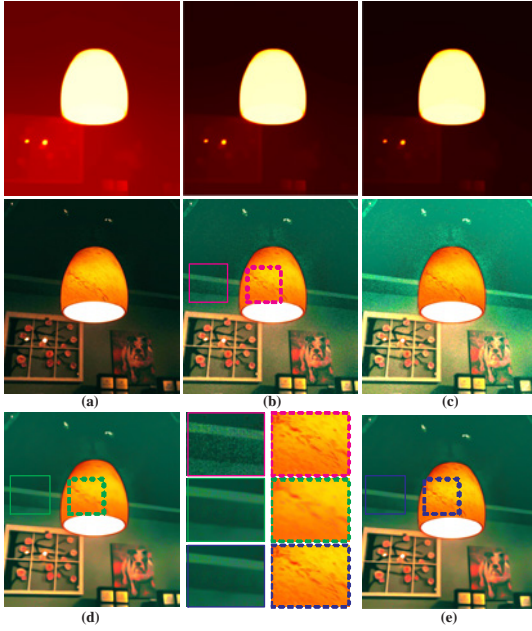
Fig. 7: Gamma correction for illumination maps in the range of $[0, 1]$, denosing and recomposition. (a)-(c) are the recovered images using $\mathbf{T}^\gamma$ with $\gamma = 0.5$, $\gamma = 0.8$ and $\gamma = 1$, respectively. The corresponding illumination map is given on the top of each recovery. Noises appear in the light enhanced images. (d) is the denoised version of (b), while (e) is the recomposed result of (b) and (d). It can be seen from the zoomed-in patches that the recomposition adaptively keeps the fine details of the bright region and suppresses the noises of the dark region.

In the sequel, we have:

$$\mathbf{W}_h(x) \leftarrow \frac{1}{|\nabla_h \hat{\mathbf{T}}(x)| + \epsilon}; \quad \mathbf{W}_v(x) \leftarrow \frac{1}{|\nabla_v \hat{\mathbf{T}}(x)| + \epsilon}. \quad (21)$$

Strategy III: Inspired by Relative Total Variation (RTV) [34], for each location, the weight is set via:

$$\begin{aligned}
\mathbf{W}_h(x) &\leftarrow \sum_{y \in \Omega(x)} \frac{G_\sigma(x, y)}{|\sum_{y \in \Omega(x)} G_\sigma(x, y) \nabla_h \hat{\mathbf{T}}(y)| + \epsilon}; \\
\mathbf{W}_v(x) &\leftarrow \sum_{y \in \Omega(x)} \frac{G_\sigma(x, y)}{|\sum_{y \in \Omega(x)} G_\sigma(x, y) \nabla_v \hat{\mathbf{T}}(y)| + \epsilon},
\end{aligned} \quad (22)$$

where $G_\sigma(x, y)$ is produced by the Gaussian kernel with the standard deviation σ. Formally, $G_\sigma(x, y)$ is expressed as:

$$G_\sigma(x, y) \propto \exp\left(-\frac{\text{dist}(x, y)}{2\sigma^2}\right), \quad (23)$$

where the function $\text{dist}(x, y)$ is to measure the spatial Euclidean distance between locations x and y. In fact, the second weighting strategy is an instance of this one. When $\sigma \rightarrow 0^+$, the two strategies obtain the same weight matrix. We note that, different from RTV, our weight matrix is constructed based on the given $\hat{\mathbf{T}}$ instead of being iteratively updated according to $\mathbf{T}$. That means $\mathbf{W}$ only needs to be calculated once.

Algorithm 2: LIME

Input: Low-light Input $\mathbf{L}$, positive coefficient α, Gamma transformation parameter γ.

Initialization: Constructing weight matrix by Eq. (20), Eq. (21) or Eq. (22)

Do the job

  1. Estimate initial illumination map $\hat{\mathbf{T}}$ on $\mathbf{L}$ via Eq. (2);

  2. Refine illumination map $\mathbf{T}$ based on $\hat{\mathbf{T}}$ via exact solver Alg. 1 or sped-up solver Eq. (19);

  3. Gamma correction on $\mathbf{T}$ via $\mathbf{T} \leftarrow \mathbf{T}^\gamma$;

  4. Enhance $\mathbf{L}$ using $\mathbf{T}$ according to $\mathbf{L} = \mathbf{R} \circ \mathbf{T}$;

  5. If denoising and recomposing needed, then denoise $\mathbf{R}$ by BM3D ($\mathbf{R}_d$) and recompose via Eq. (24).

Output: Final enhanced result

E. Other Operations

Having the refined illumination map $\mathbf{T}$, we can recover $\mathbf{R}$ by following (3). One can also manipulate the illumination map through gamma transformation, say $\mathbf{T} \leftarrow \mathbf{T}^\gamma$. From the upper row of Fig. 7, we can see the difference between the results by setting γ to 0.5, 0.8 and 1. For the rest experiments, we adopt $\gamma = 0.8$. Moreover, possible noises previously hiding in the dark are also accordingly amplified, especially for the very low-light inputs (regions), as shown in Fig. 7. Denoising techniques are required to further improve the visual quality. Many off-the-shelf denosing tools, such as [35], [36], [37], can be employed to do the job. Considering the comprehensive performance, BM3D [35] is the choice of this work. In our implementation, for further cutting the computational load, we only execute BM3D on the Y channel by converting $\mathbf{R}$ from the RGB colorspace into the YUV one. In addition, the magnitude of noises is not the same for different regions of the input, as the amplification is different. And BM3D treats different patches equally. Therefore, to avoid the unbalance of processing, *e.g.* some (dark) places are well-denoised while some (brighter) over-smoothed, we employ the following operation:

$$\mathbf{R}_f \leftarrow \mathbf{R} \circ \mathbf{T} + \mathbf{R}_d \circ (\mathbf{1} - \mathbf{T}), \quad (24)$$

where $\mathbf{R}_d$ and $\mathbf{R}_f$ are the results after denoising and recomposing, respectively. The merit of this operation can be viewed from Fig. 7 (e), compared with Fig. 7 (d). We would like to mention that the denoising plus recomposing, as a post-processing step, can be concatenated to any low-light image enhancing method. The whole procedure of LIME is outlined in Algorithm 2.

Please note that, sometimes other specific techniques are needed to remedy the complication caused by low-light enhancement. For image compression using like JPEG [38], the blocking effect becomes noticeable in the low-light enhanced results. Hence, deblocking techniques [39], [40], [41], [42] may be required. Further, for color distorted images, adopting some color constancy methods [43], [44] as post-processing can alleviate the negative effect. In this paper, we do not consider these issues (and other possible complications caused by low-light enhancement) for avoiding distraction.

| (a) Initial | (b) Max | (c) BF | (d) Our Exact | (e) Our Sped-up |

Fig. 8: Comparison of different illumination maps and corresponding enhanced results. **From (a) to (f):** Illumination map estimated individually on each pixel (Initial), refined by local max (7) (Max), bilateral filtering (BF), our results by exact solver and sped-up solver, respectively.

III. EXPERIMENTS

In this section, we first see the performance difference between different weighting strategies, and the effect of involved parameters. Next, the analysis of our exact and sped-up solvers is given. Then, we qualitatively and quantitatively compare our LIME with several state-of-the-art methods, including Histogram Equalization (HE), Adaptive Histogram Equalization (AHE), Gamma Correction (GC), Contextual and Variational Contrast enhancement (CVC) [6], Layered Difference Representation (LDR) [7], dehazing based method [14] (DeHz), Multi-deviation Fusion method (MF) [12], Naturalness Preserved Enhancement algorithm (NPE) [11] and Simultaneous Reflection and Illumination Estimation (SRIE) [13]. All the codes are in Matlab[2], which ensures the fairness of time comparison. All the experiments are conducted on a PC running Windows 7 OS with 64G RAM and 2.4GHz CPU.

---

[2]HE and AHE uses *histeq* and *adapthisteq* functions integrated in the Matlab toolbox. GC is achieved by $\mathbf{L}^\gamma$, while the codes of CVC, LDR, MF, NPE and SRIE are downloaded from the authors' websites. The code of DeHz is not publicly available when this paper is prepared, but it is easy to be implemented based on [20].

A. Weighting Strategy and Parameter Effect

Here, we evaluate the performance difference brought by three different weighting strategies with varying parameters α and σ of our model (8). The relationship between the weighting strategy II and III, as aforementioned, is that the former one is a special case of the latter one. Their equivalence is reached by setting $\sigma \to 0^+$ in (22). To clearly see the effect of each parameter, we first compare the weighting strategy I and II by varying the parameter α without σ involved. The effect of σ in the weighting strategy III is then tested by keeping α fixed and varying σ. Please notice that, for easier and better viewing the performance difference, we employ a color image with good visibility instead of an illumination map in this test.

Figure 5 provides the comparison of the weighting strategy I and II. It is worth mentioning that the value scale of the term $\|\mathbf{W} \circ \nabla \mathbf{T}\|_1$ may significantly change when applying different weighting strategies. Thus, for comparison fairness, we need to eliminate the scale issue. To this end, we alternatively control the difference norms (DN) of the two cases, defined as $\|\mathbf{T} - \hat{\mathbf{T}}\|_F$, to be sufficiently close. For the upper row (Strategy I), we use $\alpha \in \{0.2, 0.5, 1.0, 3.0\}$ corresponding to the four results, respectively. While for the lower row (Strategy II), we use $\alpha \in 0.15 \times \{0.2, 0.5, 1.0, 3.0\}$ to accomplish the
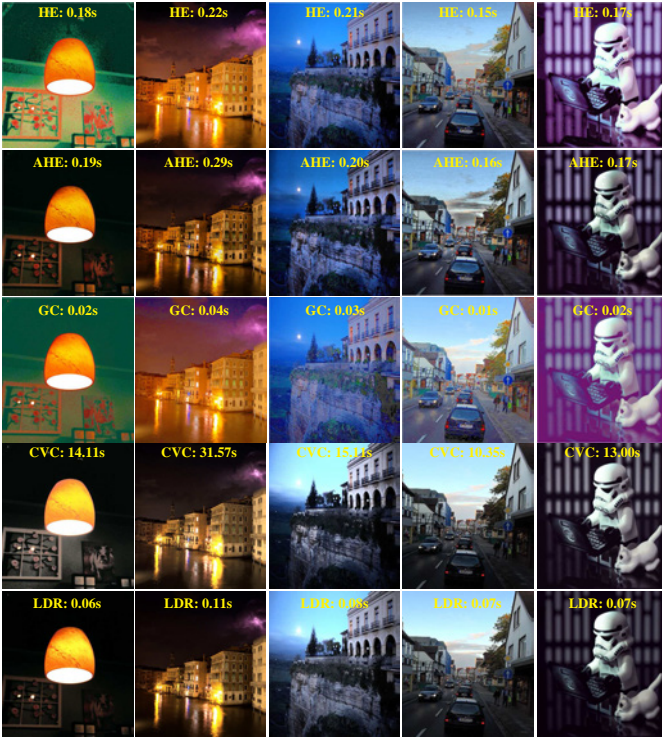
Fig. 9: Result comparison between HE, AHE, GC, CVC and LDR. Please see also Fig. 10.
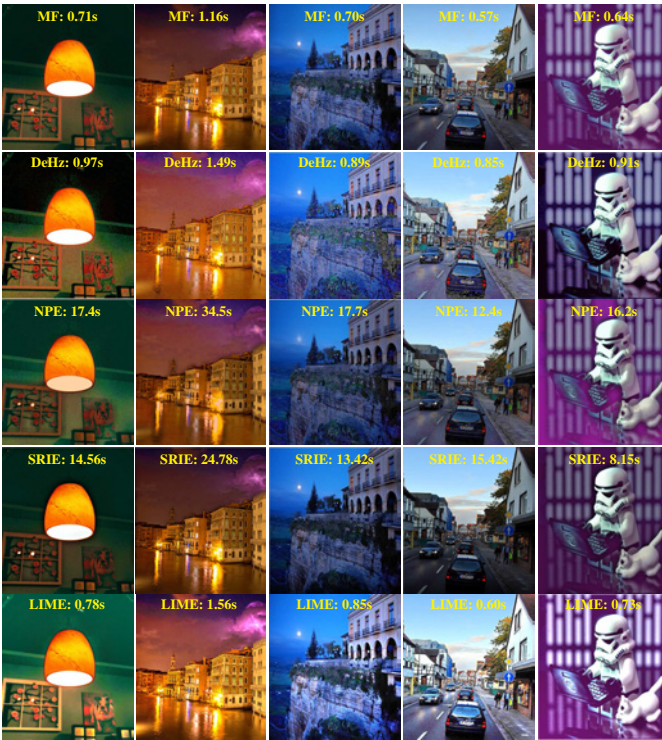

Fig. 10: Result comparison between MF, DeHz, NPE, SRIE and LIME. Please see also Fig. 9.

test. It is certain that as α grows, the fidelity between $\mathbf{T}$ and $\hat{\mathbf{T}}$ decreases, as well as the smoothness of the desired $\mathbf{T}$ increases. From the picture pairs shown in Fig. 5, we observe that the results by the weighting strategy II retain the overall structure as well as smooth the texture better than those by the strategy I. This corroborates that the ℓ^2 loss TV model is short of ability to distinguish between strong structural edges and texture [34]. Further, Figure 6 shows a set of results obtained by fixing $\alpha = 0.15 \times 0.2$ and varying σ in the weighting strategy III. The result using $\sigma = 10^{-5}$, as shown in Fig. 6 (b), confirms the relationship between the weighting strategy II and III. From Fig. 6 (a) to (e), we can see the smoothing effect becomes heavier. This is because increasing σ involves more neighboring locations taken into account. Even though, the overall structure is well preserved. For the rest experiments, we simply adopt the weighting strategy III with $\sigma = 2$.

B. Exact Solver vs. Sped-up Solver

Although the theoretical complexity of the exact solver is given as $\mathcal{O}(tN \log N)$, it would be more intuitive to see how many iterations Algorithm 1 needs to converge empirically. The left picture in Fig. 13 plots four curves. From the plots, we find that the trends of the four plots are very close. And the exact algorithm converges within 60 iterations for all the four tests. In the rest experiments, we set the maximum number of iterations of Algorithm 1 to 60. Please notice that, to eliminate the scale difference in the stop criterion, we have normalized the stop criterion in $[0, 1]$ for different cases.

The second picture shown in Fig. 13 gives the comparison between the exact and sped-up solvers in terms of time cost. From the curves, we see that when the image size is smaller than 400, both the two solvers are sufficiently efficient. But, after that, the processing time demanded by the exact solver rapidly grows up while that of the sped-up solver varies gently and keeps reasonably low. In other words, the benefit of the sped-up solver compared with the exact one becomes more conspicuous, as the image size increases.

Figure 8 shows a comparison of the exact solver and sped-up solver on illumination maps, from which, we can see that the illumination maps by the exact solver are sharper than those by the sped-up solver. We adopt $\alpha = 0.15$ for both the exact and sped-up solvers in this subsection and for all the rest experiments. Although there is a gap between the illumination maps obtained by these two solvers, the visual difference between their enhanced results is acceptable. Sometimes, relatively soft illumination maps may provide visually more comfortable results (see the second case). Considering the efficiency, the sped-up solver is more attractive for practical use. Further, compared to the other illumination maps, the advance of our results is obvious.

C. Comparison

Figures 9 and 10 provide several comparisons. The inputs are from the top row of Fig. 1. The operations of HE, AHE and GC are executed on the V channel of images by first converting it from the RGB colorspace to the HSV one and then converting the processed HSV back to the RGB
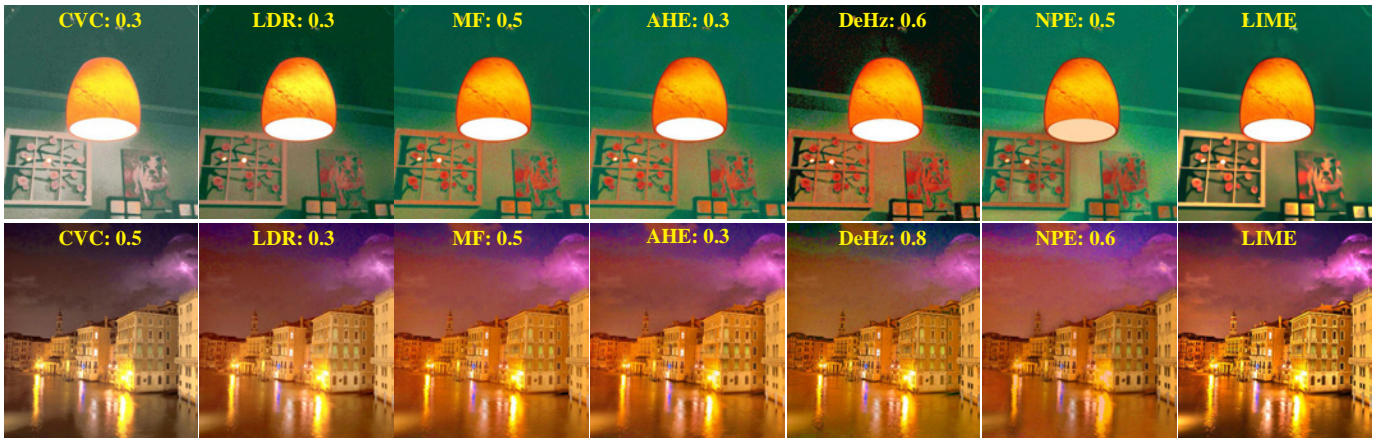
Fig. 11: Results with GC operation as post-processing. We tune γ (given on the top of each picture) of GC to achieve their best possible visual quality.

(a) Input   (b) Illumination map   (c) Without denoising   (d) With denoising
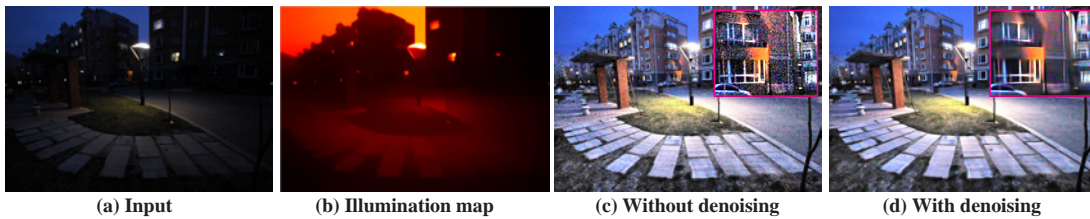
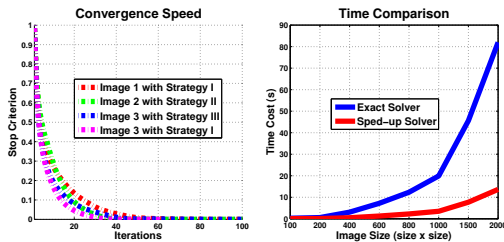Fig. 12: Comparison of results without and with denoising.

Fig. 13: **Left:** convergence speed of the proposed exact solver (Algorithm 1). The images used in this experiment are, without loss of generality, the second and third pictures in Fig. 1 (Image 1 and 2), and the one in Fig. 4 (Image 3), respectively. **Right:** time comparison between the exact solver and the sped-up one with varying image sizes.

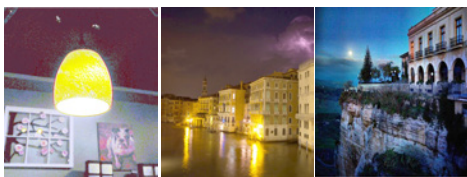Fig. 14: **From Left to Right:** results by HE, GC and AHE directly on the RGB colorspace, respectively.

Fig. 15: Samples from the HDR dataset

colorspace. Directly manipulating on each channel of R, G and B leads results to be visually inconsistent with real scenes, *e.g.* considerably changing the tone of images, please see Fig. 14 for example. We can observe from Fig. 9 and 10 that AHE,

CVC and LDR can not effectively recall the information in dark regions. This problem almost exist always. HE, DeHz, MF, SRIE and NPE outperform AHE, CVC and LDR in most of the given examples, as shown in Fig. 9 and 10, but are inferior to our method (Strategy III, $\sigma = 2$, $\alpha = 0.15$, Sped-up solver) in terms of visual quality. In terms of time cost, although our method spends more than HE, AHE, LDR and GC, it is comparable to or even more efficient than MF and DeHz, while much more efficient than NPE, SRIE and CVC. Most cost of DeHz comes from the estimation of atmospheric light.

From the pictures shown in Fig. 9 and 10, the lightness is still somewhat dim for CVC, LDR, AHE, NPE, MF and NPE, which can be further enhanced by gamma correction intuitively. We note that SRIE itself has a gamma correction step on the estimated illumination. Figure 11 depicts the results

TABLE I: Quantitative performance comparison on the HDR dataset in terms of LOE. LOE has a factor 10^3.

Method	BabyAtWin	BabyOnGrass	ChrisRider	FeedingTime	HighChair	LadyEating	PianoMan	SantaHelper	Ave. LOE
HE	4.536	4.492	2.433	3.117	3.127	3.395	3.759	3.652	3.564
AHE	3.481	2.471	2.127	2.098	**1.919**	2.648	2.591	2.907	2.530
GC	4.518	4.496	2.430	3.101	3.141	3.401	3.755	3.645	3.561
CVC	4.549	4.488	2.557	3.132	3.148	3.402	3.823	3.695	3.599
LDR	4.501	4.500	2.509	3.120	3.134	3.401	3.775	3.670	3.572
MF	3.626	2.838	2.124	**2.005**	2.291	2.749	3.113	3.145	2.736
NPE	3.811	4.489	3.191	3.183	3.401	3.250	3.872	3.773	3.621
DeHz	4.591	4.527	2.854	3.114	3.227	3.408	3.837	3.732	3.661
SRIE	4.133	4.224	2.770	3.047	3.201	3.196	3.233	3.497	3.413
LIME	**3.263**	**2.356**	**1.945**	2.091	2.330	**2.305**	**2.513**	**2.352**	**2.394**

after executing gamma correction. For the pictures obtained by different methods, we tune the parameter γ to achieve their possible best visual quality. The lightness is indeed increased, but similar to using only GC, the visual artifact appears for all the further enhanced results of AHE, CVC, LDR, MF, DeHz and NPE. This is mainly because the nonlinear operation of GC is carried out on each pixel individually without considering the relationship of a certain pixel with its neighbors. Although our LIME also employs the GC as described in Sec. II-E, the estimated illumination map itself is structure-aware and thus LIME survives from such artifacts. It is worth noticing that the parameter γ used in LIME is fixed to 0.8 for all the experiments in Sec. III instead of being fine-tuned image by image.

Figure 12 gives another test. The very low-light input hides intensive noises in the dark. After performing LIME, the details of the scene get enhanced, but the noises also come out, as shown in the middle of Fig. 12. This is an inevitable problem encountered by almost all of existing low-light enhancement algorithms. As we have discussed in Sec. II-E, denoising is required. The right picture in Fig. 12 is the denoised result by executing BM3D on the middle of Fig. 12, from which we can see the improvement in terms of visual quality. Fig. 17 provides more qualitative results by LIME.

As pointed out in [11], the relative order of lightness represents the light source directions and the lightness variation, the naturalness of an enhanced image is related to the relative order of lightness in different local areas. Therefore, we employ the lightness order error (LOE) as our objective metric to measure the performance. The definition of LOE is as follows:

$$LOE = \frac{1}{m} \sum_{x=1}^{m} \sum_{y=1}^{m} \left(U(\mathbf{Q}(x), \mathbf{Q}(y)) \oplus U(\mathbf{Q}_r(x), \mathbf{Q}_r(y)) \right),$$
(25)

where m is the pixel number. The function $U(p, q)$ returns 1 if $p >= q$, 0 otherwise. $\oplus$ stands for the exclusive-or operator. In addition, $\mathbf{Q}(x)$ and $\mathbf{Q}_r(x)$ are the maximum values among R, G and B channels at location x of the enhanced and reference images, respectively. The lower the LOE is, the better the enhancement preserves the naturalness of lightness. Due to the heavy load of computing LOE, as suggested in [11], down-sampling is used to reduce the complexity. Similarly, in this part, we set the resize factor r to $100/\min(h, w)$, where h and w are the height and width of the image, respectively.

From the definition of LOE, we notice that $\mathbf{Q}_r$ plays an important role in quantitatively measuring the quality of enhancement. Using the low-light input as the reference is problematic. Because, take an extreme case for example, the LOE is 0 when no enhancement is performed. While few datasets with ground truth are publicly available and it is not easy to construct such datasets, we have to choose reliable data to do the job for the sake of objectiveness. Different from [11], we adopt the HDR [45] result as the reference instead of the input low-light image. The HDR reconstruction results from a set of bracketed exposures are more proper to act as the reference. The HDR dataset contains eight groups, several samples from this dataset are shown in Fig. 15. Table I contains the LOE numbers of all the competitors on the HDR dataset. From the numbers, we observe that our LIME significantly outperform the others. In addition, we give the visual comparison on two cases in Fig. 16, from which, we can find that the results obtained by LIME are more visually pleasant and closer to the references than the others. To allow more experimental verification and comparisons, we release our code at http://cs.tju.edu.cn/orgs/vision/~xguo/LIME.htm.

IV. CONCLUSION

In this paper, we have proposed an efficient and effective method to enhance low-light images. The key to the low-light enhancement is how well the illumination map is estimated. The structure-aware smoothing model has been developed to improve the illumination consistency. We have designed two algorithms: one can obtain the exact optimal solution to the target problem, while the other alternatively solves the approximate problem with significant saving of time. Moreover, our model is general to different (structure) weighting strategies. The experimental results have revealed the advance of our method compared with several state-of-the-art alternatives. It is positive that our low-light image enhancement technique can feed many vision-based applications, such as edge detection, feature matching, object recognition and tracking, with high visibility inputs, and thus improve their performance.

REFERENCES

[1] D. Oneata, J. Revaud, J. Verbeek, and C. Schmid, "Spatio-temporal object detection proposals," in *ECCV*, pp. 737–752, 2014.

[2] K. Zhang, L. Zhang, and M. Yang, "Real-time compressive tracking," in *ECCV*, pp. 866–879, 2014.

Fig. 16: Visual comparison among the competitors on the HDR dataset.

[3] E. Pisano, S. Zong, B. Hemminger, M. DeLuce, J. Maria, E. Johnston, K. Muller, P. Braeuning, and S. Pizer, "Contrast limited adaptive histogram equalization image processing to improve the detection of simulated spiculations in dense mammograms," *Journal of Digital Imaging*, vol. 11, no. 4, pp. 193–200, 1998.

[4] H. Cheng and X. Shi, "A simple and effective histogram equalization approach to image enhancement," *Digital Signal Processing*, vol. 14, no. 2, pp. 158–170, 2004.

[5] M. Abdullah-Al-Wadud, M. Kabir, M. Dewan, and O. Chae, "A dynamic histograme equalization for image contrast enhancement," *IEEE Trans. on Consumer Electronics*, vol. 53, no. 2, pp. 593–600, 2007.

[6] T. Celik and T. Tjahjadi, "Contextual and variational contrast enhancement," *TIP*, vol. 20, no. 12, pp. 3431–3441, 2011.

[7] C. Lee and C. Kim, "Contrast enhancement based on layered difference representation," *TIP*, vol. 22, no. 12, pp. 5372–5384, 2013.

[8] E. Land, "The retinex theory of color vision," *Scientific American*, vol. 237, no. 6, pp. 108–128, 1977.

[9] D. Jobson, Z. Rahman, and G. Woodell, "Properties and performance of a center/surround retinex," *TIP*, vol. 6, no. 3, pp. 451–462, 1996.

[10] D. Jobson, Z. Rahman, and G. Woodell, "A multi-scale retinex for bridging the gap between color images and the human observation of scenes," *TIP*, vol. 6, no. 7, pp. 965–976, 1997.

[11] S. Wang, J. Zheng, H. Hu, and B. Li, "Naturalness preserved enhancement algorithm for non-uniform illumination images," *TIP*, vol. 22, no. 9, pp. 3538–3578, 2013.

[12] X. Fu, D. Zeng, Y. Huang, Y. Liao, X. Ding, and J. Paisley, "A fusion-based enhancing method for weakly illuminated images," *Signal Processing*, vol. 129, pp. 82–96, 2016.

[13] X. Fu, D. Zeng, Y. Huang, X. Zhang, and X. Ding, "A weighted varia-

tional model for simultaneous reflectance and illumination estimation," in *CVPR*, pp. 2782–2790, 2016.

[14] X. Dong, G. Wang, Y. Pang, W. Li, J. Wen, W. Meng, and Y. Lu, "Fast efficient algorithm for enhancement of low lighting video," in *ICME*, pp. 1–6, 2011.

[15] L. Li, R. Wang, W. Wang, and W. Gao, "A low-light image enhancement method for both denoising and contrast enlarging," in *ICIP*, pp. 3730–3734, 2015.

[16] R. Grosse, M. Johnson, E. Adelson, and W. Freeman, "Ground-truth dataset and baseline evaluations for intrinsic image algorithms," in *ICCV*, pp. 2335–2342, 2009.

[17] P. Gehler, C. Rother, M. Kiefel, L. Zhang, and B. Schölkopf, "Recovering intrinsic images with a global sparsity prior on reflectance," in *NIPS*, pp. 765–773, 2011.

[18] J. Barron and J. Malik, "Color constancy, intrinsic images, and shape estimation," in *ECCV*, pp. 57–70, 2012.

[19] Y. Li and M. Brown, "Single image layer separation using relative smoothness," in *CVPR*, pp. 2752–2759, 2014.

[20] K. He, J. Sun, and X. Tang, "Single image haze removal using dark channel prior," *TPAMI*, vol. 33, no. 12, pp. 2341–2353, 2011.

[21] S. Narasimhan and S. Nayar, "Contrast restoration of weather degraded images," *TPAMI*, vol. 25, no. 6, pp. 713–724, 2003.

[22] G. Meng, Y. Wang, J. Duan, S. Xiang, and C. Pan, "Efficient image dehazing with boundary constraint and contextual regularization," in *ICCV*, pp. 617–624, 2013.

[23] D. Forsyth, "A novel approach to color constancy," in *ICCV*, pp. 9–18, 1988.

[24] B. Funt and L. Shi, "The rehabilitation of maxrgb," in *Proceedings of Color Imaging Conference*, pp. 256–259, 2010.
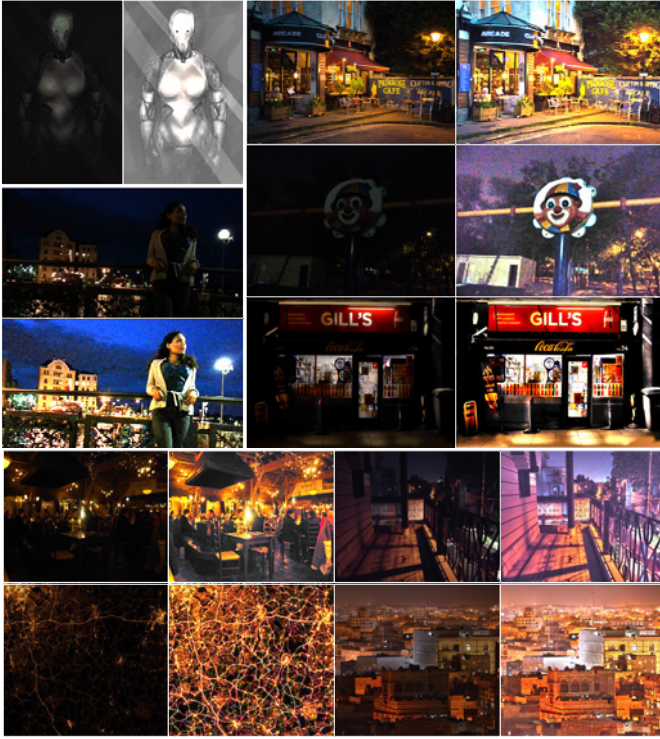
Fig. 17: More results by our proposed LIME (sped-up). All the results are obtained by adopting weighting strategy III and fixing $\alpha = 0.15$, $\sigma = 2$ and $\gamma = 0.8$. The results are better viewed in color.

[25] H. Joze, M. Drew, G. Finlayson, and P. Rey, "The role of bright pixels in illumination estimation," in *Proceedings of European Conference on Colour in Graphics, Imaging and Vision*, 2012.

[26] Z. Lin, R. Liu, and Z. Su, "Linearized alternating direction method with adaptive penalty for low rank representation," in *NIPS*, pp. 612–620, 2011.

[27] M. Hong and Z. Luo, "On the linear convergence of the alternating direction method of multipliers," *arXiv preprint arXiv:1208.3922*, 2013.

[28] Z. Farbman, R. Fattal, D. Lischinski, and R. Szeliski, "Edge-preserving decomposition for multi-scale tone and detail manipulation," *TOG*, vol. 27, no. 3, 2008.

[29] D. Krishnan and R. Szeliski, "Multigrid and multilevel preconditioners for computational photography," *TOG*, vol. 30, no. 6, 2011.

[30] A. Levin, D. Lischinski, and Y. Weiss, "Colorization using optimization," *TOG*, vol. 23, no. 3, pp. 689–694, 2004.

[31] D. Lischinski, Z. Farbman, M. Uyttendaele, and R. Szeliski, "Interactive local adjustment of tonal values," *TOG*, vol. 25, no. 3, pp. 646–653, 2006.

[32] R. Szeliski, "Locally adapted hierarchical basis preconditioning," *TOG*, vol. 25, no. 3, pp. 1135–1143, 2006.

[33] S. Chan, R. Khoshabeh, K. Gibson, P. Gill, and T. Nguyen, "An augmented lagrangian method for total variation video restoration," *TIP*, vol. 20, no. 11, pp. 3097–3111, 2011.

[34] L. Xu, Q. Yan, Y. Xia, and J. Jia, "Strucutre extraction from texture via relative total variation," *TOG*, vol. 31, no. 6, pp. 139:1–139:10, 2013.

[35] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3d transform-domain collaborative filtering," *TIP*, vol. 16, no. 8, pp. 2080–2095, 2007.

[36] H. Burger, C. Schuler, and S. Harmeling, "Image denoising: Can plain neural networks compete with bm3d," in *CVPR*, pp. 2392–2399, 2012.

[37] S. Gu, L. Zhang, W. Zuo, and X. Feng, "Weighted nuclear norm minimization with applications to image denoising," in *CVPR*, pp. 2862–2869, 2014.

[38] "Information technology digital compression and coding of continuous-tone still images requirements and guidelines," tech. rep., International Telecommunication Union, 1992.

[39] Y. Li, F. Guo, R. Tan, and M. Brown, "A contrast enhancement framework with jpeg artifacts suppression," in *ECCV*, pp. 174–188, 2014.

[40] S. Li, J. Zhang, and X. Guo, "Visual data deblocking using structural layer priors," in *ICME*, 2016.

[41] I. Choi, S. Kim, M. Brown, and Y. Tai, "A learning-based approach to reduce JPEG artifacts in image matting," in *ICCV*, pp. 2880–2887, 2013.

[42] A. Foi, V. Katkovnik, and K. Egiazarian, "Pointwise shape-adaptive dct for high-quality denoising and deblocking of grayscale and color images," *TIP*, vol. 16, no. 5, pp. 1395–1411, 2007.

[43] G. Buchsbaum, "A spatial processor model for object colour perception," *J. Frank. Inst.*, vol. 310, no. 1, pp. 1–26, 1980.

[44] J. Vazquz-Corral, M. Vanrell, R. Baldrich, and F. Tous, "Color constancy by category correlation," *TIP*, vol. 21, no. 4, pp. 1997–2007, 2012.

[45] P. Sen, N. Kalantari, M. Yaesoubi, S. Darabi, D. Goldman, and E. Shechtman, "Robust patch-based hdr reconstruction of dynamic scenes," *TOG*, vol. 31, no. 6, pp. 203:1–203:11, 2012.

Xiaojie Guo (M'13) received the B.E. degree in software engineering from the School of Computer Science and Technology, Wuhan University of Technology, Wuhan, China, in 2008, and the M.S. and Ph.D. degrees in computer science from the School of Computer Science and Technology, Tianjin University, Tianjin, China, in 2010 and 2013, respectively. He is currently an Associate Professor with the Institute of Information Engineering, Chinese Academy of Sciences. He was a recipient of the Piero Zamperoni Best Student Paper Award in the International Conference on Pattern Recognition (International Association on Pattern Recognition), in 2010.

Yu Li (M'16) received his Ph.D. degree in National University of Singapore. He is now with Advanced Digital Sciences Center, a research center founded by University of Illinois at Urbana-Champaign (UIUC) and the Agency for Science, Technology and Research (A*STAR), Singapore. His research interests include computer vision, computational photography, and computer graphics.

Haibin Ling received the B.S. degree in mathematics and the MS degree in computer science from Peking University, China, in 1997 and 2000, respectively, and the PhD degree from the University of Maryland, College Park, in Computer Science in 2006. From 2000 to 2001, he was an assistant researcher at Microsoft Research Asia. From 2006 to 2007, he worked as a postdoctoral scientist at the University of California Los Angeles. After that, he joined Siemens Corporate Research as a research scientist. Since fall 2008, he has been with Temple University where he is now an Associate Professor. Dr. Ling's research interests include computer vision, augmented reality, medical image analysis, and human computer interaction. He received the Best Student Paper Award at the ACM Symposium on User Interface Software and Technology (UIST) in 2003, and the NSF CAREER Award in 2014. He is an Associate Editor of IEEE Trans. on Pattern Analysis and Machine Intelligence and serves on the editorial board of Pattern Recognition, and served as Area Chairs for CVPR 2014 and CVPR 2016.