Multispectral Joint Image Restoration via Optimizing a Scale Map

Xiaoyong Shen, *Student Member, IEEE*, Qiong Yan, Li Xu, *Member, IEEE*, Lizhuang Ma, *Member, IEEE*, and Jiaya Jia, *Senior Member, IEEE*

Abstract—Color, infrared and flash images captured in different fields can be employed to effectively eliminate noise and other visual artifacts. We propose a two-image restoration framework considering input images from different fields, for example, one noisy color image and one dark-flashed near-infrared image. The major issue in such a framework is to handle all structure divergence and find commonly usable edges and smooth transitions for visually plausible image reconstruction. We introduce a novel scale map as a competent representation to explicitly model derivative-level confidence and propose new functions and a numerical solver to effectively infer it following our important structural observations. Multispectral shadow detection is also used to make our system more robust. Our method is general and shows a principled way to solve multispectral restoration problems.

Index Terms—Image restoration, image denoise, joint filtering, shadow detection, multispectral image, depth enhancement

1 Introduction

Images captured in dim light are hardly satisfactory. They could be noisy when increasing ISO in a short exposure duration. Using flash might improve lighting; but it creates unwanted shadow and highlight, or changes tone of the image. The methods of [1], [7], [20] restore a color image based on flash and no-flash inputs of the same scene. Recently, because of the popularity of other imaging devices, more computational photography and computer vision solutions based on images captured under different configurations were developed.

For example, near infrared (NIR) images are with a single channel recording infrared light reflected from objects with the spectrum ranging from 700-1,000 nm in wavelength. NIR images contain many similar structures as visible color ones when taken from the same camera position. This enables a configuration to take an NIR image with less noisy details by dark flash [15] to guide corresponding noisy color image restoration. The main advantage is on only using NIR flash invisible to naked human eyes, making it a suitable way for daily portrait photography and of remarkable practical importance.

In previous methods, Krishnan and Fergus [15] used gradients of a dark-flashed image capturing ultraviolet (UV) and NIR light to guide noise removal in the color image. Considering rich details in NIR images, Zhang et al. [31] enhanced the RGB counterpart by transferring contrast and detail via Haar wavelets. In [32] and [23], the detail layer was manipulated differently for RGB and haze image enhancement.

Methods exploring other image fusion applications are two-image deblurring [30], matting [24], tone mapping [8], upsampling [14], context enhancement [21], relighting [2], to name a few. Bhat et al. [3] proposed GradientShop to edit gradients, which can also be used to enhance images.

We note existing methods work well for their respective applications by handling different detail layers or gradients from multiple images. But in terms of two-image high-quality image restoration, there remain a few major and fundamental issues. We take the RGB-NIR images shown in Fig. 1 as an example to reveal issues generally existing in multispectral images. In this example, an NIR image differs from the corresponding RGB one in detail distribution and intensity formation. Structure inconsistency on many pixels can be categorized as follows.

- **Gradient Magnitude Variation**. In Fig. 1c (first row), letter "D" is with different contrast due to varied reflectance to infrared and visible light.
- **Gradient Direction Divergence**. In the second row, edge gradients are in opposite directions in the two images, causing structural deviation.
- **Gradient Loss**. In the last row, the characters are completely lost in the NIR image but not in the color one.
- **Shadow and highlight by flash**. If one uses flash only for the NIR image, it inevitably generates highlight/shadow that is not contained in the other image. Examples are presented later.

These issues are caused by inherent discrepancy of structures in the two different types of images. The algorithms to address them can be generally referred to as *multispectral image restoration*. Simple joint image filtering [12], [25] could blur weak edges, not to mention the inherent smoothing property. Directly transferring guidance gradients to the noisy field also results in notable appearance change.

- *X. Shen and J. Jia are with the Department of Computer Science and Engineering, The Chinese University of Hong Kong. Shatin, Hong Kong. E-mail: {xyshen, leojia}@cse.cuhk.edu.hk.*
- *L. Ma is with the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, China. E-mail: ma-lz@cs.sjtu.edu.cn.*
- *Q. Yan and L. Xu are with Lenovo R&T.*

Manuscript received 3 Mar. 2014; revised 19 Dec. 2014; accepted 10 Mar. 2015. Date of publication 7 Apr. 2015; date of current version 6 Nov. 2015. Recommended for acceptance by K. Lee.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TPAMI.2015.2417569

Fig. 1. Appearance comparison of RGB and NIR images. (a) RGB image. (b) Corresponding NIR image. (c) Close-ups. The four columns are for channels R, G, B, and NIR.

In this paper, we propose a framework via novel scale map construction. This map captures the nature of structure discrepancy between images and has clear statistical and numerical meaning. Based on its analysis, we design functions to form an optimal scale map considering adaptive smoothing, edge preservation and guidance strength manipulation. All aforementioned multispectral issues are discussed and addressed in this framework. We also develop an effective and fast solver via advanced robust function approximation and problem decomposition, which converges in less than five passes compared to other solutions that may need tens or hundreds of iterations. For high robustness, we also develop a multispectral shadow detection algorithm. Good quality results are produced.

This manuscript extends its conference version [29] with the following differences. First, we provide more analysis and understanding of the scale map in solving the multispectral image restoration problem. Second, a robust joint shadow detection method is developed. Third, we propose a multispectral image restoration framework considering shadow influence. Finally, more restoration applications are presented.

The rest of the paper is organized as follows. Section 2 describes modeling and formulation of multispectral joint image restoration. A multispectral shadow detection method is elaborated on in Section 3. In Section 4, we introduce an efficient numerical solution for the framework. We conduct experiments and present several applications in Section 5.

2 Modeling and Formulation

Our system takes the input of a noisy RGB image I_0 and a guidance image G captured from the same camera position. G can be a dark-flashed NIR image or others with possible structure variation as discussed above. Other multispectral configurations are allowed in our framework, presented in Section 5. Pixel values in each channel of the images are scaled to $[0, 1]$. G and I_0 could have different number of channels. Our goal is to recover an image from I_0 with noise removed and structure retained under guidance G. We process all channels separately.

Fig. 2. Optimal scale map s computed from images in Fig. 1 according to Eq. (1). Dark to bright pixels correspond to negative to positive values in different scales.

With aforementioned issues, using G to guide restoration is non-trivial. We introduce an auxiliary map s with the same size as G, which is key to our method, to adapt structure of G to that of I^*—the expected ground truth noise-free image. The s map is defined with relation

$$\min \|\nabla I^* - s \cdot \nabla G\|. \tag{1}$$

Here ∇ is an operator forming a vector with x- and y-direction gradients. Each element s_i in map s, where i indexes pixels, is a scalar, measuring robust difference between corresponding gradients in the two images. Simply put, s is a scale or ratio map between the guidance and ground truth. The optimal s corresponding to the multispectral example in Fig. 1 is shown in Fig. 2, visualized as a color image after pixel-wise value normalization to [0,1].

We analyze the unique properties of s with regard to structure discrepancy between ∇G and ∇I^*, and present them as follows with the facilitation of illustration in Fig. 3.

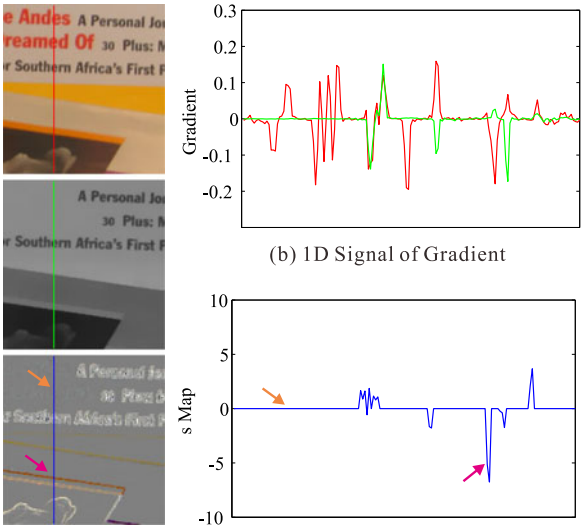
Property of s. First, sign of each s_i can be either positive or negative. A negative s_i means the ground truth edge exists also in the guidance, but edge direction is reverted, as demonstrated in Fig. 3c. Second, when the guidance image G contains extra shadow and highlight caused by flash, which do not exist in ∇I^*, s_i with value 0 can help ignore them. Finally, s_i can be any value when $\nabla G_i = 0$. That is, guidance edge does not exist (the red letters in Fig. 3a). Under local smoothness, making $s_i = 0$ is a good choice.

In short, if the s map is constructed optimally, all structure discrepancy problems can be addressed. We will discuss it more in Section 4.2. This map is first-of-a-kind to avail solving this set of multispectral restoration problems. Its additional and notable benefit is the role as latent variables in developing an efficient optimization procedure.

More of the function. Denoting by I our estimate towards I^*, Eq. (1) is updated to

$$\min \|\nabla I - s \cdot \nabla G\|. \tag{2}$$

As it involves unknowns ∇I and s, which correlate, the function is ill-posed. We take its variation as a possible data



(a) 2D Images (b) 1D Signal of Gradient (c) 1D Signal of s Map

Fig. 3. 1D illustration. (a) contains patches in the color image, NIR image and s map. Plot (b) contains gradients along the vertical line in the top two patches. (c) shows corresponding s values. Most of them are zeros; positive and negative values are also allowed.

term expression, together with regularization on s, to construct an objective function.

2.1 Data Term about s

In $|s_i \nabla G_i - \nabla I_i|$, where i indexes pixels, ∇G_i can be analogously regarded as a scale map for s_i due to the dual relation between s_i and ∇G_i. It controls the penalty when computing s_i for different pixels. The final cost resulted from $|s_i \nabla G_i - \nabla I_i|$ is dependent of the value of ∇G_i. For example, if ∇G_i and ∇I_i are doubled simultaneously, although s remains the same, the cost from $|s_i \nabla G_i - \nabla I_i|$ will get twice larger.

To stabilize costs w.r.t. s_i, we perform normalization

$$\sum_i \left| s_i - \frac{\nabla_x I_i}{\nabla_x G_i} \right| + \left| s_i - \frac{\nabla_y I_i}{\nabla_y G_i} \right|, \tag{3}$$

by dividing the two components of ∇G_i respectively. It removes the unexpected scaling effect caused by ∇G_i. Further to avoid the extreme situation when $\nabla_x G_i$ or $\nabla_y G_i$ is close to zero, and enlist the ability to reject outliers, we define our data term as

$$E_1(s, I) = \sum_i \Big(\rho(|s_i - p_{i,x} \nabla_x I_i|) + \rho(|s_i - p_{i,y} \nabla_y I_i|) \Big), \tag{4}$$

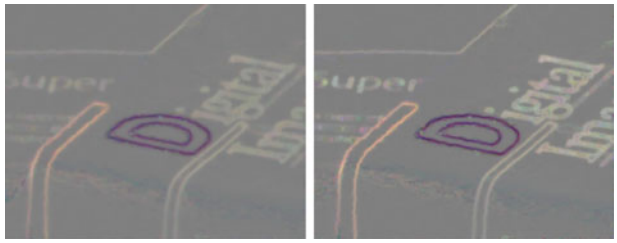
where ρ is a robust function defined as

$$\rho(x) = |x|^\alpha, 0 < \alpha < 1. \tag{5}$$

It is used to remove estimation outliers. We set $\alpha = 0.9$ in experiments. $p_{i,k}$, where $k \in \{x, y\}$, is a truncation function

$$p_{i,k} = \frac{1}{\text{sign}(\nabla_k G_i) \cdot \max(|\nabla_k G_i|, \varepsilon)}, \tag{6}$$

where $\text{sign}(x)$ is the sign operator, outputting 1 if $\nabla_k G_i$ is positive or zero and outputting -1 otherwise. $\max(|\nabla_k G_i|, \varepsilon)$ returns the larger value between $|\nabla_k G_i|$ and ε. The threshold ε is used to avoid division by zero and is set to 0.004 empirically.



(a) Isotropic Smoothing (b) Anisotropic Smoothing

Fig. 4. Isotropic versus anisotropic smoothing of the s map. Result in (b) resulted from anisotropic smoothing contains clearer and higher contrast structure. The input images are shown in Figs. 7a and 7b.

2.2 Data Term for I

The data term for I is simply set as

$$E_2(I) = \sum_i \rho(|I_i - I_{0,i}|), \tag{7}$$

where ρ is the same robust function and $I_{0,i}$ is the color of pixel i in I_0. $E_2(I)$ requires the restoration result not to wildly deviate from the input noisy image I_0 especially in important and salient edge areas. The robust function ρ helps reject part of the noise from I_0.

2.3 Regularization Term

Our regularization term is defined with anisotropic gradient tensors [4], [19]. It is based on the fact that s values are locally similar only in certain directions. For instance, s values should change smoothly or be constant along an edge more than those across it. As shown in Fig. 4, uniformly smoothing s in all directions blurs sharp edges.

Our anisotropic tensor scheme is able to preserve sharp edges according to gradient directions of G. After a few algebraic operations, an anisotropic tensor is expressed as

$$D(\nabla G_i) = \frac{1}{(\nabla G_i)^2 + 2\eta^2} \left((\nabla G_i^\perp)(\nabla G_i^\perp)^T + \eta^2 \mathbf{1} \right), \tag{8}$$

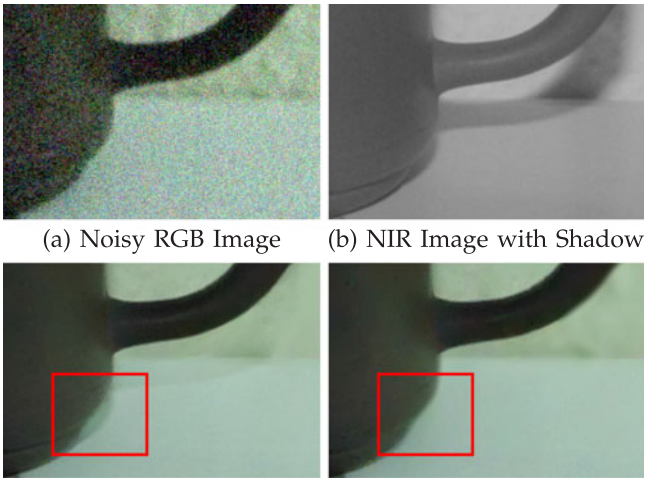
where $\nabla G_i^\perp = (\nabla_y G_i, -\nabla_x G_i)^T$ is a vector perpendicular to ∇G_i, $\mathbf{1}$ is an identity matrix and scalar η controls the isotropic smoothness. When ∇G_i is much smaller than η, Eq. (8) reduces to $0.5 \cdot \mathbf{1}$ and the structure tensor is therefore isotropic. In all our experiments, η is set to 0.1 empirically.

Generally, the two orthogonal eigenvectors of $D(\nabla G_i)$ are

$$\mathbf{v}_{i,1} = \frac{\nabla G_i}{|\nabla G_i|}, \quad \mathbf{v}_{i,2} = \frac{\nabla G_i^\perp}{|\nabla G_i|}, \tag{9}$$

with corresponding eigenvalues

$$\mu_{i,1} = \frac{\eta^2}{(\nabla G_i)^2 + 2\eta^2}, \quad \mu_{i,2} = \frac{(\nabla G_i)^2 + \eta^2}{(\nabla G_i)^2 + 2\eta^2}. \tag{10}$$



(a) Noisy RGB Image (b) NIR Image with Shadow

(c) No Shadow Treatment (d) Shadow Treatment

Fig. 5. A shadow handling example. (a) is a high-noise-level RGB image and (b) is the corresponding NIR image with shadow. (c) is the result restored without shadow handling while (d) is the result considering shadow.

This decomposes the tensor to

$$D(\nabla G_i) = (\,\mathbf{v_{i,1}} \quad \mathbf{v}_{i,2}\,)\begin{pmatrix}\mu_{i,1} & 0\\ 0 & \mu_{i,2}\end{pmatrix}\begin{pmatrix}\mathbf{v}_{i,1}^T\\ \mathbf{v}_{i,2}^T\end{pmatrix}. \tag{11}$$

This form makes it possible to express regularization for each ∇s_i as

$$E_3(\nabla s_i) = \mu_{i,1}\big(\mathbf{v}_{i,1}^T\nabla s_i\big)^2 + \mu_{i,2}\big(\mathbf{v}_{i,2}^T\nabla s_i\big)^2. \tag{12}$$

Different smoothing penalties are controlled by $\mu_{i,1}$ and $\mu_{i,2}$ in directions $\mathbf{v}_{i,1}$ and $\mathbf{v}_{i,2}$, across and along edges respectively. Stronger smoothness is naturally imposed along edges. The final smoothing term is thus defined as

$$E_3(\nabla s) = \sum_i \Big(\mu_{i,1}(\mathbf{v}_{i,1}^T\nabla s_i)^2 + \mu_{i,2}(\mathbf{v}_{i,2}^T\nabla s_i)^2\Big). \tag{13}$$

2.4 Objective Function without Special Shadow Treatment

The objective function involving the above terms for estimating the s map and restoring image I is written as

$$E(s, I) = E_1(s, I) + \lambda E_2(I) + \beta E_3(\nabla s), \tag{14}$$

where λ controls the confidence on noisy image I_0, and β corresponds to smoothness of s.

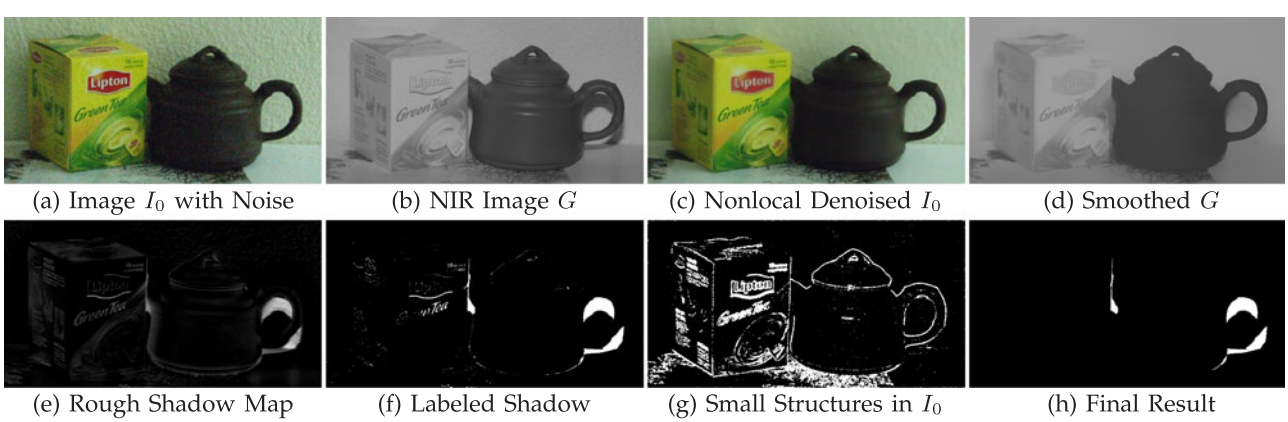
This function with the scale map s is already sufficiently powerful to produce reasonable restoration results for many data. However, when strong shadow exists in only one image and the noise level is high, the inherent structure discrepancy still brings trouble due to the possibly affected s estimation. An example is shown in Fig. 5.

3 Multispectral Shadow Detection

Many shadow detection methods [9], [11], [17], [26], [27] exist for color images. Employing these methods for multispectral images face challenges since the input can be gray-scale images. There are methods using additional color information to help detect shadow. Zhuo et al. [32] detected shadow in the NIR image only using the intensity difference in NIR and color images and finally set threshold to get the shadow mask. Rüenacht et al. [22] presented a method to detect shadow appearing in both the NIR and color images. These methods rely heavily on the intensity values and assume dark regions are more likely to be shadow. Because noise could be strong and shadow regions in NIR images are not always dark, shadow detection in multispectral images is hardly satisfactory. We propose a method in this section to deal with this problem.

To illustrate the difficulty in multispectral shadow detection and show how our framework works, the multispectral RGB/NIR image pair in Fig. 6 is used. The noisy RGB image in (a) is captured under ambient light with a high ISO while the NIR image in (b) is obtained with NIR flash, which reduces noise. But unwanted shadow in the NIR image is produced.

Our multispectral shadow detection algorithm is depicted in Algorithm 1. The main idea is to take the slightly de-noised RGB image as a guidance one to back filter the NIR image. This step is different from common restoration where structures in the NIR image are used as the aiding information. It can effectively find shadow because most edges in the two images are coincident and strong shadow inevitably yields notable discrepancy when marking structure difference. We describe our method below.



(a) Image I_0 with Noise (b) NIR Image G (c) Nonlocal Denoised I_0 (d) Smoothed G

(e) Rough Shadow Map (f) Labeled Shadow (g) Small Structures in I_0 (h) Final Result

Fig. 6. Multispectral shadow detection. Given noisy RGB image (a) and NIR image (b), our algorithm detects shadow regions in (b). (c) is the denoising result of (a) with the help of (b). (d) is the shadow removed and smoothed result of (b). (e) is the rough shadow map while (f) is obtained by applying labeling to (e). (g) is the exacted small structures of I_0 and (h) shows the final shadow detection result.

Algorithm 1. Multispectral Shadow Detection

1: **input:** multispectral image pair (noisy I_0 and clean G);
2: denoise I_0 with the guidance G by Eq. (15);
3: smooth G by Eq. (16);
4: get rough shadow map S by Eq. (18);
5: label shadow by computing Eq. (19);
6: final process by DoG and median filter;
7: **output:** shadow map.

3.1 Nonlocal Image Smoothing

We first weakly denoise the RGB image by nonlocal smoothing. The objective function is defined as

$$E(I') = \sum_p \left\|I'_p - I_{0,p}\right\|^2 + \lambda_1 \sum_{q\in N(p)} \omega_{p,q} \left\|I'_p - I'_q\right\|^2, \qquad (15)$$

where I_0 is the input noisy image and I' is our target. $N(p)$ is the neighborhood of pixel p. λ_1 is the weight to control smoothness. $\omega_{p,q}$ is the affinity weight between p and q, which is defined as

$$\omega_{p,q} = \exp\left(-\frac{\|G_p - G_q\|^2}{\sigma_G^2} - \frac{\|p-q\|^2}{\sigma_x^2}\right)$$

like bilateral filter [4]. It helps keep edges and remove noise. σ_G and σ_x are two parameters to control spatial and range influence.

This function slightly smoothes I_0 aware of inherent structure. It would not strongly damage edges. The nonlocal image smoothing can be efficiently solved iteratively by fast bilateral filtering [18]. In our experiments, we set λ_1 to 0.01. σ_G and σ_x are set to 0.01 and 6 respectively. Size of $N(p)$ is set to $(2\sigma_x + 1) \times (2\sigma_x + 1)$. One initial denoising result is shown in Fig. 6c.

3.2 Shadow Map Detection

Our shadow detection uses the denoised I' to guide NIR image G smoothing. It is particularly suitable and effective because I' does not contain any flash shadow information and thus can naturally eliminate shadow in the NIR image without assuming dark-pixel or light-direction prior knowledge.

Our joint shadow removal is achieved by solving the function similar to Eq. (15), written as

$$E(G') = \sum_p \left\|G'_p - G_p\right\|^2 + \lambda_2 \sum_{q\in N'(p)} \omega'_{p,q} \left\|G'_p - G'_q\right\|^2, \qquad (16)$$

$\omega'_{p,q}$ is the affinity again taking the denoised I' as the guidance. It is expressed as

$$\omega'_{p,q} = \exp\left(-\frac{\|I'_p - I'_q\|^2}{\sigma_I^2} - \frac{\|p-q\|^2}{\sigma_x'^2}\right). \qquad (17)$$

Obviously, if shadow is not presented at one pixel in I', smoothing will not stop at it in the NIR image. This effect results in automatic shadow removal in the filtered G'. Setting of σ'_x relates to the shadow region size. It is not sensitive. The size of $N'(p)$ is twice of σ'_x. Both σ_I and λ_2 are set to 0.01 in our experiments.

In general, intensity change in G' in shadow regions is much larger than others. An example is shown in Fig. 6d. The difference between G and G' primarily represent shadow in many of our experiments. Therefore, we compute

$$S = \max(0, G' - G), \qquad (18)$$

as the rough shadow map. $\max(.,.)$ is the max function to return the larger value. As smoothing shadow increases intensity values, a positive $G' - G$ becomes a good indication.

Note this map may not be accurate and possibly contains several errors, as illustrated in Fig. 6e. Problems in S mainly originate from two aspects. First, the guidance I' still contains a level of noise since our denoising process is not strong in order not to damage underlying structures. Second, small structures of G are smoothed out. We remedy them in the next two steps.

Shadow labeling. We refine the shadow map S by graph cuts. The graph is set as $\mathcal{G} = \langle \mathcal{V}, \mathcal{E} \rangle$, where $\mathcal{V}$ is the set of all nodes and $\mathcal{E}$ is the set of all edges connecting adjacent nodes. Here, the nodes are pixels, and the edges depict the adjacency relationship under four- or eight-connection between neighborhood pixels. The energy is defined on $\mathcal{G}$ as

$$E(X) = \sum_{i\in\mathcal{V}} E_D(x_i) + \lambda_s \sum_{(i,j)\in\mathcal{E}} E_S(x_i, x_j), \qquad (19)$$

where x_i is the binary label with 1 denoting shadow pixels and 0 for others. $E_D(x_i)$ is the likelihood cost when the label of node i is x_i. $E_S(x_i, x_j)$ denotes the cost when the labels of adjacent nodes i and j are x_i and x_j respectively. $E_D(x_i)$ is defined as

$$\begin{cases} E_D(x_i) = (1-x_i)\cdot K & \forall i \in \mathcal{S} \\ E_D(x_i) = x_i \cdot K & \forall i \in \mathcal{N} \\ E_D(x_i) = (1-x_i)\beta_s |S_i - F_c| + x_i |S_i - B_c| & \forall i \in \mathcal{U}, \end{cases}$$

where $\mathcal{S}$ and $\mathcal{N}$ denote shadow and non-shadow pixel seeds. In our experiments, $\mathcal{S}$ and $\mathcal{N}$ contain the top 2 percent and bottom 20 percent pixels with highest and lowest shadow values respectively based on the fact that shadow generally occupies a small portion of area in the whole image. K is a sufficiently large constant.

$\mathcal{U} = \mathcal{V} \backslash (\mathcal{S} \cup \mathcal{N})$ is the set of uncertain pixels. For these pixels, we do not know whether they are shadow or not. Thus, we set their respective cost assigned to shadow and non-shadow as the difference of shadow costs, i.e., $|S_i - F_c|$ and $|S_i - B_c|$ where F_c and B_c are center shadow map value in $\mathcal{S}$ and $\mathcal{N}$ respectively. β_s is a parameter decided by the relative difference between shadow and non-shadow regions. Setting β_s to 10 is good enough in our experiments.

The regularization term $E_S(x_i, x_j)$ imposing smoothness between labels x_i and x_j is defined as

$$E_S(x_i, x_j) = |x_i - x_j| \cdot \exp(|S_i - S_j|). \qquad (20)$$

The shadow labeling result is obtained by optimizing Eq. (19) by graph cuts. One result is shown in Fig. 6f.

Final operations. Now the shadow map is with good quality except for a few small regions still mistakenly detected as shadow. It is because a few edges, such as the character strokes in Fig. 6b, are removed from G during smoothing, as shown in (d). This produces difference

when comparing G with its smoothed version even if pixels are not shadow ones.

To redress this problem, we explicitly extract small structures in I_0 by Difference of Gaussian, as illustrated in Fig. 6g. This simple operation is fast to separate out most small structures. Then we apply the relative complement of (g) in (f) to keep non-zero pixels in (f) when they are zeros in (g). This process can remove more than 98 percent of the errors in our experiments since nearly all small regions not belonging to shadow are eliminated. Note small shadow regions could also be discarded. It is acceptable in all our experiments because small shadow would not influence much the quality of final restored images.

Median filtering is applied afterwards to remove isolated noise. We denote the final shadow detection result as S_r, shown in Fig. 6h.

3.3 Final Objective Function Considering Shadow

After we get the shadow map S_r, it is straightforward to update the energy function. We exclude shadow pixels in G in the data term in Eq. (4) and employ single image restoration separately for these pixels in the RGB image. Eq. (4) is therefore updated to

$$E_1(s,I)=\sum_i(1-S_{r,i})(\rho(|s_i-p_{i,x}\nabla_x I_i|)+\rho(|s_i-p_{i,y}\nabla_y I_i|)), \tag{21}$$

where $S_{r,i}$ is the shadow map value of pixel i.

For shadow pixels, because the NIR image does not provide correct guidance, we employ single-image denoising on I_0 by BM3D [6]. Its result for these shadow pixels is denoted as $\tilde{I}$. To make the objective function complete, we introduce a new term requiring that the resulting shadow regions are the same as those in $\tilde{I}$. This term is written as

$$E_4(I)=\sum_i S_{r,i}(\nabla I_i-\nabla\tilde{I}_i)^2, \tag{22}$$

The final objective function $E_s(s,I)$ considering shadow is expressed as

$$E_s(s,I)=E_1(s,I)+\lambda E_2(I)+\beta E_3(\nabla s)+\gamma E_4(I), \tag{23}$$

where λ and β are the same as those in Eq. (14). γ controls the influence of data term $E_4(I)$. In Section 5, we discuss how to set these parameters.

Note when G does not contain extra shadow, all pixels in S_r have value 0. Eq. (23) is thus almost the same as Eq. (14). Small edge and structure discrepancy is naturally ignored in the shadow map, as illustrated in Fig. 6. Close-up of the result is shown in Fig. 5d.

This objective function is non-convex due to the involvement of sparsity terms. Joint representation for s and I in optimization further complicates the problem. Naively solving it by simple gradient decent cannot guarantee optimality and leads to slow convergence. We propose an iterative method, which finds constraints sufficient for shaping the s map according to its ideal characteristics and having the effect to remove intensive noise from the input I_0.

4 Numerical Solution

To solve the non-convex function $E_s(s,I)$ defined in Eq. (23), we employ the iterative reweighted least squares (IRLS) scheme [16], which converts the original problem to solving a few corresponding linear systems without losing the original properties. This process is not trivial however.

Initially, robust function $\rho(x)$ in Eq. (5) for any scalar x is written as $x^2/|x|^{2-\alpha}$, further approximated as

$$\rho(x)\approx\phi(x)\cdot x^2, \tag{24}$$

where $\phi(x)$ is defined as

$$\phi(x)=\frac{1}{|x|^{2-\alpha}+\epsilon}. \tag{25}$$

ϵ is a small number to avoid division by 0. We set it to $1e-4$ empirically. This form enables splitting the robust function into two parts where $\phi(x)$ can be regarded as a weight for x^2. In our method, following tradition of IRLS, $\phi(x)$ and x^2 are updated alternatively because they form simpler representation together with other necessary terms to profit optimization.

Vector form. To ease derivation, we re-write Eq. (14) in the vector form by taking the expression in Eq. (24) into computation. It yields

$$\begin{aligned}E(\mathbf{s},\mathbf{I})=&(\mathbf{s}-P_xC_x\mathbf{I})^TA_x(\mathbf{s}-P_xC_x\mathbf{I})\\&+(\mathbf{s}-P_yC_y\mathbf{I})^TA_y(\mathbf{s}-P_yC_y\mathbf{I})\\&+\lambda(\mathbf{I}-\mathbf{I}_0)^TB(\mathbf{I}-\mathbf{I}_0)+\beta\mathbf{s}^TL\mathbf{s}\\&+\gamma(\mathbf{I}-\mathbf{I}')^TD(\mathbf{I}-\mathbf{I}'),\end{aligned} \tag{26}$$

where $\mathbf{s}$, $\mathbf{I}$, $\mathbf{I}_0$ and $\mathbf{I}'$ are vector representations of s, I, I_0 and I'. C_x and C_y are discrete backward difference matrices that are used to compute image gradients in the $x-$ and $y-$directions. P_x, P_y, A_x, A_y and B are diagonal matrices, whose ith diagonal elements are

$$\begin{aligned}(P_x)_{ii}&=p_{i,x},\quad (A_x)_{ii}=\phi(s_i-p_{i,x}\nabla_xI_i)(1-S_{r,i}),\\(P_y)_{ii}&=p_{i,y},\quad (A_y)_{ii}=\phi(s_i-p_{i,y}\nabla_yI_i)(1-S_{r,i}),\\B_{ii}&=\phi(I_i-I_{0,i}).\end{aligned}$$

Note A_x, A_y and B account for the re-weighting process and are typically computed using estimates from previous iterations; P_x and P_y are normalization terms from the guided input. The first three terms in Eq. (27) correspond to E_1 and E_2; $\mathbf{s}^TL\mathbf{s}$ is created by E_3 and the last one is the vector form of E_4.

The term $\mathbf{s}^TL\mathbf{s}$ controls the spatial smoothness of $\mathbf{s}$, where matrix L is a smoothing Laplacian, nontrivially expressed as

$$\begin{aligned}L=&C_x^T\left(\Sigma_1V_x^2+\Sigma_2V_y^2\right)C_x+C_y^T\left(\Sigma_2V_x^2+\Sigma_1V_y^2\right)C_y\\&+2C_y^T(\Sigma_1-\Sigma_2)V_xV_yC_x\end{aligned} \tag{27}$$

after derivations. Σ_1, Σ_2, V_x and V_y are all diagonal matrices. Their ith diagonal elements are

$$\begin{aligned}(\Sigma_1)_{ii}&=\mu_{i,1},\quad (V_x)_{ii}=\nabla_xG_i/\max(|\nabla G_i|,\varepsilon),\\(\Sigma_2)_{ii}&=\mu_{i,2},\quad (V_y)_{ii}=\nabla_yG_i/\max(|\nabla G_i|,\varepsilon).\end{aligned}$$

The last term $(\mathbf{I}-\mathbf{I}')^T D(\mathbf{I}-\mathbf{I}')$ is the vector form of $E_4(I)$ with $D = C_x^T \mathbf{S}_r C_x + C_y^T \mathbf{S}_r C_y$. $\mathbf{S}_r$ is a diagonal matrix whose ith diagonal element is $S_{r,i}$. This term helps restore I in shadow regions with the constraint of $\tilde{I}$ in gradient domain.

Analysis. L is actually an inhomogeneous term, reflecting the anisotropic property of our smoothing regularizer. To understand it, we consider the extreme case that ∇G approaches zero. It leads to $\Sigma_1 = \Sigma_2$ and $V_x = V_y = 0$, making L a homogenous Laplacian. The resulting s map is therefore smooth equally in all directions. But on natural images, ∇G on an edge is not isotropic, which enlists different regularization strength. Also, since C_x and C_y are sparse, the Laplacian matrix L is also sparse that facilitates system solving given that many mature sparse matrix solvers exist.

Algorithm 2. Multispectral Image Restoration

1 **input:** noisy image I_0, guidance image G, parameters β and λ;
2 initialize $I \leftarrow I_0$, $s \leftarrow 1$;
3 **repeat**
4 estimate s according to Eq. (30);
5 estimate I according to Eq. (32);
6 **until** convergence
7 **output:** s map and restored image I.

4.1 Solver

We propose an alternating minimization algorithm to solve for $\mathbf{s}$ and $\mathbf{I}$ based on above derivations. Results of $\mathbf{s}$ and $\mathbf{I}$ in each iteration t are denoted as $\mathbf{s}^{(t)}$ and $\mathbf{I}^{(t)}$. Initially, we set $\mathbf{s}^{(0)} = \mathbf{1}$, whose elements are all 1 and $\mathbf{I}^{(0)} = \mathbf{I}_0$.

By setting all initial s_i to 1, total smoothness is obtained. It yields zero cost for $E_3(s)$, a nice starting point for optimization. This initialization also makes the starting ∇I same as ∇G with many details. Then at iteration $t+1$, we solve two subproblems sequentially

- Given $\mathbf{s}^{(t)}$ and $\mathbf{I}^{(t)}$, minimize $E(\mathbf{s}, \mathbf{I}^{(t)})$ to get $\mathbf{s}^{(t+1)}$.
- Given $\mathbf{s}^{(t+1)}$ and $\mathbf{I}^{(t)}$, minimize $E(\mathbf{s}^{(t+1)}, \mathbf{I})$ to update $\mathbf{I}^{(t+1)}$.

The procedure is repeated until $\mathbf{s}$ and $\mathbf{I}$ do not change too much. Usually, a small number of iterations (4-6) are enough to generate visually compelling results. The algorithm is depicted in Algorithm 2, with the solvers elaborated on below.

Solve for $\mathbf{s}^{(t+1)}$. The energy function with respect to $\mathbf{s}$ can be expressed as

$$E(\mathbf{s}) = (\mathbf{s} - P_x C_x \mathbf{I})^T A_x (\mathbf{s} - P_x C_x \mathbf{I}) + (\mathbf{s} - P_y C_y \mathbf{I})^T A_y (\mathbf{s} - P_y C_y \mathbf{I}) + \beta \mathbf{s}^T L \mathbf{s}. \quad (28)$$

Computation of A_x and A_y depends on estimates $\mathbf{s}$ and $\mathbf{I}$ from the previous iteration. We denote by $A_x^{t,t}$ and $A_y^{t,t}$ the matrices computed with $\mathbf{s}^{(t)}$ and $\mathbf{I}^{(t)}$, which lead to

$$\tilde{E}(\mathbf{s}) = (\mathbf{s} - P_x C_x \mathbf{I}^{(t)})^T A_x^{t,t} (\mathbf{s} - P_x C_x \mathbf{I}^{(t)}) + (\mathbf{s} - P_y C_y \mathbf{I}^{(t)})^T A_y^{t,t} (\mathbf{s} - P_y C_y \mathbf{I}^{(t)}) + \beta \mathbf{s}^T L \mathbf{s}. \quad (29)$$

It is simply quadratic. Taking derivatives on $\mathbf{s}$ and setting them to 0, we obtain the sparse linear system

$$\left(A_x^{t,t} + A_y^{t,t} + \beta L\right)\mathbf{s} = A_x^{t,t} P_x C_x \mathbf{I}^{(t)} + A_y^{t,t} P_y C_y \mathbf{I}^{(t)}. \quad (30)$$

We solve it by pre-conditioned conjugate gradient (PCG). The solution is denoted as $\mathbf{s}^{(t+1)}$.

Solve for $\mathbf{I}^{(t+1)}$. Similarly, the energy function to solve for $\mathbf{I}$ is given by

$$\tilde{E}(\mathbf{I}) = (\mathbf{s}^{(t+1)} - P_x C_x \mathbf{I})^T A_x^{t+1,t} (\mathbf{s}^{(t+1)} - P_x C_x \mathbf{I}) + (\mathbf{s}^{(t+1)} - P_y C_y \mathbf{I})^T A_y^{t+1,t} (\mathbf{s}^{(t+1)} - P_y C_y \mathbf{I}) + \lambda (\mathbf{I}-\mathbf{I}_0)^T B^{t+1,t} (\mathbf{I}-\mathbf{I}_0) + \gamma (\mathbf{I}-\mathbf{I}')^T D (\mathbf{I}-\mathbf{I}'), \quad (31)$$

where $A_x^{t+1,t}$ and $A_y^{t+1,t}$ are calculated with available $\mathbf{s}^{(t+1)}$ and $\mathbf{I}^{(t)}$. $B^{t+1,t}$ depends on $\mathbf{I}^{(t)}$. The final linear system in the matrix form is

$$\left(\left(C_x^T (P_x)^2 A_x^{t+1,t} C_x + C_y^T (P_y)^2 A_y^{t+1,t} C_y\right) + \lambda B^{t+1,t} + \gamma D\right)\mathbf{I} = \left(C_x^T P_x A_x^{t+1,t} + C_y^T P_y A_y^{t+1,t}\right)\mathbf{s} + \lambda B^{t+1,t} \mathbf{I}_0 + \gamma D \mathbf{I}'. \quad (32)$$

The linear system is also solved using PCG and the solution is denoted as $\mathbf{I}^{(t+1)}$. The steps are included in Algorithm 2.

4.2 Why Does It Work?

According to the linear system defined in Eq. (30), the resulting s_i for pixel i is a weighted average of $p_{i,x} \nabla_x I_i \approx \nabla_x I_i / \nabla_x G_i$ and $p_{i,y} \nabla_y I_i \approx \nabla_y I_i / \nabla_y G_i$, whose weights are determined by $(A_x)_{ii}$ and $(A_y)_{ii}$. Even if the two weights are quite different due to noise or other aforementioned issues in Section 1, our method can still find the optimum.

To understand it, we assume $p_{i,x} \nabla_x I_i$ is larger than the other term. In solving for I according to Eq. (32), s_i reduces the gradient in the x-direction and increases the other so that ∇I_i lies closer to $s \nabla G_i$. In the meantime noise is reduced. After each iteration, a less noisy I is fed into Eq. (30) to produce new values of $p_{i,x} \nabla_x I_i$ and $p_{i,x} \nabla_y I_i$, which become closer than those in previous iterations.

Eventually when the two estimates meet each other, s converges. I will accordingly be optimal when ∇I and $s \nabla G$ are close enough according to λ. Considering the smoothness term L in Eq. (30), it helps avoid discontinuities in the s map along edges of G.

We show in Fig. 7e the initial constant s map. Figs. 7f, 7g are maps produced in two iterations, and (h) shows the final s. Initially the map is noisy because of erroneous gradient magnitudes and directions in the input images. As of more iterations being taken, it becomes better regularized following our rules while not overly smoothed. Our final scale map adapts the gradient field of G to match that of I_0 with noise removed.

Difference from previous methods. Our method is particularly effective for multispectual joint image restoration. In [12], a guided image filter was proposed in the form of a local linear regression model, which involves estimating a parameter a_i for each pixel i. Although a_i is seemingly analogous to our scale map s_i, they are by nature different. a_i is estimated based on patch regression which may not be accurate when the input RGB image is with a high level of noise and inverse

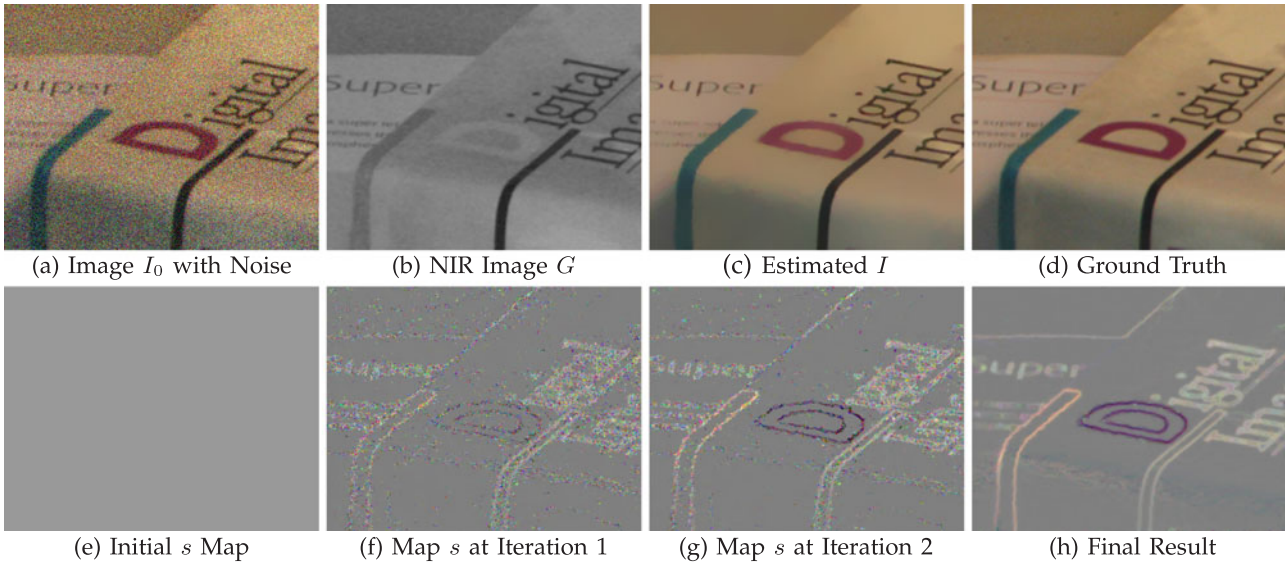


Fig. 7. s map estimation in iterations. Given image pairs in (a) and (b), our method can get the high-quality restoration result in (c). The s maps in different iterations are shown in (e)-(h).

gradients. One example is shown in Fig. 8. The result difference is obvious. Note our s_i is defined for each pixel with the robust data and regression terms optimized in a global function, which leads to the less-noise result.

Our method is also distinct from edge-aware filtering [1], [10], [25], which employs weighted mean. The new scale map is developed in gradient domain and considers possible relationship between ∇I and ∇G. We show the comparisons in Section 5.

5 Experiments and Applications

The two input images—one is noisy and the other is clean—are aligned. The two inputs can be a RGB/NIR pair, flash/no-flash ones, RGB/depth pair, or other multispectral images. We explain our algorithm on noisy RGB and flashed NIR images due to its generality of structure discrepancy and higher challenge than flash/no-flash images.

Experiment setting and running time. Our method has three parameters β, λ and γ. β and λ control smoothness of s and confidence of noisy input respectively, which are more important than γ that controls the blending result in shadow regions. We illustrate the influence of changing these two parameters in Fig. 9. It shows a large λ helps preserve structure close to the noisy RGB input. Smoother results can be obtained with a larger β. By default we set $\lambda = 5.0$ and $\beta = 0.5$. To handle shadow influence, γ is set from 50-100. We implement our method in MATLAB. The implementation takes about 20 seconds in our restoration step and 3 seconds for shadow detection when processing an image of resolution 800×600.

Multispectral shadow detection. As shown in Fig. 6, our multispectral shadow detection can handle regions that are not with the lowest intensities, which are challenging for other single or multiple image methods. Figs. 10c and 10d compare

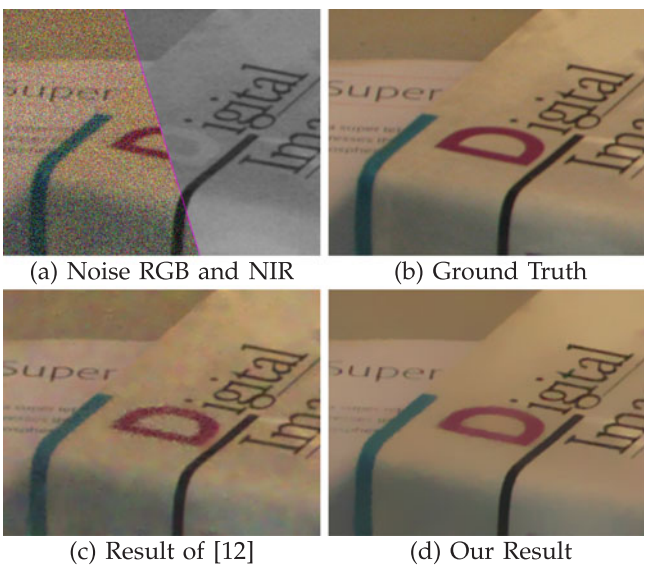


Fig. 8. Comparison with guided image filter [12]. (c) is the guided image filter result with $r = 6$ and $\epsilon = 0.03^2$. (d) is our result computed with $\lambda = 2.5$ and $\beta = 0.5$.

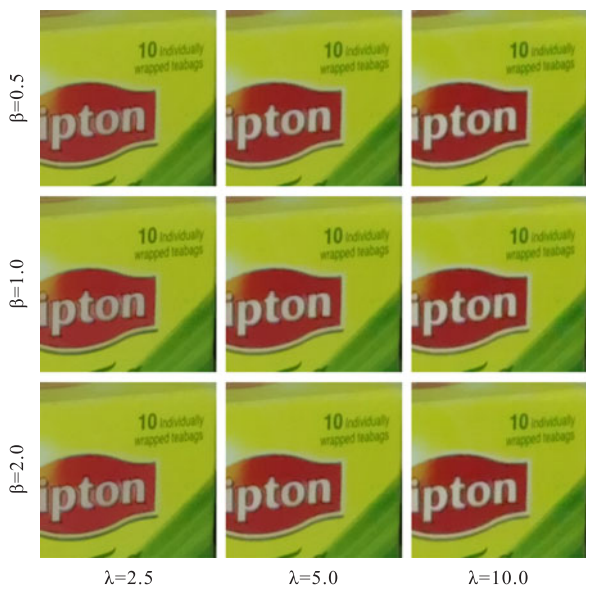


Fig. 9. Restoration results by varying parameters. The inputs are shown in Figs. 6a and 6b.

(a) Noisy Image (b) NIR Image (c) Shadow by [22] (d) Our Shadow (e) Ours Restored

Fig. 10. Shadow detection in multispectral restoration. Our shadow detection results in (d) are more accurate than those of [22] (shown in (c)). (e) are results of our method after shadow detection.

shadow detection results produced by the state-of-the-art RGB/NIR image method [22] and ours. These are challenging examples as the input images contain no color information. With reasonable shadow detection in the NIR image, our framework restores visually compelling results in (e).

Color and flashed NIR image restoration. In the example shown in Fig. 7, some gradients of the guiding NIR image are reverted or weak compared to those in the noisy color image. It is dealt with by getting negative values in the scale map s naturally in our framework.

Fig. 11 compares our result with those of state-of-the-art guided non-local means [5], guided image filtering [12], BM3D [6], and enhancement method [32]. Because generally previous work does not handle explicitly gradient variation, blurriness could be yielded. The result difference is obvious.

Experiments with images containing different levels of noise are conducted. In Fig. 12, We add different levels of noise to the ground truth RGB image and use the same NIR image to help restore it. The ground truth image pair is shown in the left of Fig. 12. PSNRs are calculated for quantitative evaluation. We compare our method with single image BM3D [6], non-local means [5] and guided image filter [12]. The PSNRs are plotted in Fig. 12. This set of experiments show multispectral image restoration is achievable and our framework is robust for it.

We also compare our result with the one presented in [15], which was generated by taking both UV and IR flashed image as guidance. Our method, by only regarding the red channel that records the IR light as G, achieves the comparable result shown in Fig. 13. More results will be put onto the project website.[1]

Flash and no-flash images. Our method is also applicable to image restoration using flash/no-flash image pairs. Since both the two images are color ones under visible light, we use each channel from the flash image to guide image restoration in the corresponding channel of the no-flash noisy image. A result with comparison to that of [20] is presented in Fig. 14. Overall, our recovered structures are sharp and clean. Gradient reversion in input images also happens in this configuration due to strong flash. Without handling it, it is hard to preserve sharp edges as gradients averaging to nearly zeros are commonly resulted in.

RGB and depth images. Depth images captured from Microsoft Kinect or other devices contain strong noise caused by missing or erroneous data for many pixels. We similarly apply our restoration method with the corresponding RGB images as guidance to refine raw depth data as shown in Fig. 15. Compared with result produced by [12], ours is with reasonable quality where background noise is mostly removed and edges are not blurred too much.

Texture smoothing. Image texture smoothing benefits structure extraction. Our framework provides a decent solution for texture smoothing. The result shown in Fig. 16c is obtained by setting the guidance image as constant. Compared with the result proposed by [28] (shown in Fig. 16b), ours preserves the similar level of structure while removing texture.

Multispectral dehazing. We also apply our method to multispectral dehazing with color and NIR images captured in haze. An image recovered from low visibility caused by haze suffers from noise and compression artifacts due to significant gradient enhancement in low contrast regions. The NIR correspondence however is a hardware solution to see more details in haze. By applying our method to single-image dehazed noisy result and the NIR input, we can much improve the quality. An example is shown in Fig. 17. The single-image dehazing result by [13] contains noise. The result of [23] however changes the original color. Our restoration result with a NIR image as guidance G is visually pleasing.

Day and night image enhancement. Our method is also applicable to day/night image enhancement. Images captured at night are usually dark with many structures eliminated. An example is shown in Fig. 18a. Brightening it by increasing color contrast in Photoshop improves the image. In the meantime, it boosts image noise, as shown in (c). With a reference day image as proposed in [8], we can remove the noise with high-quality edge preservation. The output (d) contains fine details and keeps strong and clean structures with the day image information.

1. http://www.cse.cuhk.edu.hk/leojia/projects/crossfield

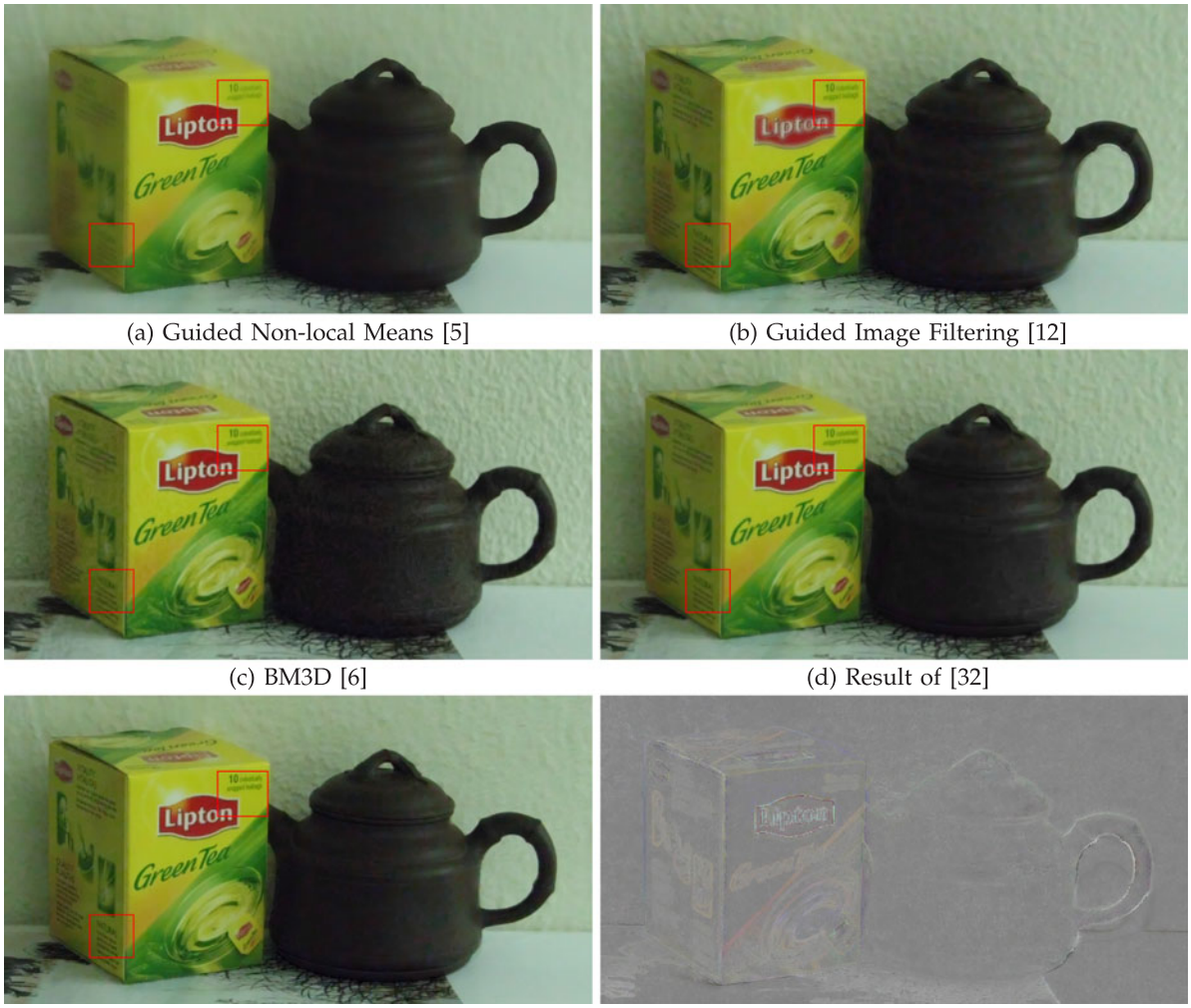


Fig. 11. Tea-bag example. The input RGB and NIR images are shown in Figs. 6a and 6b respectively. (a) is the result of guided non-local means [10]. (b) is denoised by the method of [12] with the NIR image as guidance ($r = 4$ and $\epsilon = 0.01^2$). (c) is the BM3D result while (d) is enhanced by the method of [32] ($\sigma = 35.0$). (e) is our result ($\lambda = 10.0$ and $\beta = 1.5$) and (f) shows the corresponding scale map.

6 Conclusion and Limitation

We have presented a complete system, showing a principled way for multispectral joint image restoration. Unlike transferring details or applying joint filtering, we explicitly take the possible structural discrepancy between input images into consideration. It is encoded in a scale map s involving all challenging cases to deal with. Our objective functions and optimization process are tailor made to use

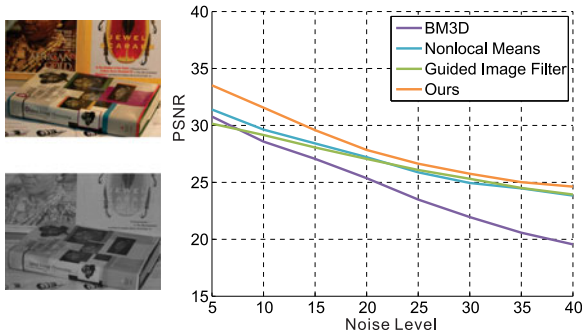


Fig. 12. Multispectral image restoration under different noise levels.

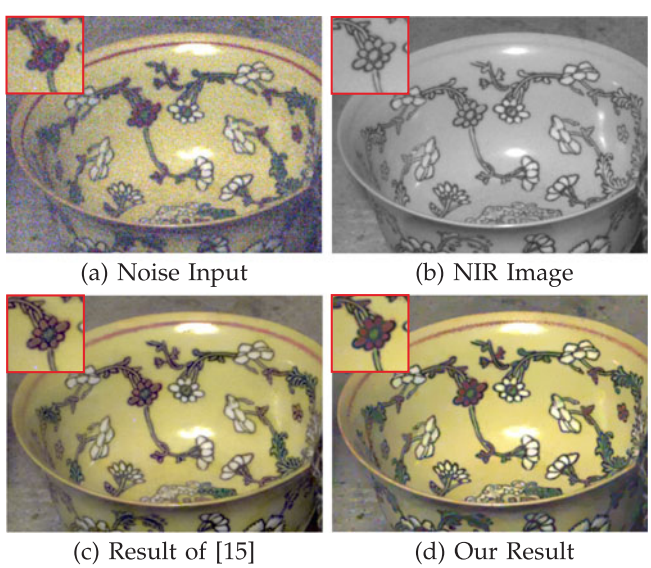


Fig. 13. RGB/flashed-NIR restoration. The input images and result of [15] are obtained from the original paper. Note the result of [15] is outputed from two guidance images while ours is with only the IR image as guidance. The parameters used to get our result are $\lambda = 5.0$ and $\beta = 0.5$.

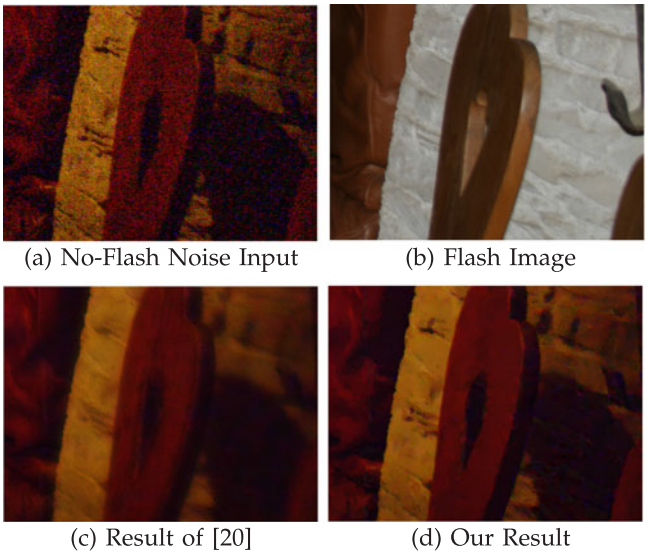


Fig. 14. Image restoration from flash/no-flash image pairs. The input images and result of [20] are obtained from the original paper. $\lambda = 6.0$ and $\beta = 0.6$ are used in our method.

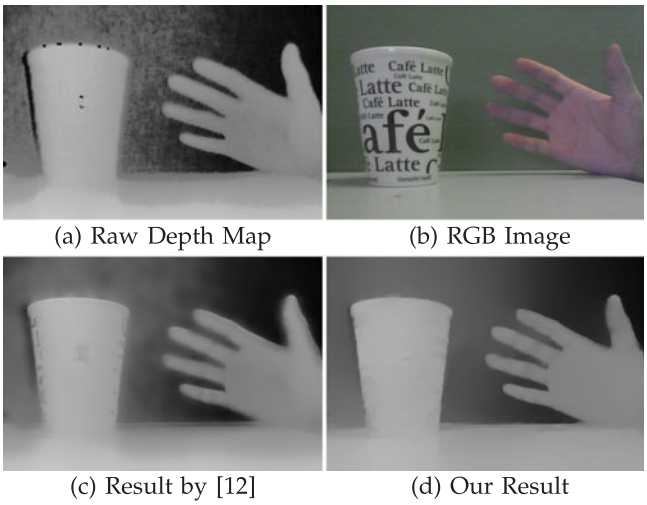


Fig. 15. Depth map restoration with RGB image as guidance. The parameter settings are $\lambda = 5.0$ and $\beta = 0.5$.

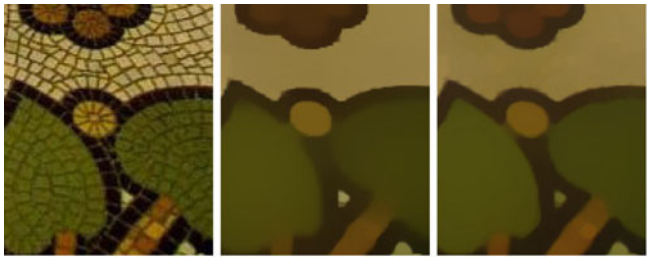


Fig. 16. Texture smoothing example. (b) is the result computed by [28] and (c) is our result computed with $\lambda = 5.0$ and $\beta = 1.0$.

the guidance from other domains and preserve only necessary details and edges.

The limitation of our current method is on the situation that the guidance does not exist, corresponding to zero ∇G and non-zero ∇I^* pixels. One example is shown in Fig. 19.

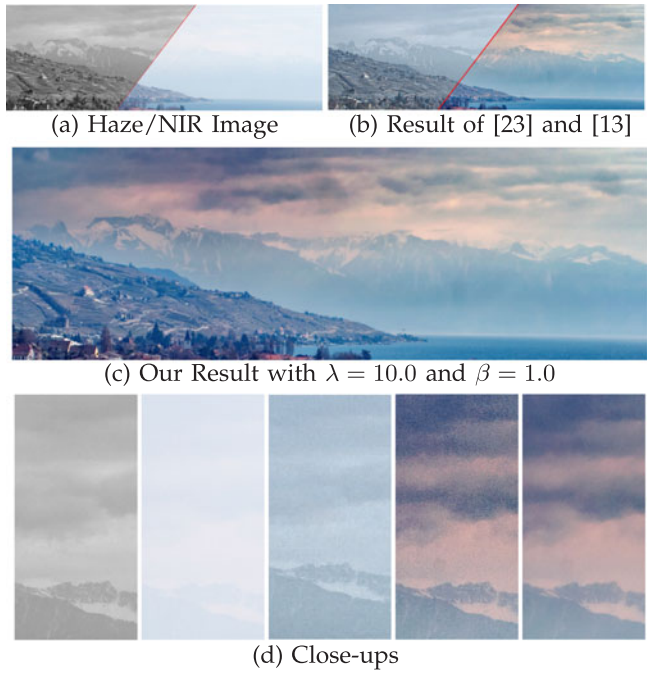


Fig. 17. Image restoration from haze image. Close-ups shown in (d) are from (a-c). The left two are NIR and haze images. The right three are results of [23], [13] and ours.

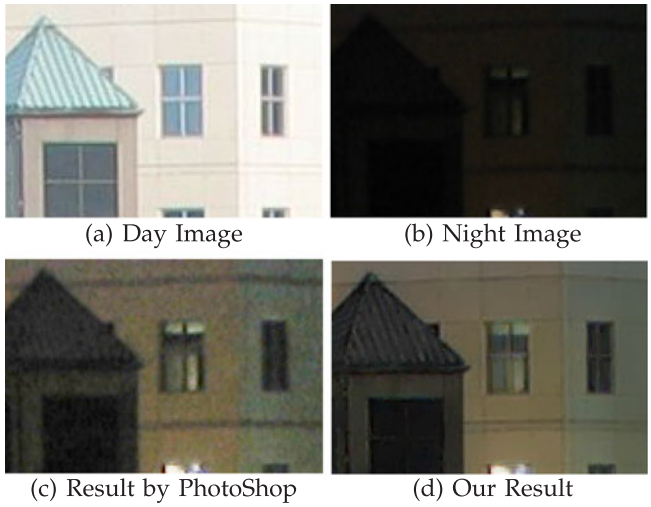


Fig. 18. Day and night image pair enhancement. (c) is from (b) by improving the color contrast in Photoshop. (d) is our result get from (c) in our framework using (a) as guidance. The parameters used to compute our result are $\lambda = 3.0$ and $\beta = 0.5$.

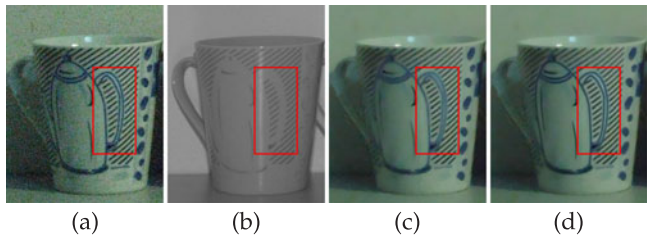


Fig. 19. Limitation. There is no guidance structure in the rectangle of (b), making restoration less-constrained. Our two-image restoration result is shown in (c). These regions can be further single-image denoised, as shown in (d).

Because the guidance does not exist, image restoration naturally degrades to single-image denoising.

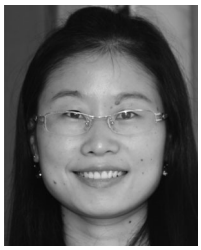
ACKNOWLEDGMENTS

The work described in this paper was supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region (Project No. 412911).

REFERENCES

[1] A. K. Agrawal, R. Raskar, S. K. Nayar, and Y. Li, "Removing photography artifacts using gradient projection and flash-exposure sampling," *ACM Trans. Graph.*, vol. 24, no. 3, pp. 828–835, 2005.
[2] D. Akers, F. Losasso, J. Klingner, M. Agrawala, J. Rick, and P. Hanrahan, "Conveying shape and features with image-based relighting," in *Proc. IEEE Vis.*, 2003, pp. 349–354.
[3] P. Bhat, C. L. Zitnick, M. F. Cohen, and B. Curless, "Gradientshop: A gradient-domain optimization framework for image and video filtering," *ACM Trans. Graph.*, vol. 29, no. 2, p. 10, 2010.
[4] M. J. Black, G. Sapiro, D. H. Marimont, and D. Heeger, "Robust anisotropic diffusion," *IEEE Trans. Image Process.*, vol. 7, no. 3, pp. 421–432, Mar. 1998.
[5] A. Buades, B. Coll, and J.-M. Morel, "A non-local algorithm for image denoising," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2005, pp. 60–65.
[6] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, "Image denoising by sparse 3-d transform-domain collaborative filtering," *IEEE Trans. Image Process.*, vol. 16, no. 8, pp. 2080–2095, Aug. 2007.
[7] E. Eisemann and F. Durand, "Flash photography enhancement via intrinsic relighting," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 673–678, 2004.
[8] R. Fattal, D. Lischinski, and M. Werman, "Gradient domain high dynamic range compression," *ACM Trans. Graph.*, vol. 21, pp. 249–256, 2002.
[9] G. D. Finlayson, S. D. Hordley, and M. S. Drew, "Removing shadows from images," in *Proc. 7th Eur. Conf. Comput. Vis.*, 2002, pp. 823–836.
[10] E. S. L. Gastal and M. M. Oliveira, "Adaptive manifolds for real-time high-dimensional filtering," *ACM Trans. Graph.*, vol. 31, no. 4, pp. 33:1–33:13, 2012.
[11] R. Guo, Q. Dai, and D. Hoiem, "Single-image shadow detection and removal using paired regions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2011, pp. 2033–2040.
[12] K. He, J. Sun, and X. Tang, "Guided image filtering," in *Proc. Eur. Conf. Comput. Vis.*, 2010, pp. 1–14.
[13] K. He, J. Sun, and X. Tang, "Single image haze removal using dark channel prior," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 12, pp. 2341–2353, Dec. 2011.
[14] J. Kopf, M. F. Cohen, D. Lischinski, and M. Uyttendaele, "Joint bilateral upsampling," *ACM Trans. Graph.*, vol. 26, no. 3, p. 96, 2007.
[15] D. Krishnan and R. Fergus, "Dark flash photography," *ACM Trans. Graph.*, vol. 28, no. 3, p. 96, 2009.
[16] A. Levin and Y. Weiss, "User assisted separation of reflections from a single image using a sparsity prior," in *Proc. Eur. Conf. Comput. Vis.*, 2004, pp. 602–613.
[17] M. D. Levine and J. Bhattacharyya, "Removing shadows," *Pattern Recognit. Lett.*, vol. 26, no. 3, pp. 251–265, 2005.
[18] S. Paris and F. Durand, "A fast approximation of the bilateral filter using a signal processing approach," in *Proc. Eur. Conf. Comput. Vis.*, 2006, pp. 568–580.
[19] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, no. 7, pp. 629–639, Jul. 1990.
[20] G. Petschnigg, R. Szeliski, M. Agrawala, M. F. Cohen, H. Hoppe, and K. Toyama, "Digital photography with flash and no-flash image pairs," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 664–672, 2004.
[21] R. Raskar, A. Ilie, and J. Yu, "Image fusion for context enhancement and video surrealism," in *Proc. 3rd Int. Symp. Non-Photorealistic Animation Rendering*, 2004, pp. 85–152.
[22] D. Rüenacht, C. Fredembach, and S. Süstrunk, "Automatic and accurate shadow detection using near-infrared information," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 8, pp. 1672–1678, Aug. 2014.
[23] L. Schaul, C. Fredembach, and S. Süsstrunk, "Color image dehazing using the near-infrared," in *Proc. 16th IEEE Int. Conf. Image Process.*, 2009, pp. 1629–1632.
[24] J. Sun, S. B. Kang, Z. Xu, X. Tang, and H.-Y. Shum, "Flash cut: Foreground extraction with flash and no-flash image pairs," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2007, pp. 1–8.
[25] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proc. IEEE Int. Conf. Comput. Vis.*, 1998, pp. 839–846.
[26] Y. Weiss, "Deriving intrinsic images from image sequences," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2001, vol. 2, pp. 68–75.
[27] L. Xu, F. Qi, and R. Jiang, "Shadow removal from a single image," in *Proc. 6th Int. Conf. Intell. Syst. Design Appl.*, 2006, pp. 1049–1054.
[28] L. Xu, Q. Yan, Y. Xia, and J. Jia, "Structure extraction from texture via relative total variation," *ACM Trans. Graph.*, vol. 31, no. 6, p. 139, 2012.
[29] Q. Yan, X. Shen, L. Xu, S. Zhuo, X. Zhang, L. Shen, and J. Jia, "Cross-field joint image restoration via scale map," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2013, pp. 1537–1544.
[30] L. Yuan, J. Sun, L. Quan, and H.-Y. Shum, "Image deblurring with blurred/noisy image pairs," *ACM Trans. Graph.*, vol. 26, no. 3, p. 1, 2007.
[31] X. Zhang, T. Sim, and X. Miao, "Enhancing photographs with near infra-red images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2008, p. 1–8.
[32] S. Zhuo, X. Zhang, X. Miao, and T. Sim, "Enhancing low light images using near infrared flash images," in *Proc. 16th IEEE Int. Conf. Image Process.*, 2010, pp. 2537–2540.

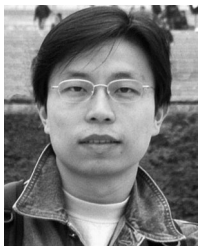
Xiaoyong Shen received the BS and MS degrees in mathematics from Zhejiang University (ZJU) in 2010 and 2012, respectively. He is currently working toward the PhD degree in the Chinese University of Hong Kong (CUHK). His research interests are computer vision and computer graphics. He is a student member of the IEEE.



Qiong Yan received the bachelor's degree in computer science and technology from the University of Science and Technology of China in 2009, and the PhD degree in computer science and engineering from the Chinese University of Hong Kong in 2013. She is currently a researcher in Image and Visual Computing Lab in Lenovo. Her research interest is in saliency detection, image filtering and image enhancement.

Li Xu received the BS and MS degrees in computer science and engineering from Shanghai Jiao Tong University (SJTU) in 2004 and 2007, respectively, and the PhD degree in computer science and engineering from the Chinese University of Hong Kong (CUHK) in 2010. He joined Lenovo R&T Hong Kong in August 2013, where he leads the imaging & sensing group in the Image & Visual Computing (IVC) Lab. He received the Microsoft Research Asia Fellowship Award in 2008 and the best paper award of NPAR 2012. His major research areas include motion estimation, motion deblurring, image/video analysis, and enhancement. He is a member of the IEEE.

Lizhuang Ma received the BSc and PhD degrees from Zhejiang University, China, in 1985 and 1991, respectively. He received the China National Excellent Young Scientist Foundation. He is currently a distinguished professor in the Department of Computer Science, Shanghai Jiao Tong University. His research interests include CAD, computer graphics, image processing, and image based graphical techniques such as rendering, reconstruction and modeling. He is a member of the IEEE.



Jiaya Jia received the PhD degree in computer science from the Hong Kong University of Science and Technology in 2004 and is currently a professor in Department of Computer Science and Engineering, Chinese University of Hong Kong (CUHK). He heads the research group in CUHK, focusing specifically on computational photography, machine learning, practical optimization, and high- and low-level computer vision. He currently serves as an associate editor for the *IEEE Transactions on Pattern Analysis and Machine Intelligence* (TPAMI) and served as an area chair for ICCV 2011 and ICCV 2013. He was on the program committees of several major conferences, including ICCV, ECCV, ICCP, SIGGRAPH and CVPR, and co-chaired the Workshop on Interactive Computer Vision, in conjunction with ICCV 2007. He received the Young Researcher Award 2008 and Research Excellence Award 2009 from CUHK. He is a senior member of the IEEE.