Removing Reflection From a Single Image With Ghosting Effect

Yan Huang, Yuhui Quan*, Yong Xu, Ruotao Xu, Hui Ji

Abstract—Removing the undesired reflections of images taken through glass is an important problem in digital photography and many other vision applications. The so-called ghosting effect, *i.e.* the pattern repetitiveness in reflection, is an effective cue used by existing techniques to remove reflection from images. Existing methods take a two-stage approach that first estimates the parameters of ghosting effect and then models reflection removal as a two-layer separation problem: reflection layer and latent image layer. This paper aimed at addressing one main challenge in such an approach, *i.e.* how to distinguish the repetitive patterns on the later image layer and the ghosting patterns on the reflection layer. Based on the observation that the number of repeats of natural image patterns is often different from that of ghosting patterns, we proposed a wavelet transform based regularization method. Together with a novel weighting scheme, the proposed method is capable of accurately separating two layers, and experimental results justified its advantages over the existing ones on both synthetic and real data set.

Index Terms—Reflection removal, Image separation, Ghosting effects, Image decomposition

I. Introduction

It often happens that one take pictures through glass windows, *e.g.*, taking pictures of the paintings in the museum [1], [2], [3], [4], [5], the dresses in the showcase [6], [7], [8], the landscape outside when in a train or plane [9], [10]. Unfortunately, it is known that the resulting picture taken through glass often contains the reflection of the scene behind the camera. The avoidance of such reflection requires either specific hardware (*e.g.* polarizers) or manual control of lighting conditions around camera. There is certainly the need to develop effective post-processing techniques to remove the reflections from those images taken through glass windows.

In general, without additional priors or assumptions on the reflection, it is not possible to separate the reflection caused by glass and the content of interest. One powerful cue for identifying reflection in the image is the so-called *ghosting effect*, especially for the images taken through thick or insulating glass windows. See Fig. 1 for an illustration. Ghosting effect of an image caused by the reflection of glass refers to the two or multiple reflections of the reflected scene off the different glass surfaces. For example, both panels

Yan Huang, Yuhui Quan, Ruotao Xu and Yong Xu are with School of Computer Science & Engineering at South China University of Technology, Guangzhou 510006, China, and also with the Guangdong Provincial Key Laboratory of Computational Intelligence and Cyberspace Information, China. (email: huangkaiyan@scut.edu.cn; csyhquan@scut.edu.cn; ruotao.xu@mail.scut.edu.cn; yxu@scut.edu.cn)

Hui Ji is with Department of Mathematics at National University of Singapore, Singapore 119076. (email: matjh@nus.edu.sg)

Asterisk indicates the corresponding author.

Fig. 1: Illustration of removing reflection with ghosting effect from a single image. (a) The image with reflection. (b) The result of removing reflection using the proposed method.
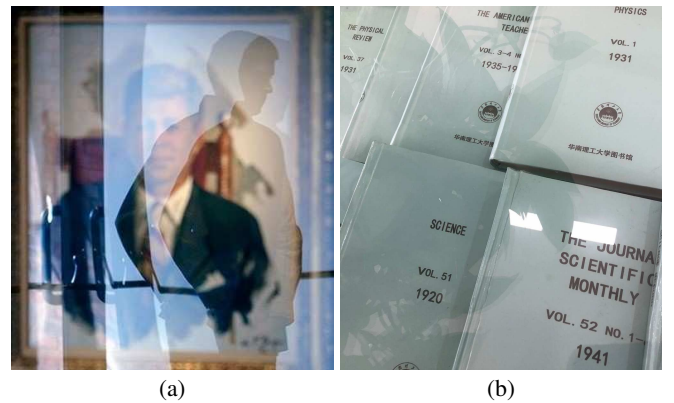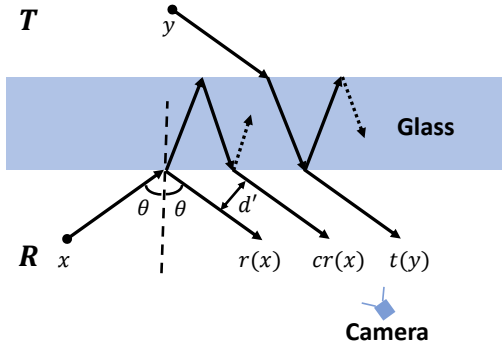

Fig. 2: Illustration of ghosting effect. (a) Ghosting reflection caused by both panels of glass. (b) Ghosting reflection caused by both sides of thick glass.

of a two-panel window will reflect the scene behind and the two reflections differ from each other by their positions and brightness strengths. The similar effect with weakened brightness happens for a single-panel window with thick glass. See Fig. 2 for an illustration of such two types of reflections. As the brightness strength of the third reflection generally is much less than the primary reflection, only the double reflection is usually considered in practice; see e.g. [11], [12].

The image with ghost effect is modeled in [11], [12] by the following composite model:

$$\boldsymbol{Y} = \boldsymbol{T} + \boldsymbol{R} \otimes \boldsymbol{k} + \boldsymbol{N}, \tag{1}$$

where $\otimes$ denotes the convolution operation, $\boldsymbol{Y}$ denotes the input image, $\boldsymbol{N}$ denotes the noise, $\boldsymbol{T}$ denotes the transmission layer which is the captured frame of the scene behind the window, and $\boldsymbol{R} \otimes \boldsymbol{k}$ denotes the reflection layer which is the captured frame of the scene in front of the window. The

Fig. 3: Diagram of ghosting effect.

Image with local periodic patterns   Results by Shih *et al.* [12]

Fig. 4: Demonstration of the results from Shih *et al.* [12] on the image whose transmission layer contains periodic patterns.

reflection layer is modelled by an image pattern $\boldsymbol{R}$ convoluted with a two-impulse kernel of the form:

$$k = \delta(\boldsymbol{x}) + c\delta(\boldsymbol{x} - \boldsymbol{d}), \qquad (2)$$

where δ denotes the Dirac, and c, $\boldsymbol{d}$ are two parameters.

The convolutional model of the reflection layer, expressed by (1) and (2), is derived from the diagram of ghosting effect illustrated in Fig. 3. It can be seen that each point is sensed into different pixels when dissipating its energy among the different reflection orders, and thus the effect is created by the second reflection. With unpolarized light and flat glass, both t and r are only dependent on the incident angle. The distance between the primary reflection and the successive one is $\boldsymbol{d'}$ for constant incident angle. See [11] for more details. Therefore, the effect can be modeled by two replicates of the same object with the displacement $\boldsymbol{d'}$ and with different average brightness, which then can be formulated as a convolution with a specific kernel function parameterized by the ghosting shift $\boldsymbol{d}$ which is proportional to the physical distance $\boldsymbol{d'}$ and the attenuation factor c which is the ratio between the brightness strength of the first reflection and the successive second reflection. In other words, The displacement and the difference of average brightness is parameterized by $\boldsymbol{d}$ and c in $\boldsymbol{k}$. These two parameters depend on the reflectance property and the configuration of glass surface.

Most existing methods, *e.g.* [12], take a two-stage approach for reflection removal with ghosting effect. Such an approach first estimates the two parameters of the kernel $\boldsymbol{k}$ and then estimates the reflected image region $\boldsymbol{R}$ using the estimated $\boldsymbol{k}$. The second stage requires solving a challenging ill-posed inverse problem, *i.e.* simultaneously estimating the transmission layer $\boldsymbol{T}$ and the reflection layer $\boldsymbol{R} \otimes \boldsymbol{k}$ from (1). Motivated by the fact that the reflection layer $\boldsymbol{R} \otimes \boldsymbol{k}$ contains the replicates of the same object, the method proposed in [12] imposes the periodicity prior on $\boldsymbol{R} \otimes \boldsymbol{k}$ for separating two layers. Unfortunately, such a prior also holds true for the transmission layer as well, especially for urban scenes. Thus, the periodicity prior on the reflection layer is not sufficient for distinguishing the ownership of repeating patterns. Therefore, the output from the method proposed in [12] often show noticeable artifacts on the transmission layer that contains periodic patterns. See Fig. 4 for an illustration. It can been seen that there are artifacts on the windows and the long white roof.
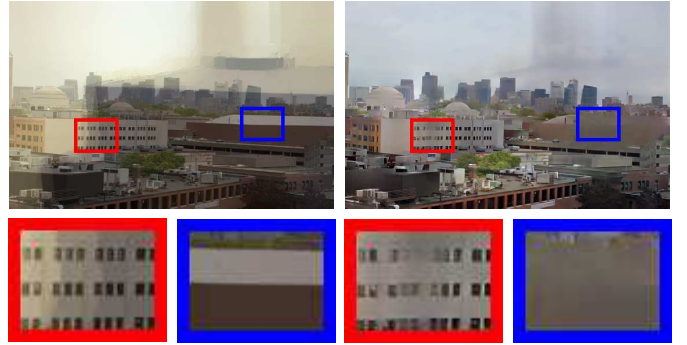
Motivated by the weakness of existing methods that use the ghosting effect for reflection removal, this paper aims at developing a more powerful method for removing reflection from the input image, which focuses on a better regularization model for distinguishing the repetitive patterns on the transmission layer and that from the reflection layer. Notice that the number of the repeats of natural image patterns usually is different from that of ghosting patterns: the former is usually more than two while the latter is fixed to two. Thus, the number of repetitive patterns along the direction of the estimated ghosting shift indicates the ownership of the repetitive patterns. Motivated by such an observation, in this paper we propose a wavelet transform based regularization method with a novel re-weighting scheme for solving the inverse problem of reflection removal. Together with a refined version of the estimator of ghosting parameters used in [12], the proposed method provides better performance than existing methods on both synthetic data and real data. In short, the contributions of the proposed method are summarized as follows:

- A new periodicity-based prior for accurately identify the ownership of repeating patterns for two layers.
- A wavelet transform based regularization model with a novel re-weighting scheme for effectively separate the two layers.

In addition to the performance gain, the proposed one has other merits over existing ones. For example, Shih *et al.*'s method [12] requires a large dataset for GMM (Gaussian Mixture Model) training and requires solving a complex optimization problem. In contrast, the proposed one is free from training and data, and has a much simpler optimization problem to solve.

The paper is organized as follows. In section II we give a brief discussion on the related work on reflection removal. In section III, we present our method and the implementation details. In section IV, the experiments are conducted for the evaluation of the proposed work and for the comparison to other related methods. Section V concludes the paper.

II. RELATED WORK

Generally, reflection removal considers the following composite model for images with reflections (see *e.g.* [13], [2],

[6], [14], [7], [15]):

$$\boldsymbol{Y} = \boldsymbol{T} + \widetilde{\boldsymbol{R}}, \qquad (3)$$

where $\boldsymbol{Y} \in \mathbb{R}^{M \times N}$ is the input which is the superposition of the transmission (background) layer $\boldsymbol{T} \in \mathbb{R}^{M \times N}$ and the reflection layer $\widetilde{\boldsymbol{R}} \in \mathbb{R}^{M \times N}$. The goal of reflection removal is then to estimate the transmission layer $\boldsymbol{T}$ from $\boldsymbol{Y}$. Clearly, the problem of reflection removal is ill-posed, owing to the ambiguities between $\boldsymbol{T}$ and $\widetilde{\boldsymbol{R}}$. Additional priors on both $\boldsymbol{T}$ and $\widetilde{\boldsymbol{R}}$ are needed to separate these two layers from the input. Based on available input source, reflection removal can be classified into two categories, *i.e.* multiple-image based approach and single-image based approach.

A. Multiple-image based reflection removal

Earlier work on reflection removal treated the problem as a blind source separation problem, which separates the mixed signals into two independent signals. Farid *et al.* [16] collected two images taken with different polarization angles and estimated the mixing coefficients using independent component analysis. Richard *et al.* [17] proposed to use constrained least squares to recover the background layers based on an arbitrary number of composite images with known motion estimates. Bernard and Michal [18] presented an approach for separating reflections under various real-world scenarios, including images obtained under different polarizations or videos with non-rigid transparent motions. Ni *et al.* [19] focused on the images captured by light field camera which provides extra directional information of incoming rays. These methods directly model the reflection generation process and solve it without using specific properties of each layer.

More recent work [20], [6], [4], [9], [14], [7], [21], [8] focused on finding specific properties of each layer or exploiting the different motion behaviors between layers. The most often-seen natural image prior is the sparsity of image gradients [20], [6], [4], [9], [14], [21]. The Laplacian approximation is used in [6], [9] for its quick convergence and simplicity. The Hyper-laplacian approximation used in [14], [21] is regarded as a better distribution to model the long-tail distribution of the image gradients. In order to make the distribution of image gradients adaptive to input, Gai *et al.* [20] proposed to learn the distribution of gradients from the given data.

In addition to the image-prior-based approaches, there is also a class of approaches which utilizes motion cues from different layers for reflection removal. Based on the fact that the reflection layer moves slower than the background since it is closer to the camera than the transmission layer, Xue *et al.* [9] proposed to separate the two layers using the difference of motion speed between layers. Built upon the observation that the background layer as the major part of an image is rather static, a number of approaches used the SIFT flow [22] to first warp the images to the reference one, and then to estimate the edge map using different techniques, *e.g.* the sparsity of edge occurrence [6], the low-rank matrix completion [8], the motion and intensity score [7]. One main challenge in these flow-based methods is that the flow estimation often is not stable for images with reflections. Yang *et al.* [23] addressed such an issue by considering a generalized version of the brightness constancy constraint. Ajay *et al.* [24] constructed a spatio-temporal optimization model which jointly solves the flow estimation and reflection separation.

B. Single-image based reflection removal

As single-image based reflection removal problem is ill-posed, additional information is required for the separation. Levin *et al.* [13] searched for a decomposition of two layers with the minimal amount of edges and corners. Additionally, Levin *et al.* [25] also required user interactions for marking a number of edges within backgrounds or reflection labels, and then refined the separation framework based on the sparsity of derivative filters with an iterative re-weighted least squares scheme. Wan *et al.* [15] proposed to retrieve external patches based on a natural image database and combine the sparsity prior and non-local similarity prior of images for the removal.

Similar to some multiple-image-based ones, some single-image-based methods also considered the different properties of the two mixed layers. An assumption is used in [26], [27], [10] that the background is in focus of the camera and the reflection is out of focus. Thus, the reflection layer is relatively smoother than the background layer. Based on such an assumption, Li *et al.* [26] separated the two layers by imposing different sparsity properties on two layers respectively: first-order derivative filters on the transmission layer and second-order derivative filter on the reflection layer. Wan *et al.* [27] used a multi-scale technique to calculate the confidence map for background edge selection. Nikolaos *et al.* [10] directly removed the reflections based on a Laplacian data fidelity and an ℓ_0 gradient sparsity term. Shih *et al.* [12] exploited the ghosting cue that happens on images taken through double-panel window. The work of Shih *et al.* is the most related one to the proposed method.

Recently, taking the advantages of deep learning, several approaches [28], [29], [30], [31] have been proposed to use neural networks for reflection removal. Fan *et al.* [28] used synthetic images to train a deep neural network. However, it did not show noticeable performance gain on real images when compared to the traditional non-learning methods. Such a phenomena comes from the difference between synthetic images and real images, in terms of complex optical characteristics of reflection. Thus, Wan *et al.* [32], [29] built a dataset of real images with reflections, and combined it with synthetic images to train a multi-scale edge-guided concurrent network for reflection removal. Using both real data sets and synthetic data sets, Zhang *et al.* [31] optimized the convolutional neural network in terms of perceptual loss.

C. Single-image reflection removal for ghosting effect

The existing work on removing reflection with ghosting effect is the most related to this paper. Indeed, there are few single-image based reflection removers that are specifically designed for the reflection with ghosting effect. The seminal work along this line is done in [11] which formulated multiple reflections (*i.e.* ghosting reflection) by the convolutional model (1) with a kernel of the form (2). The transmission layer

is then recovered by a deconvolution process. The advantage of convolution model for ghosting, *i.e.*, $\widetilde{R} = R \otimes k$, is that it introduces different properties of the transmission layer T and the reflection layer $\widetilde{R}$. This helps resolving the ambiguities between T and $\widetilde{R}$, and the separation becomes feasible.

Following the model of [11], Shih *et al.* [12] proposed to solve the reflection removal problem by a two-stage approach. The proposed method first estimated the two parameters of kernel k, *i.e.* the spatial shift d and the attenuation c in (2), and then imposed the mixture of Gaussian prior on each layer for layer separation. The ghosting effect of the reflection layer is implemented by imposing periodic prior on reflection layer. Better results than precedent methods were reported on both the real and synthetic images. However, the method often yields poor results when the transmission layer also contain repeating patterns, which is often seen in urban scenes. See Fig. 4 for an example. It can be seen that the method [12] wrongly treats the background objects with the period similar to the shift distance as the reflection and tries to remove such objects from the background, which results in poor separation of the two layers. Also, the method [12] is built on the patch-based GMM [33] learned from a large set of natural images.

It is noted that the ghosting effect is also studied in the merging of multi-exposure images; see *e.g.* [34], [35], [36]. In such a case, the pixels of the different images are to be aligned. The motion of the camera or moving objects in the scene will cause ghosting artifacts if the motion is large. Then, a de-ghosting is needed for recovering images from such a degradation. Such problems have a different setting from ours in terms of available sources, and they are usually not formulated as a layer separation problem as ours.

III. MAIN BODY

Through the paper, unless specified, bold upper letters are used for matrices, *e.g.* Y, T, bold lower letters for column vectors (*e.g.* y, t), light lower letters for scalars (*e.g.* y, t), and calligraphic letters for operators (*e.g.* $\mathcal{S}, \mathcal{T}$). Let 0 and I denote zero matrix and identity matrix respectively. Let $\mathbb{N}$ denote the set of natural numbers, $\mathbb{Z}$ denote the set of integers, $\mathbb{Z}_M$ denote the integer set $1, \cdots, M$, and $\mathbb{R}$ denote the set of real numbers. Given a sequence $\{y^{(t)}\}_{t \in \mathbb{N}}$, $y^{(t)}$ denotes the t-th entry in the sequence. For a vector $x \in \mathbb{R}^N$, let $x(i)$ denote the i-th element of x, and define $\|x\|_2 = \sqrt{\sum x(i)^2}$. For a matrix X, let $X(i, j)$ denote the element of X at the i-th row and the j-th column. Define $\|X\|_F = \sqrt{\sum_{i,j}(X(i,j))^2}$ and $\|X\|_1 = \sum_{i,j}|X(i,j)|$. For matrix concatenation, semicolons are used for adding elements row-wisely and commas are used for adding elements column-wisely.

We proposed the following regularization-based optimization model for estimating both layers R and T:

$$\min_{T, R} \frac{1}{2}\|Y - T - R \otimes k\|_F^2 + \lambda\Gamma_1(T) + \beta\Gamma_2(R), \quad (4)$$

where the first term denotes the fidelity term in the presence of i.i.d. Gaussian noise, the second term and the third term represent the regularizations on both two layers. It is noted that the ghosting kernel k is also unknown. Same as [12], we also take a two-stage approach to solve the above problem:

- First stage: estimate the ghosting kernel k. In this stage, our method is mainly based on the techniques proposed in [11], [12], with further refinements for better performance.
- Second stage: estimate two layers R and T via solving (4). The main effort in this stage is how to construct two regularization terms Γ_1, Γ_2 so as to accurately distinguish T and R.

The key issue we aim to address when designing two regularization terms is how to accurately determine the ownership of repeating patterns for two layers. Our idea is based on the observation that the repeating times of such patterns are different between the two layers. In addition, the difference of the sharpness between the two layers is also exploited. These observations lead to a weighted wavelet transform based regularization for effectively separating the two layers. Compared to Shih *et al.*'s method [12], ours use a more accurate prior on pattern recurrence to separate two layers. Also, our wavelet transform based method is not learning based method, and thus is less computation demanding and data demanding than the GMM prior used in [12].

A. Estimation of ghosting kernel

Recall that the ghosting kernel k defined by (2) is a two-impulse kernel with two parameters: the amplitude c and displacement d of the displaced impulse. We proposed a refined version of the techniques proposed in [12] to more accurately estimate such two parameters.

In [12], the displacement d is the first to be estimated and the amplitude c is the second. The displacement d of the kernel k is actually the displacement of the reflected contents, which can be estimated by checking the position difference of the edges of reflected contents. Consider an input image $Y \in \mathbb{R}^{M \times N}$, its edge map, denoted by $S \in \mathbb{R}^{M \times N}$, is obtained by running the Sobel edge detector:

$$S(i, j) = \sqrt{|(g_x \otimes Y)(i,j)|^2 + |(g_y \otimes Y)(i,j)|^2}, \quad (5)$$

where $g_x = [-1, 0, 1; -2, 0, 2; -1, 0, 1]$ and $g_y = [1, 2, 1; 0, 0, 0; -1, -2, -1]$. Then, the autocorrelation map of the edge map S:

$$C = S \otimes S(\cdot-), \quad (6)$$

measures the self-similarity of S with different displacements. A local maximum of C would indicate possible repetitive patterns in S, *i.e.* shifted copies of the reflection layer R in the context of ghosting effect. In our implementation of maxima detection, we apply the max pooling on C, *i.e.* to search on C for all local maximums using a neighbor of size $r \times r$. For better robustness, the following local maximums are discarded:

- Local maximums in C corresponds to the value in S larger than a threshold σ ($= 0.2$ in default setting). Such maximums are likely to be the repeating patterns in the transmission layer, as they corresponding edges are those strong ones.
- Local maximum within t ($= 5$ in default setting) pixels from the original point. This is an often-seen trick to

remove the ambiguities caused by the edges with large width, as a edge with large width can be viewed as repetitive patterns.

Finally, the estimation of the ghosting displacement d is defined as the displacement to the origin of the one which has the largest magnitude among all remaining local maximums. The sign of d remain to be determined, and we choose the sign of d such that $c < 1$, which is based on the assumption that the second reflection has lower energy than the first one.

The amplitude parameter c is evaluated as follows. Firstly, a set of interest points are detected from the input using a Harris corner detector. Secondly, a 5×5 contrast normalized patch is extracted around each corner feature. Lastly, the variances of each pair of matched patches, denoted by $\text{var}[p_i]$ and $\text{var}[p_j]$ respectively, are calculated. Lastly, the parameter c is defined as the weighted summation over the ratios of two variances:

$$c = \frac{1}{z_0} \sum_{ij} w_{ij} \sqrt{\frac{\text{var}[p_i]}{\text{var}[p_j]}}, \tag{7}$$

where z_0 is the normalization factor, and w_{ij} is the weight for evaluating the contribution of each pair of patches to estimate the attenuation c, which is set as

$$w_{ij} = \exp(-\|p_i - p_j\|_2^2 / 0.1)$$

in the implementation.

It is noted that the kernel estimation scheme described above can be viewed as a refined version of that of [11], [12], with the improvement on simplicity and performance. The main refinements over that of [11], [12] are listed as follows.

- The Laplacian operator is replaced by the Sobel edge detector to construct the edge map, for better performance on detecting strong edges and better robustness to noise.
- A thresholding strategy is implemented to discard those estimations on displacement that are likely to obtained from the edges belonging to the transmission layer.

B. Regularization model for layer separation

After estimating the ghosting kernel k using the procedure described in the previous section, the next step is to separate two layers via solving the optimization model (4). The success of the model (4) lies in about how to design two regularization terms: one for the transmission layer and the other for the reflection layer. We proposed the following regularization models. Let $\{f_i\}_{i=1}^L$ denote the filter bank of a wavelet transform, and let A denote the weighting matrix which is very related to the likeliness of the ownership of the edges for two layers. Then, the regularization model is expressed as:

$$\min_{T,R} \frac{1}{2}\|Y - T - R \otimes k\|_F^2 + \lambda \sum_i \|A \odot (f_i \otimes T)\|_1$$
$$+\beta \sum_i \|(1 - A) \odot (f_i \otimes R)\|_1 \tag{8}$$

where $\odot$ is the element-wise multiplication. The regularization model above involves two components: the ℓ_1-norm relating regularization under wavelet transform, and a weighting matrix A. In the following, we give detailed discussions on them.

C. Wavelet transform based regularization

It can be seen that there are two operations involved in the ℓ_1-norm relating regularization for each layer. The coefficients $\{f_i \otimes T\}$ and $\{f_i \otimes R\}$ denote the high-pass wavelet coefficients of the layer T and the layer R respectively. The ℓ_1-norm regularization on natural images under wavelet transform has been widely used in many image recovery tasks; see *e.g.* [37], [38], [39], [40]. Such a regularization exploits the sparsity prior of the wavelet coefficients of natural images.

A single-level undecimal wavelet transform is implemented in this paper, which consists of an analysis operator and a synthesis operator. Given the filter bank $\{f_\ell\}_{\ell=1}^L$ of a wavelet transform, the analysis operator is defined by

$$W : T \to C := [C_1; C_2; \ldots; C_L] \tag{9}$$

where $C_\ell = f_\ell(-\cdot) \otimes T$. In other words, the output of the analysis operator is the collection of the input convoluted with a set of filters. The synthesis operator is the adjoint operator of W, which is defined as

$$W^\top : C \to \sum_{\ell=1}^L f_\ell \otimes C_\ell.$$

The wavelet transform has the so-called perfect reconstruction property: $W^\top W = I$, which avoids calculating $W^\top W$ in our numerical scheme. Also, the filter bank associated with wavelet transform contains the filters that correspond to the difference operators with different orders at different orientations, which is very helpful to suppressing artifacts. In the implementation, the linear B-spline framelet transform [41] is implemented, which contains totally nine two-dimensional filters composed by the tensor product of the following three one-dimensional filters:

$$f_1 = \frac{1}{4}(1,2,1)^\top; \; f_2 = \frac{\sqrt{2}}{4}(1,0,-1)^\top; \; f_3 = \frac{1}{4}(-1,2,-1)^\top.$$

The wavelet transform based regularization is effective on suppressing the undesired artifacts along image edges when separating two layers. However, it does not provide sufficient discrimination capability to accurately separate two layers. The discriminative information of two layers is indeed encoded in the weighting matrix A. Recall that high-pass wavelet coefficients are very related to the image gradients with different orientations and different orders. The role of the weighting matrix A is then for the assignment of image gradients to which layer, with a particular focus on distinguishing the difference in terms of edge periodicity.

D. Construction of weighting matrix

One main challenge when attempting to separate the transmission layer and the reflection layer is that the content in the transmission layer with strong globally repetitive patterns can also be well modeled by the convolution process used in the reflection layer. As a result, the repetitive patterns in the transmission layer might be wrongly assigned to the reflection layer, and vice versa. the weighting matrix A in (8) aims at resolving such ambiguities. The range of all entries of A is $[0, 1]$, and the entry value indicates the likeliness that a point

contains the content of the reflection layer. Larger value is assigned to $\boldsymbol{A}(i,j)$ if an edge of the reflection layer is more likely on the point (i,j), which leads to more penalty on the image gradients of $\boldsymbol{T}$ than on $\boldsymbol{R}$. As a result, the majority of image gradient at that point goes to the reflection layer.

We propose the following scheme to define $\boldsymbol{A}$:

$$\boldsymbol{A} = \frac{\boldsymbol{A}_1 + \boldsymbol{A}_2}{2}, \quad (10)$$

where the definitions of $\boldsymbol{A}_1, \boldsymbol{A}_2 \in \mathbb{R}^{M \times N}$ are motivated from the following two priors:

- *Difference in sharpness.* The background objects in the transmission layer are often sharper and much more clear than that in the reflection layer. This is also the assumption used in the kernel estimation for discarding strong edges in the transmission layer.
- *Difference in periodicity.* The repeating patterns of the objects in the transmission layer often repeat several times within a local region (*i.e.* the repeats of such patterns are larger than one), while the objects in the reflection layer only repeat once in the image for ghosting effect. Such an observation is particularly useful for the weak edges wrongly processed under the first assumption.

Regarding the assumption of sharpness, we use the edge map $\boldsymbol{S}$ defined in (5) to select the background edges. First, we use K-means to cluster the values of $\boldsymbol{S}$ into three clusters. Such a single-dimensional clustering form four intervals: $\{[a_0, a_1], [a_1, a_2], [a_2, a_3], [a_3, a_4]\}$ with $a_0 < a_1 < a_2 < a_3$. The first interval $[a_0, a_1]$ corresponds to the strongest edges, while the last interval correspond to the flat or nearly-flat regions. See Fig. 7 (a) for one example on the clustering. Clearly, the first interval basically contains all the strong edges in transmission layer. The matrix $\boldsymbol{A}_1(i,j)$ is then defined by:

$$\boldsymbol{A}_1(i,j) = \begin{cases} 1, & \boldsymbol{S}(i,j) < a_1 \\ 0, & \boldsymbol{S}(i,j) \geq a_1 \end{cases}. \quad (11)$$

In other words, $\boldsymbol{A}_1$ assigns an edge point to the reflection layer if its associated edge is in the clusters of non-strong edges.

Regarding the prior on periodicity difference, refer to Fig. 5 for some examples. It can be seen that the repetitive patterns in the transmission layer often repeats more than twice. Thus, we propose to calculate $\boldsymbol{A}_2$ as follows. Firstly, the edge map $\boldsymbol{M} \in \mathbb{R}^{M \times N}$ of the input image $\boldsymbol{Y} \in \mathbb{R}^{M \times N}$ is calculated by

$$\boldsymbol{M} = \boldsymbol{g} \otimes (\mathcal{C} \circ \boldsymbol{Y}), \quad (12)$$

where $\mathcal{C}$ is the Canny edge detector[1] and $\boldsymbol{g}$ is a Gaussian smoothing kernel with the standard deviation 0.5 which is a default value in many image processing methods. Secondly, for each position $\boldsymbol{x} \in \mathbb{Z}_M \times \mathbb{Z}_N$ in $\boldsymbol{M}$, we extract the patch centered at $\boldsymbol{x}$ with size $p \times p$, which is denoted by $\boldsymbol{P}(\boldsymbol{x})$. The correlation coefficients between $\boldsymbol{P}(\boldsymbol{x})$ and $\boldsymbol{P}(\boldsymbol{x}+\boldsymbol{d}), \boldsymbol{P}(\boldsymbol{x}-\boldsymbol{d}), \boldsymbol{P}(\boldsymbol{x}+2\boldsymbol{d}), \boldsymbol{P}(\boldsymbol{x}-2\boldsymbol{d})$ are calculated respectively where $\boldsymbol{d}$ is the displacement parameter of the ghosting kernel that has been estimated, and the first two maximums of these four

---

[1]We use the Canny detector instead of the previously-used Sobel detector for obtaining additional information.

coefficients denoted by $\boldsymbol{Q}_1(\boldsymbol{x}), \boldsymbol{Q}_2(\boldsymbol{x})$. Define $\boldsymbol{Q} = |\boldsymbol{Q}_1 - \boldsymbol{Q}_2|$, then we define

$$\boldsymbol{A}_2(i,j) = \begin{cases} 1, & \boldsymbol{Q}(i,j) > \alpha \ \& \ \boldsymbol{M}(i,j) \neq 0, \\ 0, & \text{otherwise}, \end{cases} \quad (13)$$

where the constant α is a predefined threshold (0.5 in the implementation). In other words, based on the periodicity prior, $\boldsymbol{A}_2$ tells whether an edge point on a weak edge belongs to the reflection layer or not. If an edge point on a weak edge has repetitive patterns that repeat only once around its neighborhood, then it is likely to belong to the reflection layer.

See Fig. 6 for an example of the correlation coefficients. It can be seen that for an edge pixel on the reflection layer (*e.g.* for the center pixel in the third blue rectangle), its four correlation coefficients have one large value and three small values, and thus the difference between $\boldsymbol{Q}_1$ and $\boldsymbol{Q}_2$ would be large. In contrast, for an edge pixel on the transmission layer (*e.g.* the center pixels in the third yellow, red, magenta rectangles), there are at least 2 large values or all 4 small values in the four correlation coefficients, and thus the difference between $\boldsymbol{Q}_1$ and $\boldsymbol{Q}_2$ would be small. Thus, the reflections from background can be recognized according to the value of the difference of matrix $\boldsymbol{Q}_1$ and $\boldsymbol{Q}_2$. See Fig. 7 (b) for the visualization of the difference map $\boldsymbol{Q}$

Fig. 5: Periodic patterns of real objects and reflected contents.
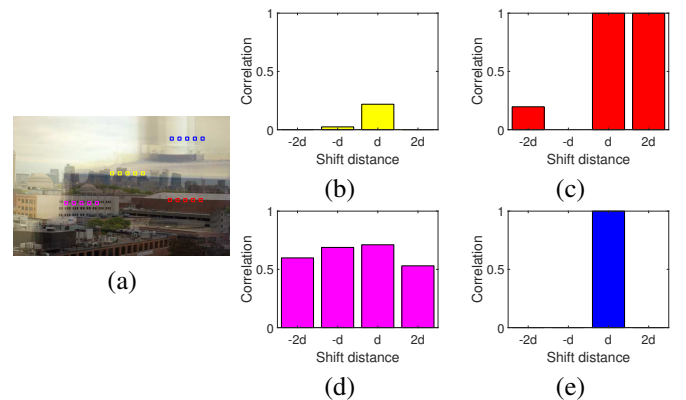

Fig. 6: Local correlation coefficients of an image shown in (a). (b)-(e) Correlation coefficients between the central patch and its neighboring four patches with displacement $-2\boldsymbol{d}, -\boldsymbol{d}, \boldsymbol{d}, 2\boldsymbol{d}$ respectively. The bar in each sub-figure corresponds to the patches with the same color in (a).

Once $\boldsymbol{A}_1$ and $\boldsymbol{A}_2$ are constructed, we have the weighting matrix $\boldsymbol{A}$ defined using (10). See Fig. 8 for an example of the matrices $\boldsymbol{A}_1, \boldsymbol{A}_2, \boldsymbol{A}$. See Alg. 1 for the outline of our method.
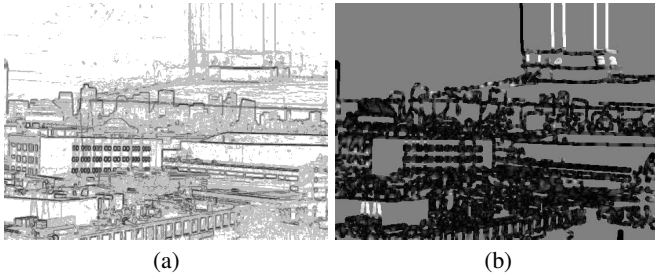
(a)   (b)

Fig. 7: Illustration of intermediate results for calculating A_1 and A_2. (a) Clustered intervals on the Sobel edge map for generating A_1. (b) The map Q computed on the sorted maximal correlation coefficients for generating A_2.
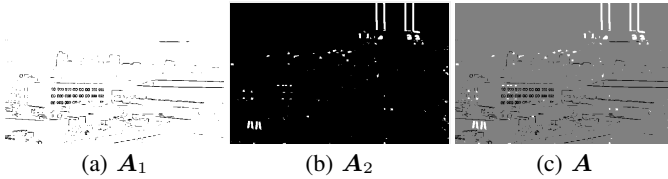

(a) A_1 (b) A_2 (c) A

Fig. 8: Illustration of matrices A_1, A_2 and A. (a) Matrix A_1. (b) Matrix A_2. (c) Spatial weight matrix A.

E. Numerical solver

For completeness, this section give a detailed description on the numerical algorithm used for solving model (8). For the convenience of presentation, we represent the related variables by the vectorization form. Let $y, t, r, a \in \mathbb{R}^{MN \times 1}$ denote the vectorization of the input image Y, the transmission layer T, the reflection layer R, and the weighting matrix A respectively. The vectorization is done by sequentially concatenating the columns of the matrix. Let $K \in \mathbb{R}^{MN \times MN}$ denote the convolution matrix for the two-dimensional kernel k, that is, the Kr is equivalent to the vectorization of $k \otimes R$. Let $\mathcal{D}(x)$ denote the diagonal matrix with x as its diagonal elements. Let W denote the analysis operator of the two-dimensional B-spline wavelet frame. Using these notations, the model (4) can be rewritten as

$$\min_{u} \frac{1}{2}\|y - Hu\|_F^2 + \|Du\|_1, \qquad (14)$$

where $u = [t^\top, r^\top]^\top$, $D = \begin{bmatrix} \mathcal{D}(\lambda a)W & 0 \\ 0 & (\beta - \mathcal{D}(\beta a))W \end{bmatrix}$, and $H = \begin{bmatrix} I & K \end{bmatrix}$. The problem (14) is a standard ℓ_1-minimization problem which can be efficiently solved by the split-Bregman method [42]. See Appendix A for all the details.

Regarding the initialization, we define $u^{(0)} = [t^{(0)}, r^{(0)}]$ with $t^{(0)} = y$ and $r^{(0)} = 0$. For better performance, we first set $a = [0.5, \cdots, 0.5]$ and run the algorithm for 5 iterations. Then the outputs are used as the initialization to restart the algorithm. The final output t is reshaped into an image, with values projected to $[0, 255]$.

IV. EXPERIMENTS

To evaluate the proposed method, we conducted several experiments on both synthetic data sets and real images. For

---

Algorithm 1 Removing reflection with ghosting effect

INPUT: Image Y with ghosting reflections
OUTPUT: Transmission layer T
Main procedure:
1) Estimate the ghosting kernel k;
2) Calculate the weighting matrix A;
3) Solve T from (4).

---

comparison, in addition to Shih *et al.* [12] which is the most related work to ours, we also compared our method against the other state-of-the-art methods: Li *et al.* [26] and Nikolaos *et al.* [10]. The results of these methods for comparison are either directly cited from published paper or generated using the codes from the authors with suggested parameter setting.

Through all the experiments, the parameters of the proposed method are set as follows. The two parameters in the model (4) are set as $\lambda = 0.01$, $\beta = 0.01$, according to the simple parameter tuning on several images with the candidate values $\{0.001, 0.01, 0.1, 0.5, 1\}$. In kernel estimation, the neighbourhood size used in searching local maximums is set to $r = 5$. The patch size for computing local correlation coefficients in the calculation of weighting matrix A is set to $p = 8$. The parameters in the split-Bregman are simply set as $\gamma_1 = 1, \gamma_2 = 1$.

A. Results on synthetic images

Fig. 9 shows the results by the proposed method and by the method [12] on an synthetic image. Such a synthetic images is synthesized as follows. The transmission layer is composed of a long rectangle and five short rectangles, which are the simulation of real objects, and the reflection layer contains the ghosting version of the circle, which is generated using model (1) with the convolution kernel parameters $d = [25, 0]$, $c = 0.5$. The periods of the short rectangles are the same as the shifted distance of ghosting circle, and the period of the long rectangle is less than half the length. The results by [12] are shown in Fig. 9 (b-c) and the results of our method are shown in Fig. 9 (e)-(f). It can be seen that the results by [12] are not good, which mix up the reflections and the period patterns in transmission layer. In contrast, by utilizing the periodicity difference of repeating patterns in two layers, our method can identify the background objects which show similar shifting properties and thus achieved better results.

To quantitatively evaluation, totally 143 images with ghosting reflections are synthesized using the model (1). In the synthesis, the brightness parameter c is randomly sampled from $[0.5, 1]$, and the displacement parameter d is a random vector with $1 < \|(d)\|_2 \leq 20$. The transmission layer T and reflection layer R are randomly sampled from the BSDS-500 dataset [43]. See Fig. 10 for some synthesized examples.

The results by ours and other methods on such a dataset are evaluated in terms of both PSNR and SSIM. See Table I for the results. The results from our method score average PSNR of 20.43dB and SSIM [44] of 0.81, while the results from Li *et al.*'s method [26] only score average PSNR of 14.48dB and SSIM of 0.57. The other two methods for comparison perform better than Li *et al.*'s method [26] but
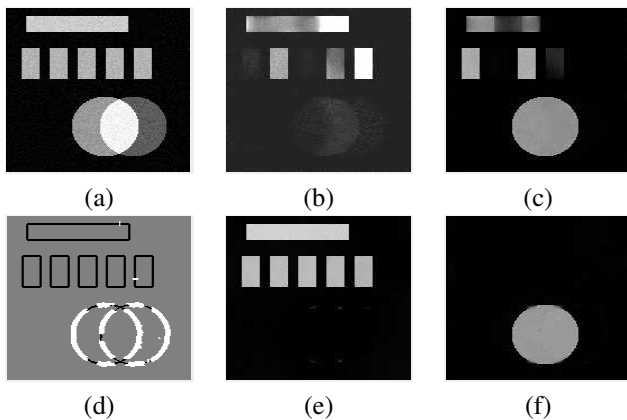
Fig. 9: Comparison of the results from ours and [12] on a synthesized image shown in (a). (b) Transmission layer T computed by [12]. (c) Reflection layer R computed by [12]. (d) Spatial weight matrix A computed by the proposed scheme. (e) Transmission layer T computed by our method. (f) Reflection layer R computed by our method.

TABLE I: PSNR and SSIM values by different methods on synthetic images. Left/right values are for the result with and without boundary regions being removed.

Metric	Li *et al.* [26]	Nikolaos *et al.* [10]	Shih *et al.* [12]	Ours
PSNR (dB)	14.48/14.28	17.77/17.72	17.48/18.29	20.43/20.55
SSIM	0.5694/0.5642	0.7862/0.7846	0.7820/0.7993	0.8098/0.8146

still outperformed by our method. In Fig. 11, some recovered results are visualized. Clearly, the results from our method have overall the best visual quality. The results from Li *et al.*[26] generally are darker caused by the issue of the method on color fidelity. Nikolaos *et al.*'s method [10] can preserve more details and achieve better color quality, but it failed to remove the reflections from the images. Both Shih *et al.*'s method [12] and our method performed better than others on removing reflection from images.

Compared with Shih *et al.*'s method [12], the proposed method shows better performance when processing the images with repetitive transmission objects, which leads to much less artifacts on the boundaries of objects. See Fig. 11 for the demonstration of the artifacts along the boundaries of objects in the results of Shih *et al.*'s method. We also recalculated the numerical results by cutting 25 pixels off around image boundaries. The recalculated results are also listed in Table I. It is found that discarding the issue on image boundaries, our method still outperformed Shih *et al.*'s method.

B. Results on real images from [12]

Three real images from [12] are used for evaluating our method. The results are shown in Fig. 12, 13, 14. It can be seen that our method outperforms the other three compared methods in terms of visual quality. In Fig. 12, the chimney at the top right is hardly removed by the methods of Li *et al.* [26] and Nikolaos *et al.* [10], while our method can remove the

TABLE II: PSNR(dB) under different thickness of glass on the SIR2-Postcard dataset.

	3mm	5mm	10mm	Average
Shih *et al.* [12]	19.60	19.29	19.50	19.46
Ours	19.72	19.35	19.63	19.57

chimney and yields consistent color at the upper right corner of the recovered image. Shih *et al.*'s method [12] can deal with the chimney well but has poor color consistency. Moreover, the result leave the long white roof and repetitive windows in the image, which indicates it did not wrongly assigned the ownership of repeating pattern in these regions. In contrast, our method achieved better visual results on the long white roof and repetitive windows. Similarly, in Fig. 13. The repetitive windows recovered by Shih *et al.* [12] are bad on both color consistency and structure, while our result has better visual quality. In Fig. 14, our method performed comparably with Shih *et al.*'s [12] and performed much better than that of Li *et al.* [26] and Nikolaos *et al.* [10].

C. Results on SIR2-Postcard dataset

We also evaluate our method on one real image dataset, SIR2-Postcard dataset [32], which uses postcards to compose 20 different controlled scenes. In the experiments, 20 groups of images are taken from SIR2, which are taken under three controlled thickness of glass (3mm, 5mm, and 10mm). Each group of images is composed of a triplet, which contains the mixture image, as well as the ground truth of background and reflection. The mixture image is captured through the thick glass, the ground truth of the reflection is captured with a sheet of black paper behind the glass, and the ground truth of the background is obtained by removing the glass. The numerical results are listed in Table II, where we show the average PSNR values on the dataset as well as on each glass thickness. It can be seen that our method have 0.11dB gain over Shih *et al.*'s method. Some results after resizing are shown in Fig. 15-Fig. 17, where the results from our method has better visual quality than that of the others.

V. SUMMARY

Based on the convolutional composite model for ghosting reflection, this paper proposed a regularization model for separating the transmission layer and reflection layer. The main issue we addressed here is how to determine the ownership of repeating patterns for two layers. Based on the observation that repeating times of such patterns are different between the two layers, a weighted wavelet transform based regularization is developed for effectively separating the two layers. The experiments showed that the proposed method outperformed the existing related ones.
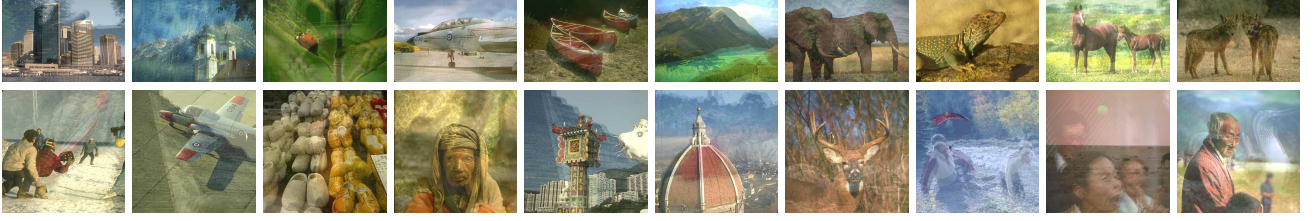
Fig. 10: Samples of synthetic images used for evaluation.

APPENDIX A
THE DETAIL OF NUMERICAL OPTIMIZATION SCHEME

The split-Bregman algorithm for solving (14) is as follows. First, the problem is rewritten as the constrained problem:

$$\min_{\boldsymbol{u}} \frac{1}{2}\|\boldsymbol{y} - \boldsymbol{H}\boldsymbol{u}\|_F^2 + \|\boldsymbol{d}\|_1, \quad \text{s.t.} \quad \boldsymbol{d} = \boldsymbol{D}\boldsymbol{u}, \qquad (15)$$

which is the "splitting" step to separate $\boldsymbol{u}$ from the ℓ_1 penalty. Next, the Bregman iteration is used to solve (15) as follows:

$$\begin{cases} (\boldsymbol{u}^{(k+1)}, \boldsymbol{d}^{(k+1)}) = \underset{\boldsymbol{u},\boldsymbol{d}}{\arg\min} \ \frac{1}{2}\|\boldsymbol{H}\boldsymbol{u} - \boldsymbol{y}\|_2^2 + \\ \qquad \|\boldsymbol{d}\|_1 + \frac{\gamma_1}{2}\|\boldsymbol{D}\boldsymbol{u} - \boldsymbol{d} + \boldsymbol{b}^{(k)}\|_2^2 \\ \boldsymbol{b}^{(k+1)} = \boldsymbol{b}^{(k)} + \gamma_2(\boldsymbol{D}\boldsymbol{u}^{(k+1)} - \boldsymbol{d}^{(k+1)}) \end{cases} \qquad (16)$$

for $k = 0, 1, \cdots$, where $\gamma_1, \gamma_2 > 0$ are two parameters arising from the Bregman iteration. By decomposing its first subproblem into two sub-problems, we rewrite (16) as follows:

$$\begin{cases} \boldsymbol{u}^{(k+1)} = \underset{\boldsymbol{u}}{\arg\min} \ \frac{1}{2}\|\boldsymbol{H}\boldsymbol{u} - \boldsymbol{y}\|_2^2 + \frac{\gamma_1}{2}\|\boldsymbol{D}\boldsymbol{u} - \boldsymbol{d}^{(k)} + \boldsymbol{b}^{(k)}\|_2^2 \\ \boldsymbol{d}^{(k+1)} = \underset{\boldsymbol{d}}{\arg\min} \ \|\boldsymbol{d}\|_1 + \frac{\gamma_1}{2}\|\boldsymbol{D}\boldsymbol{u}^{(k+1)} - \boldsymbol{d} + \boldsymbol{b}^{(k)}\|_2^2 \\ \boldsymbol{b}^{(k+1)} = \boldsymbol{b}^{(k)} + \gamma_2(\boldsymbol{D}\boldsymbol{u}^{(k+1)} - \boldsymbol{d}^{(k+1)}) \end{cases}$$

the first sub-problem has an analytic solution:

$$\boldsymbol{u}^{(k+1)} = (\boldsymbol{H}^\top \boldsymbol{H} + \gamma_1 \boldsymbol{D}^\top \boldsymbol{D})^{-1}(\boldsymbol{D}^\top \boldsymbol{y} + \gamma_2 \boldsymbol{D}^\top(\boldsymbol{d}^{(k)} - \boldsymbol{b}^{(k)})),$$

which is calculated by the conjugate gradient method. The second sub-problem also has the analytic solution given by

$$\boldsymbol{d}^{(k+1)} = \mathcal{S}_{\frac{1}{\gamma_1}}(\boldsymbol{D}\boldsymbol{u}^{(k+1)} + \boldsymbol{b}^{(k)}), \qquad (17)$$

where $\mathcal{S}_\beta(\cdot)$ is the soft-thresholding operation defined by

$$\mathcal{S}_\beta(\boldsymbol{x}) = \text{sgn}(\boldsymbol{x})\max(|\boldsymbol{x}| - \beta, 0). \qquad (18)$$

Combining all together, the problem (14) is solved by

$$\begin{cases} \boldsymbol{u}^{(k+1)} = (\boldsymbol{H}^\top \boldsymbol{H} + \gamma_1 \boldsymbol{D}^\top \boldsymbol{D})^{-1} \\ \qquad\qquad (\boldsymbol{H}^\top \boldsymbol{y} + \gamma_1 \boldsymbol{D}^\top(\boldsymbol{d}^{(k)} - \boldsymbol{b}^{(k)})) \\ \boldsymbol{d}^{(k+1)} = \mathcal{S}_{1/\gamma_1}(\boldsymbol{D}\boldsymbol{u}^{(k+1)} + \boldsymbol{b}^{(k)}) \\ \boldsymbol{b}^{(k+1)} = \boldsymbol{b}^{(k)} + \gamma_2(\boldsymbol{D}\boldsymbol{u}^{(k+1)} - \boldsymbol{d}^{(k+1)}) \end{cases}.$$

REFERENCES

[1] A. Agrawal, R. Raskar, S. K. Nayar, and Y. Li, "Removing photography artifacts using gradient projection and flash-exposure sampling," *IEEE Trans. Graphics*, vol. 24, no. 3, pp. 828–835, 2005.

[2] A. Levin and Y. Weiss, "User assisted separation of reflections from a single image using a sparsity prior," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 9, 2007.

[3] S. N. Sinha, J. Kopf, M. Goesele, D. Scharstein, and R. Szeliski, "Image-based rendering for scenes with reflections." *IEEE Trans. Graphics*, vol. 31, no. 4, pp. 100–1, 2012.

[4] X. Guo, X. Cao, and Y. Ma, "Robust separation of reflection from multiple images," in *Proc. Conf. Comput. Vision and Pattern Recognition*. IEEE, 2014, pp. 2187–2194.

[5] L. Bedini, P. Savino, and A. Tonazzini, "Removing achromatic reflections from color images with application to artwork imaging," in *Image Signal Process. Anal.* IEEE, 2015, pp. 126–130.

[6] Y. Li and M. S. Brown, "Exploiting reflection change for automatic reflection removal," in *Proc. Int. Conf. Comput. Vision*. IEEE, 2013, pp. 2432–2439.

[7] C. Sun, S. Liu, T. Yang, B. Zeng, Z. Wang, and G. Liu, "Automatic reflection removal using gradient intensity and motion cues," in *Proc. Multimedia Conf.*, 2016, pp. 466–470.

[8] B.-J. Han and J.-Y. Sim, "Reflection removal using low-rank matrix completion." in *Proc. Conf. Comput. Vision and Pattern Recognition*. IEEE, 2017.

[9] T. Xue, M. Rubinstein, C. Liu, and W. T. Freeman, "A computational approach for obstruction-free photography," *IEEE Trans. Graphics*, vol. 34, no. 4, p. 79, 2015.

[10] N. Arvanitopoulos, R. Achanta, and S. Süsstrunk, "Single image reflection suppression." in *Proc. Conf. Comput. Vision and Pattern Recognition*. IEEE, 2017, pp. 1752–1760.

[11] Y. Diamant and Y. Y. Schechner, "Overcoming visual reverberations," in *Proc. Conf. Comput. Vision and Pattern Recognition*. IEEE, 2008.

[12] Y. Shih, D. Krishnan, F. Durand, and W. T. Freeman, "Reflection removal using ghosting cues," in *Proc. Conf. Comput. Vision and Pattern Recognition*. IEEE, 2015, pp. 3193–3201.

[13] A. Levin, A. Zomet, and Y. Weiss, "Separating reflections from a single image using local features," in *Proc. Conf. Comput. Vision and Pattern Recognition*, vol. 1. IEEE, 2004, pp. I–I.

[14] C. Simon and I. Kyu Park, "Reflection removal for in-vehicle black box videos," in *Proc. Conf. Comput. Vision and Pattern Recognition*. IEEE, 2015, pp. 4231–4239.

[15] R. Wan, B. Shi, A.-H. Tan, and A. C. Kot, "Sparsity based reflection removal using external patch search," in *Proc. Int. Conf. Multimedia Expo*. IEEE, 2017, pp. 1500–1505.

[16] H. Farid and E. H. Adelson, "Separating reflections and lighting using independent components analysis," in *Proc. Conf. Comput. Vision and Pattern Recognition*, vol. 1. IEEE, 1999, pp. 262–267.

[17] R. Szeliski, S. Avidan, and P. Anandan, "Layer extraction from multiple images containing reflections and transparency," in *Proc. Conf. Comput. Vision and Pattern Recognition*, vol. 1. IEEE, 2000, pp. 246–253.

[18] B. Sarel and M. Irani, "Separating transparent layers through layer information exchange," *Proc. European Conf. Comput. Vision*, pp. 328–341, 2004.

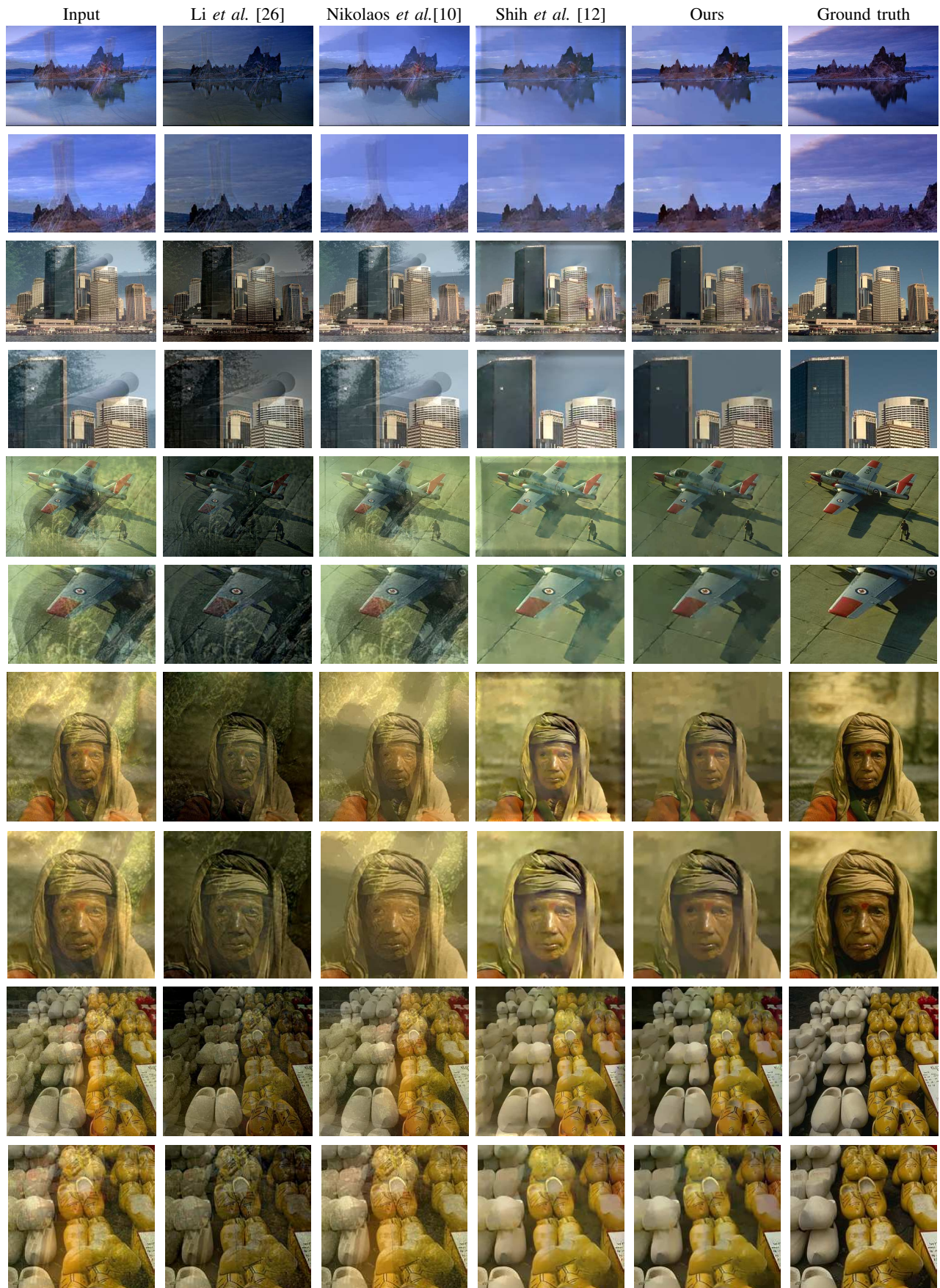| Input | Li *et al.* [26] | Nikolaos *et al.*[10] | Shih *et al.* [12] | Ours | Ground truth |

Fig. 11: The results on synthetic images. The odd/even rows show the whole/zoomed-in results.
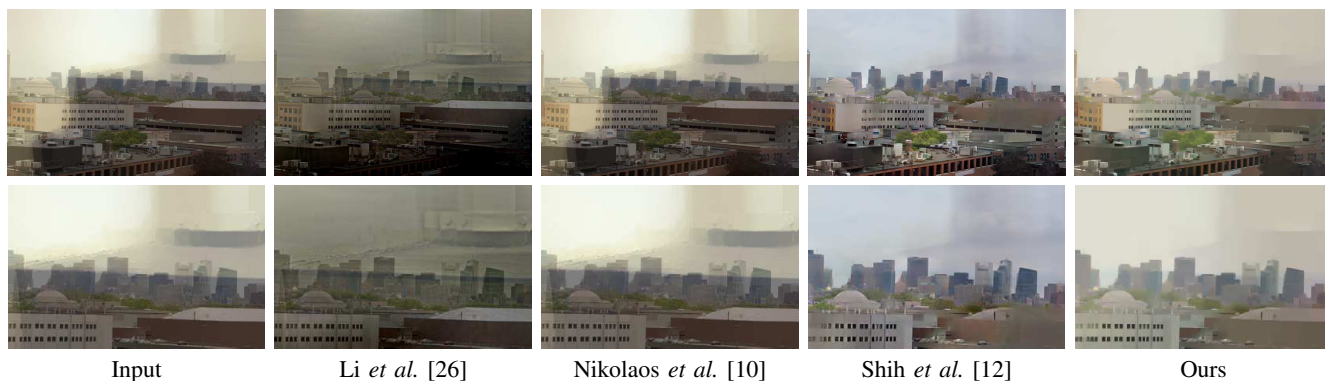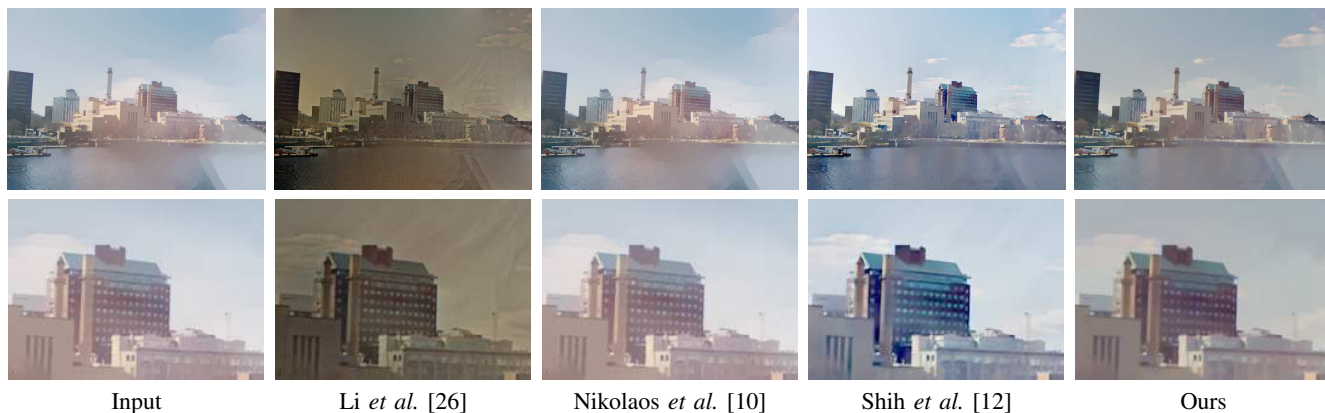
Fig. 12: Results on image "Factory".

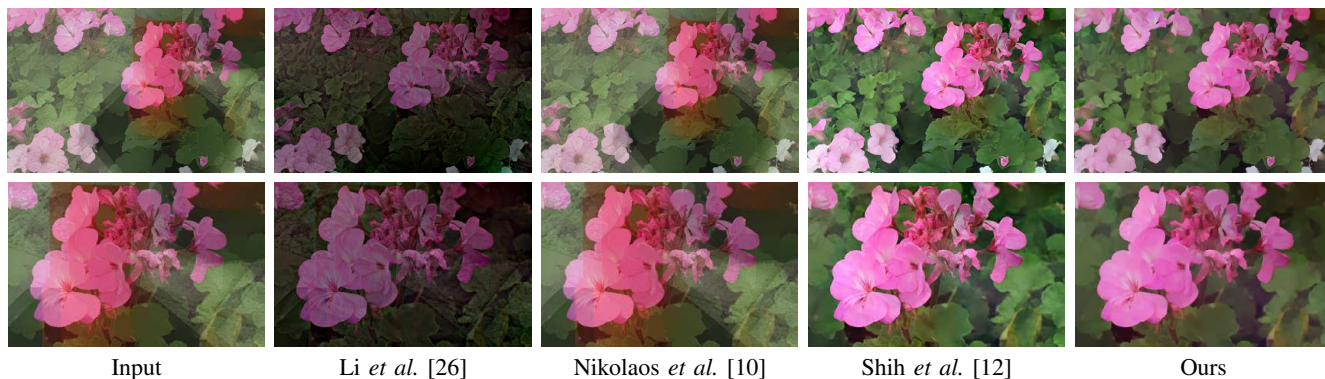
Fig. 13: Results on image "Lake".

Fig. 14: Reflection results on image "Flower".

Fig. 15: Results on image "Church".
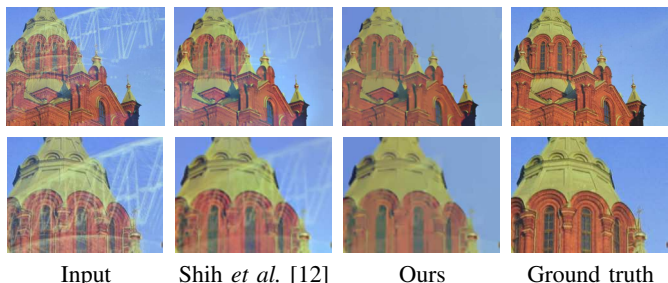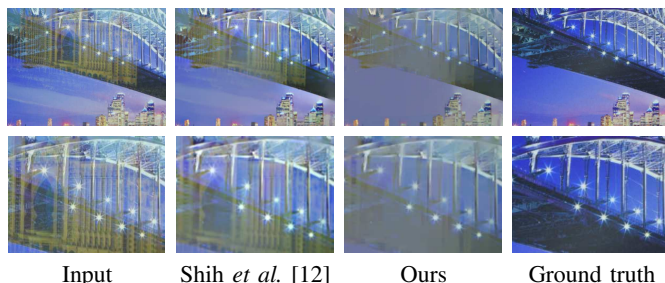

Fig. 16: Results on image "Bridge".
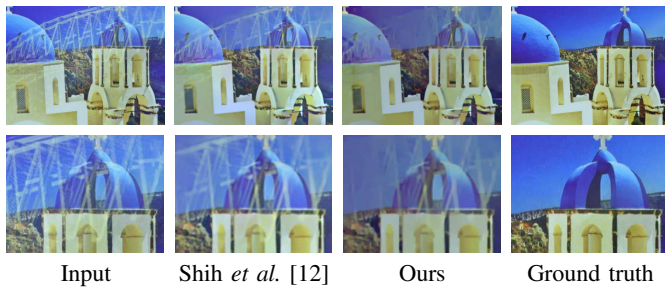
| Input | Shih *et al.* [12] | Ours | Ground truth |

Fig. 17: Reflection results on image "House".

[19] Y. Ni, J. Chen, and L.-P. Chau, "Reflection removal based on single light field capture," in *Proc. Int. Symp. on Circuits and Systems*. IEEE, 2017, pp. 1–4.

[20] K. Gai, Z. Shi, and C. Zhang, "Blind separation of superimposed moving images using image statistics," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 1, pp. 19–32, 2012.

[21] J. Y. Cheong, C. Simon, C.-S. Kim, and I. K. Park, "Reflection removal under fast forward camera motion," *IEEE Trans. Image Process.*, vol. 26, no. 12, pp. 6061–6073, 2017.

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

[23] J. Yang, H. Li, Y. Dai, and R. T. Tan, "Robust optical flow estimation of double-layer images under transparency or reflection," in *Proc. Int. Conf. Comput. Vision*. IEEE, 2016, pp. 1410–1419.

[24] A. Nandoriya, M. Elgharib, C. Kim, M. Hefeeda, and W. Matusik, "Video reflection removal through spatio-temporal optimization," in *Proc. Int. Conf. Comput. Vision*. IEEE, 2017, pp. 2430–2438.

[25] A. Levin and Y. Weiss, "User assisted separation of reflections from a single image using a sparsity prior," in *Proc. European Conf. Comput. Vision*. Springer, 2004, pp. 602–613.

[26] Y. Li and M. S. Brown, "Single image layer separation using relative smoothness," in *Proc. Conf. Comput. Vision and Pattern Recognition*. IEEE, 2014, pp. 2752–2759.

[27] R. Wan, B. Shi, T. A. Hwee, and A. C. Kot, "Depth of field guided reflection removal," in *Proc. Int. Conf. Image Process.* IEEE, 2016, pp. 21–25.

[28] Q. Fan, J. Yang, G. Hua, B. Chen, and D. P. Wipf, "A generic deep architecture for single image reflection removal and image smoothing." in *Proc. Int. Conf. Comput. Vision*. IEEE, 2017, pp. 3258–3267.

[29] R. Wan, B. Shi, L.-Y. Duan, A.-H. Tan, and A. C. Kot, "Crrn: Multi-scale guided concurrent reflection removal network," in *Proc. Conf. Comput. Vision and Pattern Recognition*. IEEE, 2018, pp. 4777–4785.

[30] Z. Chi, X. Wu, X. Shu, and J. Gu, "Single image reflection removal using deep encoder-decoder network," *arXiv preprint arXiv:1802.00094*, 2018.

[31] X. Zhang, R. Ng, and Q. Chen, "Single image reflection separation with perceptual losses," in *Proc. Conf. Comput. Vision and Pattern Recognition*. IEEE, June 2018.

[32] R. Wan, B. Shi, L.-Y. Duan, A.-H. Tan, and A. C. Kot, "Benchmarking single-image reflection removal algorithms," in *Proc. Int. Conf. Comput. Vision*. IEEE, 2017.

[33] D. Zoran and Y. Weiss, "From learning models of natural image patches to whole image restoration," in *Proc. Int. Conf. Comput. Vision*. IEEE, 2011, pp. 479–486.

[34] K. K. Hadziabdic, J. H. Telalovic, and R. Mantiuk, "Comparison of deghosting algorithms for multi-exposure high dynamic range imaging," in *Proc. Spring Conf. on Comput. Graphics*. ACM, 2013, pp. 21–28.

[35] K. Karaduzovic-Hadziabdic, J. H. Telalovic, and R. Mantiuk, "Subjective and objective evaluation of multi-exposure high dynamic range image deghosting methods," 2016.

[36] W. Zhang and W.-K. Cham, "Gradient-directed composition of multi-exposure images," in *Proc. Conf. Comput. Vision and Pattern Recognition*. IEEE, 2010, pp. 530–536.

[37] H. Ji, J. Li, Z. Shen, and K. Wang, "Image deconvolution using a characterization of sharp images in wavelet domain," *Appl. Compu. Harmonic Anal.*, vol. 32, no. 2, pp. 295–304, 2012.

[38] J.-F. Cai, H. Ji, C. Liu, and Z. Shen, "Framelet-based blind motion deblurring from a single image," *IEEE Trans. Image Process*, vol. 21, no. 2, pp. 562–572, 2012.

[39] B. Dong, H. Ji, J. Li, S. Zuowei, and X. Yuhong, "Blind motion deblurring using multiple images," *Appl. Compu. Harmonic Anal.*, vol. 32, no. 2, pp. 268–279, 2012.

[40] Y. Quan, H. Ji, and Z. Shen, "Data-driven multi-scale non-local wavelet frame construction and image recovery," *J. Sci. Comput.*, vol. 63, no. 2, pp. 307–329, 2015.

[41] I. Daubechies, B. Han, A. Ron, and Z. Shen, "Framelets: Mra-based constructions of wavelet frames," *Appl. Compu. Harmonic Anal.*, vol. 14, no. 1, pp. 1–46, 2003.

[42] T. Goldstein and S. Osher, "The split bregman method for l1-regularized problems," *SIAM J. Imaging Sci.*, vol. 2, no. 2, pp. 323–343, 2009.

[43] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Proc. Int. Conf. Comput. Vision*, vol. 2. IEEE, 2001, pp. 416–423.

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

Yan Huang received the Ph.D. degree in Computer Science from South China University of Technology in 2018. She is currently the postdoctoral research fellow in Computer Science from South China University of Technology. Her research interests include image processing and sparse representation.

Yuhui Quan received the Ph.D. degree in Computer Science from South China University of Technology in 2013. He worked as the postdoctoral research fellow in Mathematics at National University of Singapore from 2013 to 2016. He is currently the associate professor at School of Computer Science and Engineering in South China University of Technology. His research interests include computer vision, image processing and sparse representation.

Yong Xu received the B.S., M.S., and Ph.D. degrees in mathematics from Nanjing University, Nanjing, China, in 1993, 1996, and 1999, respectively. He was a Post-Doctoral Research Fellow of computer science with South China University of Technology, Guangzhou, China, from 1999 to 2001, where he became a Faculty Member and where he is currently a Professor with the School of Computer Science and Engineering. His current research interests include image analysis, video recognition, and image quality assessment. Dr. Xu is a member of the IEEE Computer Society and the ACM.

Ruotao Xu received the B.Eng degree in Computer Science from South China University of Technology in 2015. He is currently a PH.D candidate in South China University of Technology. His research interests include computer vision, image processing, and sparse coding.

Hui Ji received the B.Sc. degree in Mathematics from Nanjing University in China, the M.Sc. degree in Mathematics from National University of Singapore and the Ph.D. degree in Computer Science from the University of Maryland, College Park. In 2006, he joined National University of Singapore as an assistant professor in Mathematics. Currently, he is an associate professor in mathematics at National University of Singapore. His research interests include computational harmonic analysis, optimization, computational vision, image processing and biological imaging.