Blurriness-Guided Unsharp Masking

Wei Ye and Kai-Kuang Ma, *Fellow, IEEE*

***Abstract*— In this paper, a highly-adaptive *unsharp masking* (UM) method is proposed and called the *blurriness-guided* UM, or BUM, in short. The proposed BUM exploits the estimated local *blurriness* as the guidance information to perform pixel-wise enhancement. The consideration of local blurriness is motivated by the fact that enhancing a highly-sharp or a highly-blurred image region is undesirable, since this could easily yield unpleasant image artifacts due to over-enhancement or noise enhancement, respectively. Our proposed BUM algorithm has two powerful adaptations as follows. First, the enhancement strength is adjusted for *each* pixel on the input image according to the degree of local blurriness measured at the local region of this pixel's location. All such measurements collectively form the *blurriness map*, from which the *scaling matrix* can be obtained using our proposed mapping process. Second, we also consider the type of layer-decomposition filter exploited for generating the base layer and the detail layer, since this consideration would effectively help to prevent over-enhancement artifacts. In this paper, the layer-decomposition filter is considered from the viewpoint of *edge-preserving* type versus *non-edge-preserving* type. Extensive simulations experimented on various test images have clearly demonstrated that our proposed BUM is able to consistently yield superior enhanced images with better perceptual quality to that of using a fixed enhancement strength or other state-of-the-art adaptive UM methods.**

***Index Terms*— Image enhancement, unsharp masking, blurriness estimation, layer decomposition.**

I. Introduction

UNSHARP MASKING (UM) [1] is one of the most widely-used image enhancement techniques for improving the sharpness of image. It has become an indispensable image processing tool in digital photo editing and is often a built-in processing module in digital cameras, printers, and displays [2]. The UM enhancement framework can be summarized as a two-stage process as follows. In the first stage, the input image is decomposed into two layers by applying a linear shift-invariant *low-pass* filter (e.g., the Gaussian filter). The resultant filtered image is called the *base layer*, containing the main structure of the input image. On the other hand, the difference yielded between the base layer and the input image is called the *detail layer*, revealing the fine details of the input image. Alternatively, a linear shift-invariant *high-pass* filter (e.g., the Laplacian filter) can be exploited to extract the detail layer first, followed by subtracting it from the input image to generate the base layer. In the second stage, the obtained detail layer from either of the above-mentioned approaches is amplified by multiplying a scaling factor and then added back to the base layer to generate the enhanced version of the input image.

In the conventional UM, a *fixed* scaling factor is employed for amplifying the detail layer. That is, all the pixels on the detail layer are treated *equally* important and receive the same amount of image enhancement. One can intuitively sense that such fixed scaling practice is not always appropriate, because the contents of the input image, generally speaking, are quite dynamic and oftentimes involve high-level semantic aspects (e.g., photographer's intention). All these are difficult to be captured by low-level pixel-based image processing. This motivates us to develop an *adaptive* scaling method for the UM process so that each pixel will have its own scaling factor, even including no enhancement at all.

To compute the scaling factor for each pixel, the information-guiding approach is to exploit certain image characteristics or attributes extracted from the input image to conduct image enhancement. Most of the existing methods developed along this direction (e.g., [3]–[8]) utilize the local *contrast* as the attribute to determine the enhancement strength for each pixel. However, our studies have shown that these methods could lead to under-enhancement (i.e., insufficient amount of enhancement) and/or over-enhancement (i.e., unwanted amplification on enhancement) across various regions on the enhanced image. The former is often incurred for those regions with low-contrast details, while the latter most likely presented along sharp edges.

Rather than using the local *contrast*, we consider the local *blurriness* is the most relevant and effective attribute to exploit for guiding the image enhancement process. The use of this low-level feature is motivated by the following keen observations. First, we have observed that enhancing a highly-blurred image region often leads to unsatisfactory enhancement results. For example, the granular noise of a blurred and noisy background region as demonstrated in Fig. 1 (blue-framed region) has been undesirably amplified and become more distinct. Furthermore, it is also important to note that the two patches highlighted in Fig. 1 have nearly identical local contrast measurement; to be exact, the average gradient magnitudes of both patches are approximately equal to five intensity levels. This means that it is difficult to distinguish the foreground and background regions simply based on the measured local *contrast* information. On the other hand, the local

Manuscript received July 2, 2017; revised January 9, 2018 and April 15, 2018; accepted May 6, 2018. Date of publication May 21, 2018; date of current version June 11, 2018. This paper was presented at the 2017 IEEE International Conference on Image Processing, Beijing, China, September 17-20, 2017. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Gene Cheung. *(Corresponding author: Kai-Kuang Ma.)*

The authors are with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798 (e-mail: ye0003ei@e.ntu.edu.sg; ekkma@ntu.edu.sg).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Fig. 1. Conventional unsharp masking (UM) enhancement results of two image patches—one cropped from the foreground object (red-framed) and the other from the highly-blurred background region (blue-framed), respectively. One can see that enhancing the foreground object gives more visually desirable result, while this is not the case for the enhancement of blurred background region, where more distinct granular noise is presented.

blurriness plays an effective role to discriminate these two regions so that various enhancement treatments can be applied to them individually. This approach could effectively address a key issue commonly existing in all UM methods—i.e., how to address over-enhancement, under-enhancement, and unwanted noise enhancement within the same picture simultaneously and with a unified treatment.

Second, even if the given highly-blurred image region is noise-free, the sharpness of its image details could be too degraded to be enhanced by applying any image enhancement technique. In this case, a proper image *restoration* (rather than *enhancement*) method should be deployed instead. Lastly, the blurred regions could be purposely created by the photographer in the first place in order to highlight the main object and create aesthetic bokeh effect, such as the background region as shown in Fig. 1. In this case, it becomes totally undesirable to enhance those blurred regions.

Based on the above-mentioned discussions, it is highly convinced that the image's *blurriness* information measured at each pixel can be utilized as an effective guidance information to guide the image enhancement process such that those highly-blurred or highly-sharp regions will be less enhanced, or even receive no enhancement at all. To accomplish this objective, it is critical to first conduct pixel-wise blurriness estimation for the image under enhancement, and then utilize this information to adapt enhancement strength for each pixel. Existing blurriness estimation approaches are unable to deliver pixel-wise estimation with high accuracy as required by our enhancement task. To tackle this problem, an effective blurriness estimation approach is proposed to generate more accurate *blurriness map* that is to be converted to a scaling-factor matrix via a mapping process.

The rest of this paper is organized as follows. Section II provides a succinct review of existing unsharp masking algorithms and blurriness estimation approaches. Section III describes the proposed pixel-wise blurriness estimation approach, and the proposed BUM framework is then presented in Section IV. Experimental results, particularly on the subject performance evaluation, are provided in Section V. Conclusions are drawn in Section VI.

II. Related Works

A. Overview of Existing Unsharp Masking (UM) Methods

Extensive investigations have been conducted in the past for improving the conventional UM's performance (e.g., [3]–[15]). However, these methods mainly address the issues of under-enhancement and over-enhancement. Generally speaking, the improvements proposed in these advanced UM techniques could come from two aspects: 1) more sophisticated layer-decomposition filter for generating the base layer and the detail layer; and 2) adaptive scaling on the detail layer for amplifying the extracted details. These attempts are highlighted in the following.

For the layer-decomposition filtering, various *edge-preserving* filters (e.g., [9]–[15]) have been developed to replace the conventional UM's linear shift-invariant *low*-pass filter. These filters are fairly effective on avoiding the overshoot artifacts caused by the *over*-enhancement normally incurred on sharp edges. When such filters are applied to the input image, those distinct edges will be largely preserved in the resultant filtered image (i.e., the base layer). This means that the extracted details (i.e., the differences between the base layer and the original image) at the corresponding pixel locations on the detail layer will be quite small. As a result, less enhancement will be imposed at these pixels when a fixed scaling factor is employed for amplifying the detail layer.

For another implementation of the conventional UM method using a *high*-pass filter to extract the detail layer as mentioned in Section I, several *nonlinear* polynomial filters (e.g., [3]–[5]) have been proposed to replace the conventional linear shift-invariant high-pass filter. These polynomial filters can be viewed as multiplying a linear Laplacian filter with a weighting factor computed from the input image, which could be the local intensity mean [3], the square of local gradients [5], or the product of both [4]. With this weighting factor, these filters will extract less details on those dark regions (with low pixel-intensity mean) and/or those flat regions (with low gradient magnitude), where the noise tends to occur. Consequently, they are able to alleviate the noise amplification issue.

For the design of adaptive scaling scheme, several pixel-classification-based methods (e.g., [6], [7]) have been proposed, in which the pixels of the input image are first classified into different classes, and each class is then assigned with a fixed scaling factor. For example, the method presented in [6] utilizes the *local variance* computed from the input image to classify the pixels into three classes—i.e., low-, medium-, or high-contrast pixels. For the assignment of a fixed scaling factor for each class, the low- and high-contrast classes are assigned with smaller scaling factors than that of the medium-contrast class, since it has been observed that enhancing the pixels from these two classes often leads to noise amplification (for the low-contrast case) and overshoot artifacts (for the high-contrast case), respectively. In [7], a similar *contrast*-based pixel classification scheme is also developed, which is based on quantizing the filtered result of the input image using the *Laplacian of Gaussian* (LoG) filter. Unlike most existing UM methods that require all scaling factors to be greater

than or equal to one, this method allows the scaling factor to be less than one for those pixels in flat regions such that the image details presented by these pixels, which are noisy in many cases, will be suppressed, rather than being enhanced.

In addition to the above-mentioned methods, other advanced UM techniques have attempted to improve image enhancement performance through both layer-decomposition filtering stage and detail-layer's scaling stage. For example, in [16], an adaptive bilateral filter is developed for conducting the layer decomposition, together with an adaptive scaling scheme similar to the one as proposed in [7]. More recently, a generalized UM algorithm is proposed in [8] that exploits an iterative edge-preserving median filter in the layer-decomposition stage. Besides, the scaling factor at each pixel is computed based on the pixel intensity of the detail layer; the larger the intensity value, the smaller the scaling factor to be imposed. These improvements help to alleviate the aforementioned overshoot artifacts.

We have observed that almost all of the above-mentioned UM methods commonly exploit the *local contrast* information to guide the enhancement process. Our earlier discussion in Section I has pointed out that the local contrast, in fact, is *not* an effective guidance information for conducting image enhancement task. In this paper, a more effective image attribute—*blurriness*, is investigated for conducting image enhancement instead. Therefore, in what follows, we shall survey existing local blurriness estimation methods and then describe our proposed estimation algorithm in Section III.

B. Overview of Existing Blurriness Estimation Methods

In order to utilize the blurriness information to guide the enhancement process, a blurriness estimation process is conducted for each pixel on the input image. There are two types of blurs existing in the natural images—i.e., *defocus blur* and *motion blur*. Since the objects being captured by a photographer's camera are static in most cases, therefore we limit our concern to defocus blur only in this paper.[1] Among the existing defocus-blur estimation methods (e.g., [17]–[21]), a blurred image is often mathematically modeled as the convolution of an ideal all-in-focus image with a point spread function (PSF) (also known as a *blur kernel*), which is typically assumed to be a spatially-varying disc function [20] or the Gaussian function [17]–[19], [21]. In this case, the pixel-wise blurriness estimation can be boiled down to estimating the parameters of the blur kernel (e.g., the radius of a disc function or the standard deviation of the Gaussian function).

In [17], the first- and second-order image derivatives are utilized to estimate the edge location and the amount of blur, respectively. The method proposed in [18] exploits the so-called *local contrast priors* to infer the blurriness at each pixel, followed by applying a Markov random field (MRF) based propagation process to refine the obtained blurriness map. In [19], the amount of blur is estimated from the gradient ratio computed between the input image and its Gaussian-blurred version. More recently, a sparse-representation-based blurriness estimation approach, called the *just noticeable blur estimation* (JNBE) [21], is developed that exploits the sparseness measured from each image patch to infer its blurriness. This method has shown state-of-the-art performance among all existing blurriness estimation algorithms. However, all the aforementioned blurriness estimation methods fail to deliver the estimation accuracy that is required by our blurriness-guided image enhancement task. Therefore, a new blurriness estimation method is proposed and described in the following section.

[1] In case the image was acquired with the involvement of motion blur, the image content could be too blurred and degraded to apply any image enhancement technique. In this case, image *restoration*, rather than *enhancement*, should be considered instead.

III. Proposed Pixel-Wise Blurriness Estimation

To design our blurriness-guided UM algorithm, the computed *blurriness map*, which is denoted by M, should have the same resolution as that of the input image and possess the following two properties or requirements. First, for each region at a specific level of focus, the estimated blurriness for all the pixels within this region should be a constant, or, at least, very close to each other numerically. In other words, this region should be perceived as a (nearly) uniform or flat region on M. Second, a sharp and accurate boundary should be yielded between two adjacent uniform regions on M that have different focus levels on I. Multiple blurriness estimation methods have been experimented (i.e., [19]–[22]), but none of them is able to produce a blurriness map possessing the above-mentioned two properties. To address this problem, a novel pixel-wise blurriness estimation algorithm is proposed in this paper, which consists of the following two sequential stages: 1) an initial blurriness map $\widehat{M}$ is first generated using the JNBE algorithm [21], and 2) the *weighted least square* (WLS) estimation method is then used to refine $\widehat{M}$ to get the final accurate blurriness map M. These two stages will be described in detail in the following sub-sections, respectively.

A. Stage 1: Initial Blurriness Estimation

For this stage, the *just noticeable blur estimation* (JNBE) algorithm [21] is first employed for generating the initial blurriness map $\widehat{M}$, simply because of its state-of-the-art performance. For a color input image I, the initial blurriness map $\widehat{M}$ will be estimated based on its luminance channel recorded in the CIELAB color space. The JNBE algorithm assumes that the blur kernel is a spatially-varying Gaussian function, and the standard deviation of the function is used as the blurriness measurement. This sparse-representation-based algorithm estimates the blurriness in a patch-wise manner. To do so, it needs to establish an over-complete dictionary by using a large set of Gaussian-blurred training image patches beforehand. In [21], a training set containing 100,000 patches is employed to train a dictionary containing 256 basis vectors via the well-known dictionary learning algorithm, called the *K singular value decomposition* (K-SVD) [23]. The training patches are randomly cropped from 1,000 natural images, and each patch is blurred by a Gaussian kernel with an empirically-determined standard deviation $\sigma = 2$ [21]. With such obtained dictionary, the blurriness of each test patch is estimated based

on a critical observation that the sparse representation of a *clear* (or *in-focus*) image patch tends to be less sparse than that of a blurred one. The rationale behind this observation is that each basis vector of the dictionary contains only smooth structures as the dictionary was constructed by learning from a set of Gaussian-blurred patches. Consequently, a clear patch (often containing sharp transitions) will generally require more dictionary basis vectors to represent it; thus, yielding a less-sparse representation.

Based on this observation, the blurriness of an image patch $y \in \mathbb{R}^{n\times 1}$ with n pixels can be inferred from the sparseness of its sparse representation coefficient $x \in \mathbb{R}^{k\times 1}$, which is obtained by solving

$$x = \arg\min \|x\|_0, \quad \text{subject to } \|y - Dx\|_2 \le \epsilon, \qquad (1)$$

where $D \in \mathbb{R}^{n\times k}$ is the aforementioned dictionary with k basis vectors and was established from a set of Gaussian-blurred patches. Symbols $\|\cdot\|_0$ and $\|\cdot\|_2$ denote the l_0 and l_2 norms of the vector indicated in the argument, respectively. The constant ϵ imposes Dx to be close enough to y for finding x ($\epsilon = 0.07$ was used in [21]).

After the vector x is obtained, its *sparseness* will be measured by $\|x\|_0$ (i.e., the number of non-zero elements contained in x). In order to use such sparseness measurement to derive the degree of blurriness for the target patch y, the JNBE algorithm further establishes a mapping function, which maps the sparseness $\|x\|_0$ into the blurriness measurement (i.e., the standard deviation σ of the underlying Gaussian blur kernel). This is accomplished with another set of blurred image patches obtained by applying Gaussian filtering with various σ values on clear image patches. For each blurred patch, its corresponding σ is known and considered as its *ground-truth* blurriness, while the sparseness of its sparse representation, measured by $\|x\|_0$, can be computed via (1) as described above. Through a logistic regression conducted on about 500,000 such patches, the above-mentioned mapping function is obtained, with which the blurriness of an image patch can be estimated. For each pixel p on the input image, the above-described blurriness estimation needs be conducted on a 8×8 patch surrounding the pixel p for computing the blurriness estimation $\hat{\sigma}_p$. Consequently, a blurriness map $\widehat{M}$, which is of the same size as the input image I, will be generated; that is, $\widehat{M}_p = \hat{\sigma}_p$. In our generated blurriness map, the darker pixels denote that they are more in-focus; hence, the whiter pixels indicate that they are more blurred.

However, the generated $\widehat{M}$ is very coarse (as demonstrated in Fig. 2 (b)) and far from satisfactory for being used to guide our image enhancement task. Specifically, it has two major problems as follows. First, since the JNBE estimation is essentially conducted in a patch-wise manner, it fails to yield accurate boundaries with pixel-wise accuracy among adjacent regions that have different focus levels. In this case, the background pixels could be mistakenly treated as the foreground ones and enhanced. As a result, artifacts surrounding the object's boundary will be introduced on the enhanced image. Second, when a clear patch is highly smooth, the JNBE method will mistakenly yield a high blurriness value for it, leading to 'white' regions within an in-focus object (which is supposed to appear as a dark region) in the generated $\widehat{M}$, as illustrated in Fig. 2 (b). This could cause under-enhancement issues for the image with low-contrast image details as demonstrated in such 'white' regions. To address these issues, a refinement step of $\widehat{M}$ is clearly needed and developed as follows.

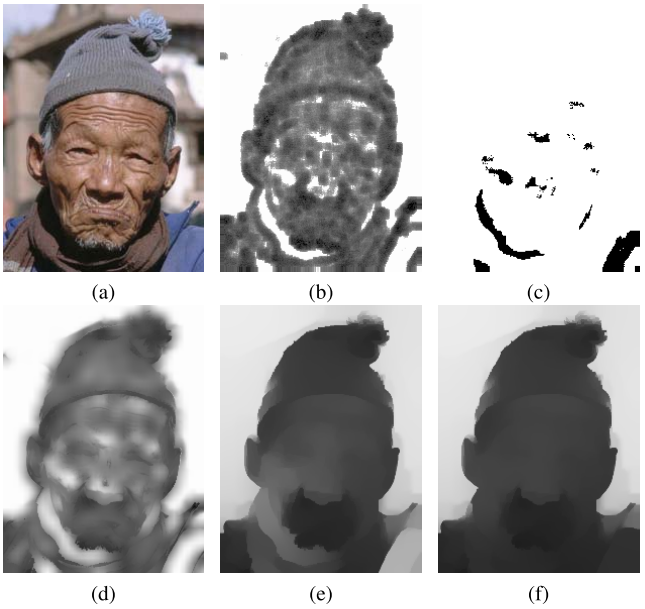


Fig. 2. Results of various blurriness map refinement schemes: (a) original image; (b) the initial blurriness map $\widehat{M}$ generated by using the JNBE algorithm [21]; (c) the confidence weightage map computed according to (3); (d) the refined map M by using the bilateral filter [10] with $\sigma_s = 5$ and $\sigma_r = 0.1$; (e) the refined map M obtained by using the WLS-based approach without the confidence weights; (f) the refined map M obtained by using the proposed WLS-based approach with the adaptive confidence weightage map as shown in (c).

B. Stage 2: Blurriness Map Refinement

In this work, the *weighted least square* (WLS) estimation method [24] is adopted to refine the coarse $\widehat{M}$ for generating a much improved blurriness map, denoted by M. Specifically, M is obtained by minimizing a cost functional $\mathcal{F}$ that is defined as [24]

$$\mathcal{F} = \sum_p \left\{ w_p \left(M_p - \widehat{M}_p\right)^2 + \lambda \left[h_p (\partial_x M)_p^2 + v_p (\partial_y M)_p^2 \right] \right\}, \quad (2)$$

where the subscript p denotes the pixel location. The partial differentiation operators ∂_x and ∂_y compute the derivatives along the horizontal (i.e., x-axis) and the vertical (i.e., y-axis) directions on M, respectively. The first term of (2) is the *data* term, which is used to prevent M from being deviated away from the initial map $\widehat{M}$ drastically. Meanwhile, the second term of (2) is the *smoothness* term, which forces M to be as smooth as possible. The constant λ provides a trade-off between these two terms, and $\lambda = 0.2$ is empirically set in this approach.

The *confidence weight* w_p in (2) indicates how confident of the initial blurriness estimation $\widehat{M}_p$ obtained at the pixel p. The main objective of introducing this weighting factor is to discard those unconfident (thus, unreliable) initial blurriness

estimations; in this case, the corresponding weighting factor can be set to zero. By doing so, the data term in (2) will take no effect, and the estimated blurriness will completely depend on the smoothness term. Similar weighting strategy has been practiced in some depth map upsampling and enhancement works (e.g., [25], [26]). Now, the issue is boiled down to how to identify these pixel locations that have unreliable blurriness estimation.

As previously mentioned, the JNBE algorithm often mistakenly yields such 'white' pixels (i.e., pixels with large blurriness) within objects that are actually in focus. Fortunately, the number of such 'white' pixels is usually quite small according to our observations. To identify these locations, the mean shift [27] segmentation algorithm is adopted and applied to $\widehat{M}$. Based on the segmentation results, w_p is determined as follows:

$$w_p = \begin{cases} 0, & \widehat{M}_p > \sigma_t \text{ and } size(R_p) < N_t; \\ 1, & \text{otherwise}, \end{cases} \tag{3}$$

where R_p denotes the segmented region containing the pixel p. Thresholds σ_t and N_t are empirically determined and used throughout all our experiments as follows: $\sigma_t = 1.8$ and $N_t = 2.5\% \cdot N$, where N is the total number of pixels of the input image I. Based on (3), $w_p = 0$ will be assigned to the pixel p, if its corresponding initial blurriness estimation $\widehat{M}_p$ is greater than σ_t and meanwhile it belongs to a small segmented region containing less than N_t pixels. An example of the confidence weightage map computed via (3) is demonstrated in Fig. 2 (c), from which one can see that the 'white' regions in Fig. 2 (b) have been successfully detected (via mean shift) and assigned with zero weights.

Furthermore, the *smoothness weights* h_p and v_p in (2) are imposed to adapt the amount of smoothing at each pixel location. They are determined by the input image I based on its gradients along the horizontal direction (i.e., x-axis) and vertical direction (i.e., y-axis); i.e., $\partial_x I$ and $\partial_y I$, respectively:

$$h_p = \left(\left|(\partial_x I)_p\right|^{\gamma} + \kappa\right)^{-1} \quad \text{and} \quad v_p = \left(\left|(\partial_y I)_p\right|^{\gamma} + \kappa\right)^{-1}, \tag{4}$$

where κ is a small constant number imposed for avoiding the divided-by-zero incidence, and the exponent γ controls the sensitivity of h_p and v_p to the above-mentioned gradients $(\partial_x I)_p$ and $(\partial_y I)_p$, respectively. In our approach, $\kappa = 10^{-5}$ and $\gamma = 2$ are used. According to (4), the smoothness weights will be small for those pixel locations that incur large gradients on I, thus imposing less amount of smoothing on $\widehat{M}$. This has the effect of preserving those edges on $\widehat{M}$ that correspond to sharp object boundaries on I.

By using the matrix notation, the objective functional as defined in (2) can be re-written as follows:

$$\begin{aligned} \mathcal{F}(M) = \mathcal{J}(V_M) &= (V_M - V_{\widehat{M}})^T W (V_M - V_{\widehat{M}}) \\ &+ \lambda \left(V_M^T D_x^T A_h D_x V_M + V_M^T D_y^T A_v D_y V_M \right), \end{aligned} \tag{5}$$

where V_M and $V_{\widehat{M}}$ are the vector form of M and $\widehat{M}$, respectively. Matrices D_x and D_x are the discrete (forward) differentiation operators ∂_x and ∂_y in the Toeplitz matrix form, respectively. Matrix W is a diagonal matrix containing the confidence weight w_p. Similarly, matrices A_h and A_v are also diagonal matrices, which contain the smoothness weights h_p and v_p, respectively. By letting $\partial \mathcal{J} / \partial V_M = 0$, the vector V_M that minimizes the above objective functional $\mathcal{J}(V_M)$ can be derived as

$$V_M = \left(W + \lambda D_x^T A_h D_x + \lambda D_y^T A_v D_y \right)^{-1} W V_{\widehat{M}}. \tag{6}$$

Equation (6) describes a sparse linear system, which can be solved using the *pre-conditioned conjugate gradients* (PCG) approach [28] as practiced in [24]. Finally, the obtained optimal vector V_M is converted back to the matrix form to yield the final refined blurriness map M.

It is worthwhile to mention that the bilateral filter [10] was originally suggested in the JNBE [21] for refining the initially-estimated blurriness map $\widehat{M}$. However, one can see from Fig. 2 (d) that the bilateral filter delivered a fairly poor refinement result M as it fails to filter out the noisy structures and yield sharp region boundaries. Other local smoothing filters, such as the guided filter [12] and the domain transform filter [14], have also been investigated, and they all produce similar poor results according to our simulations (but not presented here). On the other hand, the WLS-based approach is able to deliver a much better refined map M, even without the use of the confidence weighting (see Fig. 2 (e)). With further incorporating confidence weights, the final refined blurriness map is fairly smooth and meets our initially imposed goals about what the final blurriness estimation map should look like in order to facilitate the follow-up UM process.

IV. Proposed Blurriness-Guided Unsharp Masking (BUM) Algorithm

A. Overview

The framework of the proposed blurriness-guided unsharp masking method is illustrated in Fig. 3. If the input image I is a color image, it will be converted in the CIELAB color space, and the enhancement will be conducted only on its luminance component L. The enhanced luminance component will then be combined with the original two chrominance components, followed by converting them back to the RGB color space to generate the final enhanced color image for presentation.

In Fig. 3, the refined blurriness map M, which is obtained through the blurriness estimation process as described in the previous section, will be utilized as a guidance information to determine the scaling factor (i.e., enhancement strength) for each pixel. The entire enhancement process can be mathematically described as

$$\tilde{L} = B + \Lambda \otimes T, \tag{7}$$

where $\tilde{L}$ is the enhanced luminance component, B and T are the *base layer* and *detail layer* decomposed from the input luminance component L, respectively; i.e., $L = B + T$. Unlike many existing UM algorithms that employ a *fixed* scaling factor for all pixels on the detail layer T, an *adaptive* scaling factor matrix Λ is used in our approach to scale the detail layer T through a pixel-wise multiplication (denoted by the symbol $\otimes$).

Another novelty of the proposed BUM algorithm lies in its adaptation to the layer-decomposition filter used in the first

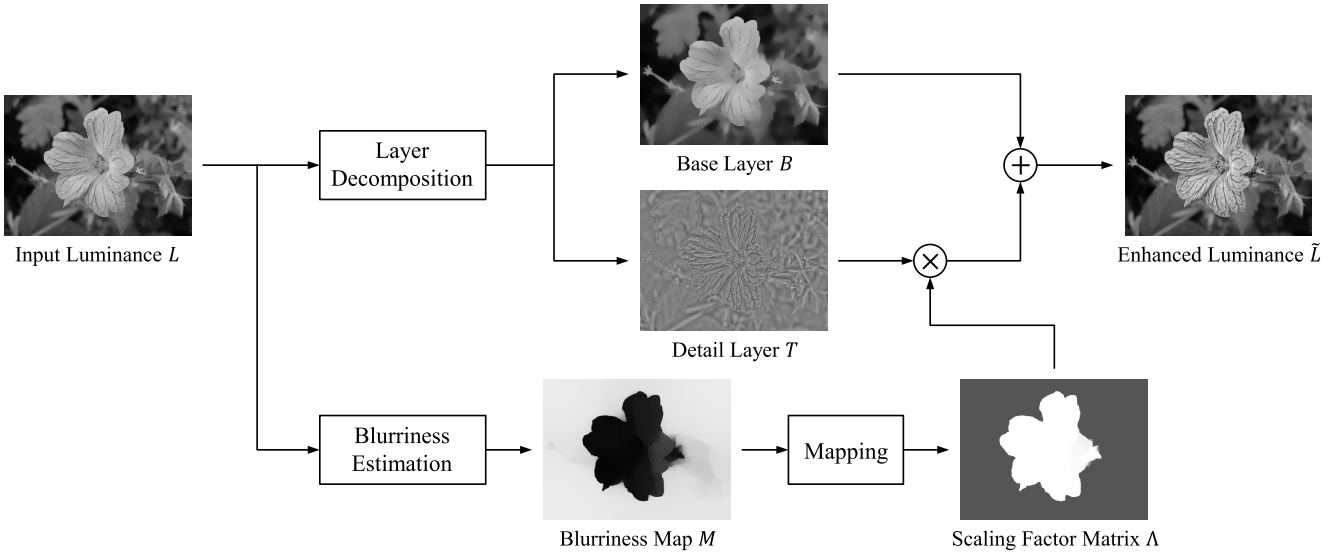


Fig. 3. The processing pipeline of the proposed *blurriness-guided adaptive unsharp masking* (BUM) method. For the color image, it is converted to the CIELAB color space first, and only the luminance component L will be enhanced. The layer-decomposition filter demonstrated here is an edge-preserving type of filter.

stage of the UM process on the generation of the base layer and the detail layer. Such consideration is important, as the type of filter used for decomposition is intimately related to the matrix Λ to be generated in the second stage. For that, we classify the layer-decomposition filters into two categories: *edge-preserving* filter (e.g., [9]–[15]) and *non-edge-preserving* filter (e.g., Gaussian filter, average filter). The rationale behind such classification is due to the fact that if an edge-preserving filter were used, more edge information will be kept on the base layer B. This means that less details will be left on the detail layer T (i.e., with smaller amplitudes). To yield effective enhancement, larger scaling factors are expected to be applied. However, the same scaling factors might be too large to be used for a detail layer that was generated by a non-edge-preserving filter, because more details with large amplitudes tend to appear on the detail layer. Hence, over-enhancement with unwanted image artifacts will be incurred in this case. This clearly justifies the need of adaptation on the computation of scaling factor matrix Λ from the perspective of the layer-decomposition filter type.

Since the scaling factor matrix Λ is obtained through a mapping process, two mapping algorithms are needed, one for each aforementioned layer-decomposition filter case. In our approach, if an edge-preserving filter were used in the layer decomposition stage, only the *blurriness* information M, computed from the input image I, will be used as the input of the mapping process (see Fig. 3). On the other hand, if a non-edge-preserving filter were used instead, the *contrast* information (computed from the detail layer T) will be incorporated as an extra guidance information besides the blurriness information. This will effectively avoid or alleviate the overshoot artifacts due to the less-sophisticated non-edge-preserving filter used for decomposition. Note that this part is not shown in Fig. 3 for simplifying the block-diagram drawing, and it can be easily understood and added into Fig. 3 according to the more detailed descriptions in what follows.

B. Case 1: Scaling Factor Matrix Computation for Edge-Preserving Filter

As highlighted previously, if an edge-preserving filter (e.g., [9]–[15]) were used in the layer-decomposition stage, only the blurriness information is exploited for the computation of the scaling factor matrix Λ. As explained in Section I, enhancing a highly-blurred image region is undesirable, because it is quite unlikely to yield an appreciable quality improvement, while often getting unwanted artifacts in return. Therefore, it is more appropriate to assign smaller scaling factors to the pixels in those highly-blurred regions to impose less amount of enhancement, or even no enhancement at all. To accomplish this objective, a mapping process based on the *generalized Gaussian function* is proposed to translate the blurriness estimation map M_p into a scaling factor Λ_p for each pixel location p. Our proposed mapping function is defined as

$$\Lambda_p = \rho \cdot \mathcal{G}_b(M_p) + \eta, \tag{8}$$

where

$$\mathcal{G}_b(M_p) = e^{-\left(\frac{M_p}{\alpha}\right)^{\beta}}. \tag{9}$$

The parameters α and β determine the *scale* and the *shape* of the function $\mathcal{G}_b(M_p)$, respectively, as shown in Fig. 4 (a) under different settings of these parameters. One can see that the scale parameter α determines how wide the range of blurriness will be assigned with large scaling factors, while the shape parameter β controls how fast of the transition changing from the higher values of the scaling factors to the lower ones. On the other hand, since $\mathcal{G}_b(M_p) \in (0, 1]$, the obtained Λ_p will fall into the range of $(\eta, \rho + \eta]$. This means that ρ and η decide the maximum and minimum scaling factors (i.e., the dynamic range) that could be imposed by (8). Setting large values to these two parameters will clearly yield more perceivable enhancement; however, this might also cause undesirable over-enhancement artifacts more easily in return.

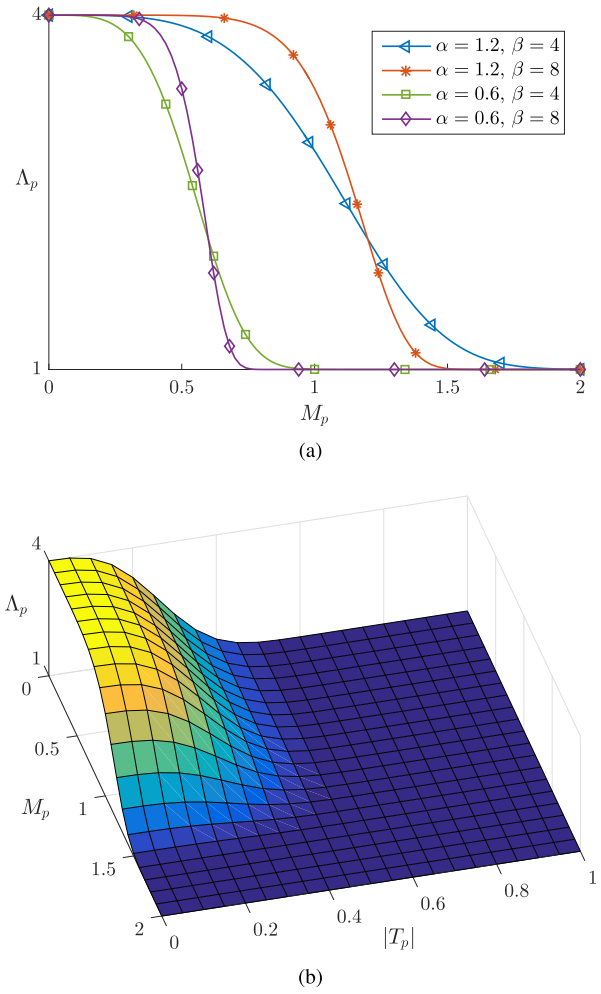


Fig. 4. An illustration of the proposed mapping functions for generating the adaptive scaling factor matrix Λ: (a) the blurriness-guided mapping function as defined in (8) for Case 1, when an edge-preserving filter is used; (b) the blurriness- and contrast-guided mapping function as defined in (10) for Case 2, when non-edge-preserving filter is used. Note that $\rho = 3$, $\eta = 1$ are used for generating the above plots, thus $\Lambda_p \in (1, 4]$ in both cases.

In practice, there is no optimal setting for the above-mentioned parameters (i.e., α, β, ρ, and η), and their values should be determined based on the requirements of each specific application or up to the user's preference. For our enhancement task conducted on common natural images, it has been found through extensive simulations that using $0.8 \leq \alpha \leq 1.4$ and $4 \leq \beta \leq 8$ generally yields visually pleasant enhancement results. As for ρ, we have found that letting $1 \leq \rho \leq 3$ is able to deliver fairly perceivable improvement on image sharpness, while not yielding the overshoot artifacts in most cases. Finally, η is typically set as $\eta = 1$, in which case no enhancement will be imposed at all if the estimated blurriness M_p is very large, since $\Lambda_p \to \eta = 1$ when $M_p \to \infty$. It is worthwhile to point out here that it is also possible to set $\eta < 1$ in (9)as practiced in [7] and [16]. This will even suppress some unwanted details (such as noise) presented in the blurred background. This will be demonstrated in Section V.

An example of the scaling factor matrix Λ generated by the above-described mapping process can be found in Fig. 3, from which one can see that relatively small scaling factors have been allocated to those *highly-blurred* regions, as expected. Consequently, such regions will receive less amount of enhancement, or even no enhancement at all. Our extensive simulation results have shown that this effectively avoids the unwanted amplified noise that often appears in those highly-blurred regions.

C. Case 2: Scaling Factor Matrix Computation for Non-Edge-Preserving Filter

If a *non-edge-preserving* filter (such as the Gaussian filter, the average filter, to name a few) is used for conducting layer decomposition, both *blurriness* and *contrast* information will be jointly exploited for the computation of the scaling factor matrix Λ. The rationale behind the use of contrast information as an additional guidance, besides the blurriness information, is justified with detailed explanations as follows.

The overshoot artifacts are typically introduced due to the enhancement imposed on those strong details incurred on the detail layer T [8]. This could happen, especially when a non-edge-preserving type of low-pass filter is deployed in the layer-decomposition stage. This means that more edge information could be incurred on the detailed layer, since the strong edge information were not well preserved on the base layer B.

As non-edge-preserving filters are commonly exploited as preferable due to their algorithmic simplicity and computational efficiency, the above-mentioned overshoot artifacts must be addressed and prevented. Obviously, working on the detail layer T is the most direct and effective way to pursue. For that, an addition multiplier is proposed and inserted into (8) to adaptively prevent overshoot artifacts. That is,

$$\Lambda_p = \rho \cdot \mathcal{G}_b(M_p) \cdot \mathcal{G}_c(|T_p|) + \eta. \tag{10}$$

where

$$\mathcal{G}_c(|T_p|) = e^{-\left(\frac{|T_p|}{\mu}\right)^{\nu}}. \tag{11}$$

That is, $\mathcal{G}_c(|T_p|)$ is another generalized Gaussian function similar to $\mathcal{G}_b(M_p)$, except that it works in the contrast domain rather than in the blurriness domain and $\mathcal{G}_c(|T_p|) \in (0, 1]$. According to (11), the larger the value of $|T_p|$, the smaller the weighting factor $\mathcal{G}_c(|T_p|)$ to be further multiplied in (10). By incorporating this extra term, the scaling factors computed via (10) will be smaller compared with the case, when only (8) is used, especially at those pixels where the detail amplitude $|T_p|$ are large. This effectively helps to avoid or alleviate the overshoot artifacts.

Similar to the α and β in (9), the μ and ν in (11) control the shape of $\mathcal{G}_c(|T_p|)$. In this work, μ and ν are empirically determined as $\mu = 1/2(\rho + \eta)$ and $\nu = 3$, respectively. Note that, according to (8), Λ_p reaches to its maximum value $\rho + \eta$ as the largest scaling factor, when $\mathcal{G}_b(M_p) = 1$. Hence, μ is set to be inversely proportional to $\rho + \eta$ as a general trend, since the larger the $\rho + \eta$ is, the more likely the overshoot artifacts will be incurred. The extra factor $1/2$ in μ is empirically determined for a more robust enhancement.

Note that a similar scaling factor adaptation scheme can be found in [8], which also utilizes the information presented

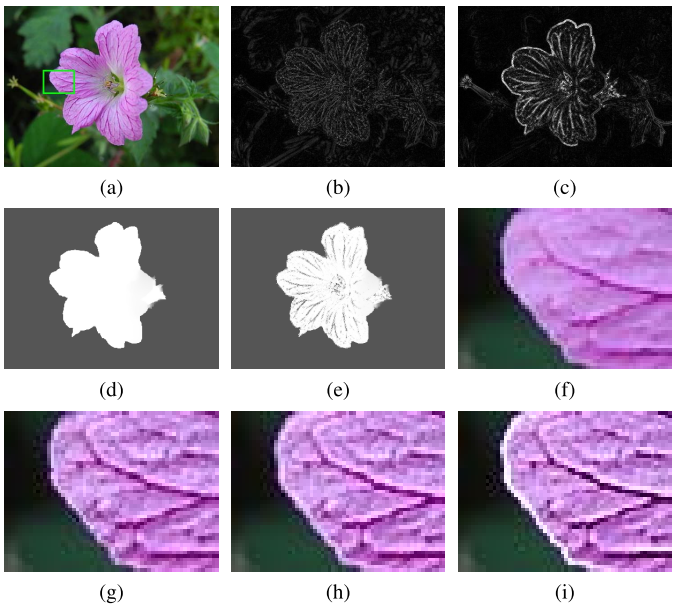


Fig. 5. Demonstrations of adaptive scaling factor matrix Λ computed for Case 1 and Case 2: (a) the input image; (b)-(c) the $|T|$ obtained via the bilateral filter for Case 1 and the Gaussian filter for Case, respectively; (d)-(e) the scaling factor matrix Λ computed for Case 1 and Case 2, respectively; (f) an image patch cropped from (a); (g)-(h) enhanced patches in Case 1 and Case 2, respectively; (i) enhanced patch in Case 2 using the mapping function (8) (i.e., without incorporating $|T|$).

on $|T|$ and exploits the same computation formula as $\Lambda_p = \rho \cdot \mathcal{G}_c(|T_p|) + \eta$, where $\mathcal{G}_c(|T_p|)$ is defined in (11). However, the parameter μ in (11) is fixed and set to 1 in [8], while ours is adaptive as the values of ρ and η are adjustable. Furthermore, the scaling scheme in [8] does not involve the blurriness-guided term $\mathcal{G}_b(M_p)$ as ours in (10). Consequently, the scaling scheme in [8] is less effective than ours on preventing both noise amplification and overshoot artifacts. All these will be demonstrated in Section V.

Fig. 5 demonstrates a set of enhanced images for Case 1 and Case 2, respectively. The representative layer-decomposition filters employed are the edge-preserving bilateral filter for Case 1 and the non-edge-preserving Gaussian filter for Case 2, respectively. Fig. 5 conveys some fundamental insights as follows. First, comparing Figs. 5 (b) and (c), it can be observed that their amplitude map $|T|$ are quite different, as expected. This clearly justifies why the layer-decomposition filter deployed in the first stage in Fig. 3 needs to be considered in the design of the mapping algorithm for the computation of the scaling factor matrix Λ. Second, it can be observed from Figs. 5 (d) and (e) that the scaling factors computed in Case 2 at those pixels with large detail amplitudes have been effectively reduced compared to their counterparts in Case 1, by incorporating the $\mathcal{G}_c(|T_p|)$ into the mapping function (10). Consequently, the enhanced images yielded in both cases, as shown in Fig. 5 (g) and (h) are fairly close to each other, have demonstrated proper enhancement without creating artifacts, compared to the original image patch cropped from the original image as shown in Fig. 5 (f). In contrast, if the mapping function (8) instead of (10) is used for Case 2, distinct overshoot artifacts will be yielded as shown in Fig. 5 (i). All these results have clearly demonstrated the effectiveness and consistence of our filter-type adaptive scaling factor computation; i.e., the proposed BUM algorithm is able to deliver satisfactory enhancement results regardless of which type of layer-decomposition filter is deployed in the first stage of the enhancement process.

V. Experimental Results

Extensive simulations have been conducted on a variety of test images to evaluate the enhancement performance using the proposed blurriness-guided unsharp masking (BUM) approach. As previously mentioned, our proposed BUM algorithm can be viewed as a *generalized* UM framework, since it is much more adaptive than the conventional UM algorithm. The powerful adaptations are incurred not only for determining the scaling factor for *each* pixel but also for adjusting the degree of enhancement according to the type of the layer-decomposition filter that was used in the layer-decomposition stage for generating the base layer and the enhancement layer. For the latter, two types of layer-decomposition filters are considered; i.e., the *edge-preserving* filter (i.e., Case 1, as defined previously) and the *non-edge-preserving* filter (i.e., Case 2). In our simulations, the bilateral filter [10] is chosen as the most representative one forCase 1, while the Gaussian filter is the selected representative for Case 2. The enhancement results of our BUM approach have been compared to those of the conventional UM [1], the *generalized unsharp masking* (GUM) [8][2] and the *cubic unsharp masking* (CUM) [5]. Besides, we noticed a recent work developed by Li *et al.* [29] for contrast enhancement and deblocking, and found its proposed texture mask can be used as an adaptive strength map for the UM process. It turns out that such texture-mask-guided UM (TUM) algorithm is able to deliver quite promising results, therefore, it is also included in our comparison.

For performance evaluation, to our best knowledge, there is no widely-accepted metric for quantitatively measuring image enhancement results, however there are some existing works in this area (e.g., [30]). Therefore, *subjective* or perceptually-based comparisons of the enhanced images using the above-mentioned methods are conducted and compared to justify the superiority of our proposed approach.

A. Parameter Setting

For the blurriness estimation stage, the parameter values as previously set in Section III are used as the default setting; that is, $\lambda = 0.2$ in (2); $\sigma_t = 1.8$ and $N_t = 2.5\% \cdot N$ (where N is the total number of the pixels of the input image) in (3); $\kappa = 10^{-5}$ and $\gamma = 2$ in (4). For computing the scaling factor for each pixel using the generalized Gaussian function, the common parameters for both Case 1 and Case 2 are set as follows: $\rho = 2.5$, $\eta = 1$ or 0 (to be discussed in detail in sub-sections V-B and V-C, respectively)), $\alpha = 1$, and $\beta = 6$.

[2]The original GUM algorithm has an extra contrast enhancement step conducted on the base layer via the histogram equalization (HE) algorithm. However, for a fair comparison, this step has been removed in our simulations.

Fig. 6. Visual comparison of some enhancement results experimented on the test image "Old Man:" (a) the original image and its enhanced versions using the following methods; i.e., (b) the conventional UM with the Gaussian filter, (c) the *modified* GUM [8] with the Gaussian filter, (d) the CUM [5] with the Laplacian filter, (e) the TUM [29] with the Gaussian filter, and (f) the proposed BUM with the Gaussian filter (when $\eta = 1$).

For Case 2, two extra parameters are required and mentioned in Section IV-C, they are $\mu = 1/2(\rho + \eta)$ and $\nu = 3$ in (10).

It is important to note that throughout our simulations all the parameters of our proposed algorithm are fixed to their default values as described above (i.e., no fine-tuning for any specific test image). However, in real applications we believe that some of these parameters could be made available to the viewers for fine-tuning. In fact, this is already practiced in some image enhancement software.

To conduct a fair subjective comparison of the proposed BUM with the conventional UM and other adaptive UM methods, the parameters of these methods that control the enhancement strength (i.e., the scaling effect imposed on the detail layer) have been carefully adjusted so that the overall amount of enhancement yielded by each of these methods is perceptually close to that by our approach. For the remaining parameters in these methods, they have been set to their originally proposed default values, respectively.

The above-highlighted two possible settings (i.e., either 1 or 0) of the parameter η are further investigated in our simulations. In most existing adaptive UM methods, $\eta = 1$ is to ensure that the original details on the input image will be, at least, preserved, if not enhanced. In our proposed BUM, on the other hand, η could be less than 1, allowing the 'enhancement' process to perform opposite—i.e., *smoothing* image details; this could be highly desirable and beneficial on suppressing noise often presented in those highly-blurred background regions. The performance evaluations of our proposed BUM for $\eta = 1$ and $\eta = 0$, respectively, will be discussed in detail, in what follows.

B. Performance Evaluation for $\eta = 1$

Fig. 6 shows a set of enhanced test image "Old Man," produced by the proposed BUM and by other four UM methods, all using the Gaussian filter as the layer-decomposition filter. It can be clearly observed from Fig. 6 (b) that the conventional UM method has amplified noise on the highly-blurred region (referred to the red- framed region cropped from the background), as well as some overshoot artifacts on the foreground region that has clear details (referred to the green-framed region). The GUM [8] and CUM [5] algorithms have delivered slightly improved enhancement results compared with that of the UM, but they tend to yield artifacts for high-contrast regions or cause under-enhancement for low-contrast areas. For example, the GUM algorithm is unable to avoid the unwanted noise amplification often incurred in blurred background regions (see the first close-up in Fig. 6 (c)), while the CUM method can result in under-enhancement for those less-blurred regions with low-contrast details (see the close-up in the second row of Fig. 6 (d)). The TUM [29] algorithm delivers very similar results to ours, but it also leads to slight over-enhancement artifacts (see the close-up in the third row of Fig. 6 (e)). On the other hand, our proposed BUM approach

Fig. 7. Visual comparison of some enhancement results experimented on the test image "Parrot:" (a) the original image and its enhanced versions using the following methods; i.e., (b) the conventional UM with the Gaussian filter, (c) the *modified* GUM with the Gaussian filter [8], (d) the CUM [5] with the Laplacian filter, (e) the TUM [29] with the Gaussian filter, and (f) the proposed BUM with the Gaussian filter (when $\eta = 1$).

is able to constantly deliver the most desirable enhancement results from the viewpoint of perceptual quality as shown in Fig. 6 (f). That is, our proposed BUM algorithm can avoid noise amplification and overshoot artifacts, without yielding under-enhancement for those regions containing low-contrast image details.

Another illustration of the enhancement results yielded by these methods is shown in Fig. 7, which is obtained from the experiment conducted on the test image "Parrot." Similar conclusion about performance comparison as drawn from Fig. 6 can be also drawn from Fig. 7; except that the TUM yields insufficient enhancement at foreground's regions with weak details (see the third close-up in Fig. 7 (e)). Again, one can see that our proposed BUM is the only method that can simultaneously avoid noise amplification on the blurred background region and under-enhancement on the foreground region that has low-contrast details. Meanwhile, it also sensitively avoids the overshoot artifacts that tend to incur at the foreground-background boundary as shown in the third close-ups of Fig. 7 (b) and (c). In closing, the proposed BUM is able to simultaneously address under-enhancement, over-enhancement, and noise-amplification issues in a unified and consistent treatment.

C. Performance Evaluation for $\eta = 0$

As mentioned previously, a common practice in the existing enhancement algorithms is to set the detail scaling factor to 1 (i.e., no enhancement) as an extremal case, if necessary. In our proposed BUM approach, this is equivalent to set $\eta = 1$ in (8) and (10). However, we have also investigated the possibility of using $\eta < 1$ in our approach; i.e., *smoothing* the image details, rather than *enhancing*. In this sub-section, the enhancement results using $\eta = 0$ are demonstrated and discussed as follows.

Note that a highly-blurred region tends to contain much less image details and stand out unwanted noise. By setting $\eta = 0$ for such case, it is beneficial to de-emphasize the presence of noise, while yielding a more visually-pleasant image. Fig. 8 demonstrates the results obtained by using our proposed BUM approach with $\eta = 0$ experimented on the test image "Bear." One can see that our approach has delivered a highly-appealing result; it suppressed the unwanted noise in the highly-blurred background and in the meantime enhanced the details in the foreground. In contrast, all the other methods under comparison failed to accomplish both objectives simultaneously. Through our extensive simulations, it has been observed that using $\eta = 0$ in our proposed BUM works especially well for those images containing with small amount of noise but with distinct foreground and background.

Lastly, it is worth to mention that for the images with a very complicated scene, the computed blurriness maps tend to be less accurate. Consequently, some desired image details might be improperly enhanced due to improper scaling factors computed and assigned. Therefore, using $\eta = 1$ is more recommended in practice, as it tends to deliver more robust enhancement results.

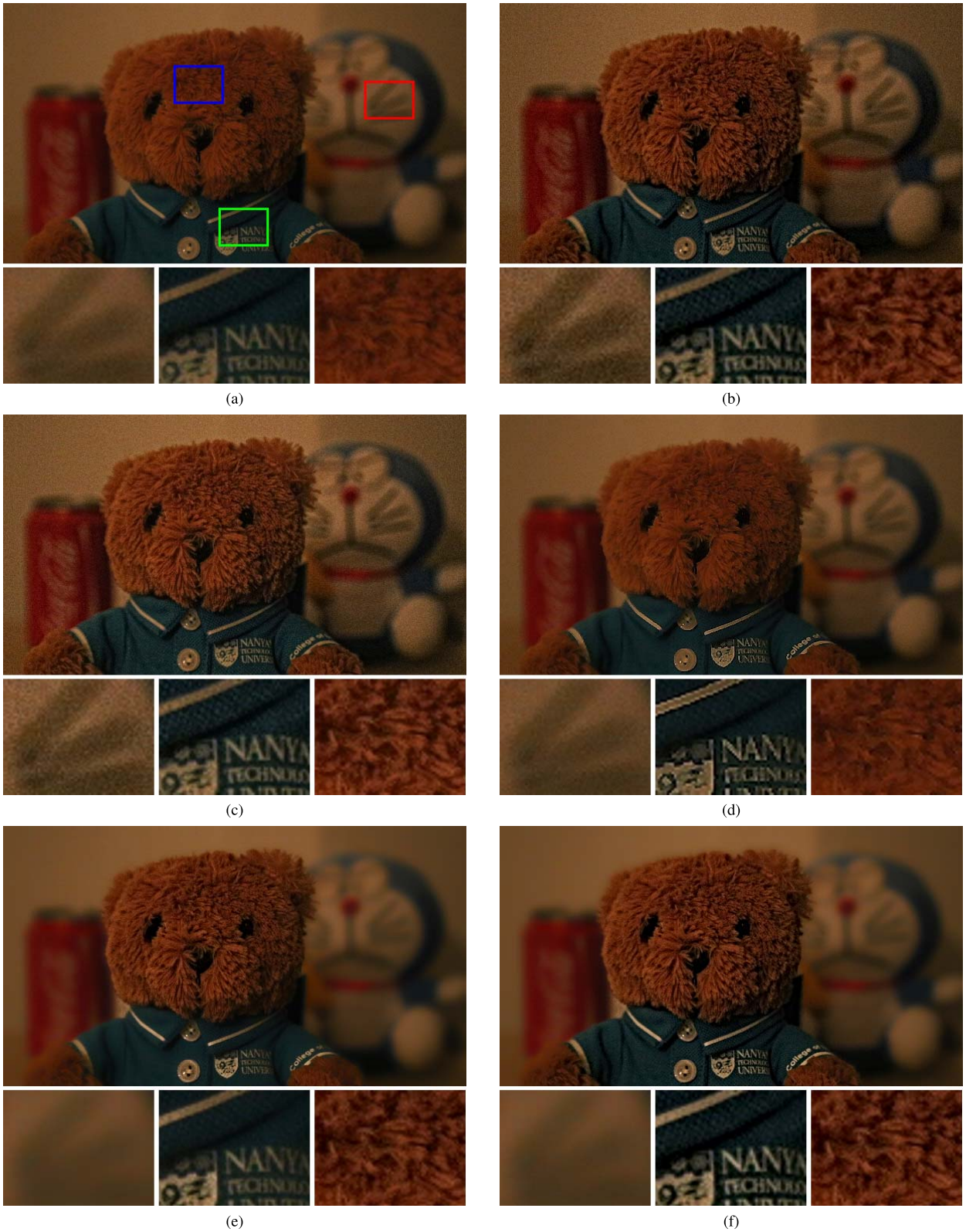


Fig. 8. Visual comparison of the enhancement results on the test image "Bear:" (a) the original image and its enhanced versions using the following methods; i.e., (b) the conventional UM, (c) the *modified* GUM, (d) the CUM, (e) the TUM, and (f) the proposed BUM (when $\eta = 0$). Note that the *bilateral* filter is exploited in each above-mentioned method for the decomposition of the base layer and the detail layer.

D. Subjective Performance Evaluations

To justify the effectiveness of our proposed algorithm, we have invited 9 volunteers who have no experience in image processing to conduct subject performance evaluation. The entire experiments are in blind test (i.e., enhanced images are unlabeled), and they were asked to choose the

TABLE I

AVERAGE RUNNING TIME (IN SECONDS) OF DIFFERENT METHODS FOR ENHANCING AN 800×533 IMAGE

UM	GUM [8]	CUM [5]	TUM [29]	Proposed
0.24	0.27	0.26	14.62	5.27

preferred image among the original and enhanced images from the aesthetic point of view. It turns out that the majority of the 9 participants prefers the enhanced images produced by our proposed BUM image enhancement method. To be exact, 7, 6, and 6 positive votes favoured our proposed BUM algorithm, for the three test images in Figs. 6-8, respectively.

E. Computational Complexity

This sections provides a comparison of the computational complexity of the proposed BUM method and other state-of-the-art UM methods. All these methods are tested using their MATLAB implementations and run on the same machine with an Intel Core i7-6700 3.4 GHz CPU and 8 GB RAM. The Gaussian filter is employed to conduct the layer decomposition in all methods except the CUM method, which employs a polynomial filter instead. The running times (average of 10 runs) of all methods for enhancing an 800×533 image are documented in Table I.

It can be seen that the proposed BUM method is computationally much more expensive than other UM methods under comparison, except the TUM. A closer examination shows that over 99.8% of the running time is spent on the generation of the blurriness map; to be specific, the generation of the initial blurriness map costs about 3.14 seconds, and the refinement process of the blurriness map costs about 2.12 seconds. Hence, in order to make our proposed algorithm more practical for real-time enhancement applications, it is definitely necessary to investigate faster and more efficient blurriness estimation methods.

VI. CONCLUSION

In this paper, a novel *blurriness-guided unsharp masking* (BUM) method is proposed that exploits the local blurriness, computed from the input image, as the guidance information to guide the image enhancement process. The use of the blurriness attribute is motivated by several key observations, including the photographer's original intention on creating an aesthetic effect by making background region blurred. All these suggest that enhancing those highly-blurred and highly sharp image regions are both undesirable, while the former could further lead to noise amplification effect via enhancemant. In our proposed BUM image enhancement algorithm, the enhancement strength is adjusted for *each* pixel according to the degree of local blurriness measured on the local region of individual pixel's location. Consequently, the blurred regions are less enhanced or even not enhanced at all to avoid noise boosting at those regions. Another novelty of our proposed BUM algorithm is that it is also highly adaptive to the type of layer-decomposition filter employed on the generation of the base layer and detail layer; this also helps to alleviate the over-enhancement issues. This strategy has not been found in any existing UM methods, to our best knowledge. Extensive simulations conducted on various test images have demonstrated that the proposed BUM method is able to consistently deliver the most visually-pleasant enhanced image compared with that by using the conventional UM and other state-of-the-art adaptive UM methods.

REFERENCES

[1] A. K. Jain, *Fundamentals of Digital Image Processing*. Upper Saddle River, NJ, USA: Prentice-Hall, 1989.

[2] A. C. Bovik, *Handbook of Image and Video Processing*. Orlando, FL, USA: Academic, 2010.

[3] S. K. Mitra, H. Li, I.-S. Lin, and T.-H. Yu, "A new class of nonlinear filters for image enhancement," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, Apr. 1991, pp. 2525–2528.

[4] G. Ramponi, N. K. Strobel, S. K. Mitra, and T.-H. Yu, "Nonlinear unsharp masking methods for image contrast enhancement," *J. Electron. Imag.*, vol. 5, no. 3, pp. 353–366, 1996.

[5] G. Ramponi, "A cubic unsharp masking technique for contrast enhancement," *Signal Process.*, vol. 67, no. 2, pp. 211–222, 1998.

[6] A. Polesel, G. Ramponi, and V. J. Mathews, "Image enhancement via adaptive unsharp masking," *IEEE Trans. Image Process.*, vol. 9, no. 3, pp. 505–510, Mar. 2000.

[7] S. H. Kim and J. P. Allebach, "Optimal unsharp mask for image sharpening and noise removal," *J. Electron. Imag.*, vol. 14, no. 2, pp. 023005-1–023005-13, 2005.

[8] G. Deng, "A generalized unsharp masking algorithm," *IEEE Trans. Image Process.*, vol. 20, no. 5, pp. 1249–1261, May 2011.

[9] 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.

[10] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proc. Int. Conf. Comput. Vis.*, Jan. 1998, pp. 839–846.

[11] Z. Farbman, D. Lischinski, and R. Szeliski, "Edge-preserving decompositions for multi-scale tone and detail manipulation," *Trans. Graph.*, vol. 27, no. 3, p. 67, Aug. 2008.

[12] K. He, J. Sun, and X. Tang, "Guided image filtering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 6, pp. 1397–1409, Jun. 2013.

[13] L. Xu, C. Lu, Y. Xu, and J. Jia, "Image smoothing via $L0$ gradient minimization," *ACM Trans. Graph.*, vol. 30, no. 6, 2011, Art. no. 174.

[14] E. S. L. Gastal and M. M. Oliveira, "Domain transform for edge-aware image and video processing," *ACM Trans. Graph.*, vol. 30, no. 4, p. 69, 2011.

[15] D. Min, S. Choi, J. Lu, B. Ham, K. Sohn, and M. N. Do, "Fast global image smoothing based on weighted least squares," *IEEE Trans. Image Process.*, vol. 23, no. 12, pp. 5638–5653, Dec. 2014.

[16] B. Zhang and J. P. Allebach, "Adaptive bilateral filter for sharpness enhancement and noise removal," *IEEE Trans. Image Process.*, vol. 17, no. 5, pp. 664–678, May 2008.

[17] J. H. Elder and S. W. Zucker, "Local scale control for edge detection and blur estimation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 7, pp. 699–716, Jul. 1998.

[18] Y.-W. Tai and M. S. Brown, "Single image defocus map estimation using local contrast prior," in *Proc. IEEE Int. Conf. Image Process.*, Nov. 2009, pp. 1797–1800.

[19] S. Zhuo and T. Sim, "Defocus map estimation from a single image," *Pattern Recognit.*, vol. 44, no. 9, pp. 1852–1858, 2011.

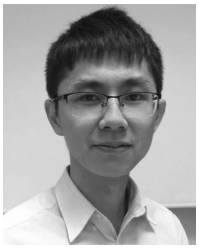
[20] X. Zhu, S. Cohen, S. Schiller, and P. Milanfar, "Estimating spatially varying defocus blur from a single image," *IEEE Trans. Image Process.*, vol. 22, no. 12, pp. 4879–4891, Dec. 2013.

[21] J. Shi, L. Xu, and J. Jia, "Just noticeable defocus blur detection and estimation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recogit.*, Jun. 2015, pp. 657–665.

[22] J. Shi, L. Xu, and J. Jia, "Discriminative blur detection features," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2014, pp. 2965–2972.

[23] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.

[24] D. Lischinski, Z. Farbman, M. Uyttendaele, and R. Szeliski, "Interactive local adjustment of tonal values," *ACM Trans. Graph.*, vol. 25, no. 3, pp. 646–653, 2006.
[25] D. Ferstl, C. Reinbacher, R. Ranftl, M. Ruether, and H. Bischof, "Image guided depth upsampling using anisotropic total generalized variation," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 993–1000.
[26] B. Ham, M. Cho, and J. Ponce, "Robust image filtering using joint static and dynamic guidance," in *Proc. IEEE Conf. Comput. Vis. Pattern Recogn.*, Jun. 2015, pp. 4823–4831.
[27] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 5, pp. 603–619, May 2002.
[28] Y. Saad, *Iterative Methods for Sparse Linear Systems*. Philadelphia, PA, USA: SIAM, 2003.
[29] Y. Li, F. Guo, R. T. Tan, and M. S. Brown, "A contrast enhancement framework with JPEG artifacts suppression," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2014, pp. 174–188.
[30] L. Krasula, P. Le Callet, K. Fliegel, and M. Klíma, "Quality assessment of sharpened images: Challenges, methodology, and objective metrics," *IEEE Trans. Image Process.*, vol. 26, no. 3, pp. 1496–1508, Mar. 2017.



Wei Ye received the B.E. degree from the Huazhong University of Science and Technology, China, in 2012, and the Ph.D. degree from Nanyang Technological University, Singapore, in 2017. His research is mainly focused on fundamental image processing techniques, including image demosaicing, image interpolation, and image enhancement.

Kai-Kuang Ma (S'80–M'84–SM'95–F'13) received the B.E. degree from Chung Yuan Christian University, Chungli, Taiwan, the M.S. degree from Duke University, Durham, NC, USA, and the Ph.D. degree from North Carolina State University, Raleigh, NC, USA, all in electrical engineering.

From 1992 to 1995, he was a Member of Technical Staff at the Institute of Microelectronics, Singapore, involved in digital video coding and the MPEG standards. From 1984 to 1992, he was with in IBM Corporation, Kingston, NY, USA, and Research Triangle Park, NC, USA, engaging on various DSP and VLSI advanced product development. He is currently a Full Professor with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. He has published extensively and holds one U.S. patent on fast motion estimation algorithm. His research interests are in the areas of digital image/video processing and computer vision, including digital image/video coding and standards, image/video segmentation, denoising and enhancement, and interpolation and super-resolution. His research interests on computer vision include image matching and registration, scene analysis and recognition, and human-computer interaction.

Dr. Ma was serving as Singapore MPEG Chairman and the Head of Delegation from 1997 to 2001. On the MPEG contributions, two fast motion estimation algorithms (Diamond Search and MVFAST) produced from his research group have been adopted by the MPEG-4 standard, as the reference core technology for fast motion estimation. He was the General Chair of organizing a series of international standard meetings (MPEG and JPEG), JPEG2000 and MPEG-7 workshops held in Singapore in 2001.

He was elected as a Distinguished Lecturer of the IEEE Circuits and Systems Society from 2008 to 2009. He is a General Co-Chair of ISPACS2017, ASIPA2017, ACCV2016 Workshop, and VCIP-2013; a Technical Program Co-Chair of ICIP-2004, ISPACS-2007, IIH-MSP-2009, and PSIVT-2010; and an Area Chair of ACCV-2009 and ACCV-2010. He has been serving as an Editorial Board Member for several leading international journals in his research area, such as Senior Area Editor for the IEEE TRANSACTIONS ON IMAGE PROCESSING from 2016 to 2019, an Associate Editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY since 2015, the IEEE SIGNAL PROCESSING LETTERS from 2014 to 2016, the IEEE TRANSACTIONS ON IMAGE PROCESSING from 2007 to 2010, the IEEE TRANSACTIONS ON COMMUNICATIONS from 1997 to 2012 as Editor, the IEEE TRANSACTIONS ON MULTIMEDIA from 2002 to 2009, the *International Journal of Image and Graphics* from 2003 to 2015 and the *Journal of Visual Communication and Image Representation* from 2005 to 2015. He is an elected member of three IEEE Technical Committees: Image and Multidimensional Signal Processing Committee, Multimedia Communications Committee, and Digital Signal Processing. He has been serving as a Technical Program Committee member, a reviewer and the Session Chair of multiple IEEE international conferences. He is the Chairman of IEEE Signal Processing Singapore Chapter from 2000 to 2002. He is a member of Sigma Xi and Eta Kappa Nu.