Color to Gray: Visual Cue Preservation

Mingli Song, *Member*, *IEEE*, Dacheng Tao, *Member*, *IEEE*, Chun Chen,
Xuelong Li, *Senior Member*, *IEEE*, and Chang Wen Chen, *Fellow*, *IEEE*

Abstract—Both commercial and scientific applications often need to transform color images into gray-scale images, e.g., to reduce the publication cost in printing color images or to help color blind people see visual cues of color images. However, conventional color to gray algorithms are not ready for practical applications because they encounter the following problems: 1) Visual cues are not well defined so it is unclear how to preserve important cues in the transformed gray-scale images; 2) some algorithms have extremely high time cost for computation; and 3) some require human-computer interactions to have a reasonable transformation. To solve or at least reduce these problems, we propose a new algorithm based on a probabilistic graphical model with the assumption that the image is defined over a Markov random field. Thus, color to gray procedure can be regarded as a labeling process to preserve the newly well-defined visual cues of a color image in the transformed gray-scale image. Visual cues are measurements that can be extracted from a color image by a perceiver. They indicate the state of some properties of the image that the perceiver is interested in perceiving. Different people may perceive different cues from the same color image and three cues are defined in this paper, namely, color spatial consistency, image structure information, and color channel perception priority. We cast color to gray as a visual cue preservation procedure based on a probabilistic graphical model and optimize the model based on an integral minimization problem. We apply the new algorithm to both natural color images and artificial pictures, and demonstrate that the proposed approach outperforms representative conventional algorithms in terms of effectiveness and efficiency. In addition, it requires no human-computer interactions.

Index Terms—Color to gray, probabilistic graphical model, visual cue.

1 Introduction

Color to gray algorithms [1], [2], [3] are used to transform color images into gray-scale images while preserving important visual cues, which are color spatial consistency, image structure information, and color channel perception priority. We will define them strictly in this paper. Although some researchers [2] mentioned the visual cue preservation (VCP) for color to gray, there were no formal definitions for visual cues. Conventional algorithms have been widely used, for example, in publishing as a less expensive alternative to providing full color images and for helping color blind people perceive visual cues better in color images. Recently, many color to gray algorithms [1], [2], [3], [4], [5], [6], [7], [29], [31], [32] have been developed, based on a wide variety of techniques.

Color to gray through linear combination of R, G, and B channels is a kind of time-efficient approach. Wyszecki and Stiles [4] have combined the R, G, and B channels using a group of linear mapping functions. Wu and Rao [5] have linearly combined the R, G, and B channels by defining a series of policies to separate luminance value from the chrominance value so as to construct the gray-scale image based on the luminance value. Neither method, however, was able to preserve the important visual cues represented by different colors. For example, as shown in the transformed gray-scale image in Fig. 1, the gray-scale rendering of the full color image by linearly combining R, G, and B according to [4] fails to represent the contrast between the fruit and leaves that can be perceived in the color image.

We can also treat color to gray as a dimensionality reduction process which degrades a three-dimensional color space to the one-dimensional gray-scale space. Therefore, some unsupervised dimensionality reduction algorithms, e.g., principal component analysis (PCA) [8] and its kernel generalization (w.r.t., kernel PCA, KPCA) [9], can be utilized to carry out the color to gray transformation. However, it is worth emphasizing that PCA for color to gray is equivalent to linearly combining the R, G, and B channels because, in PCA methods, the gray-scale image is obtained by projecting the R, G, and B values of each pixel to the leading principal color component from PCA. Furthermore, PCA [8] assumes measurements are drawn from a single Gaussian and so cannot handle the non-Gaussian distribution of pixels of a color image. KPCA is not seriously affected by this problem, but it is nonetheless very slow [10] for real-time processing and their transformations are sensitive to the kernel function. Moreover, automatic kernel selection is still an open problem.

Bala and Braun [6] have sorted all of the colors in an image according to their original lightness values and then arranged colors to gray scales by using a weight which is proportional to their color distance. This approach performs

- M. Song and C. Chen are with the College of Computer Science, Zhejiang University, Hangzhou 310027, China. E-mail: {brooksong, chenc}@zju.edu.cn.
- D. Tao is with the School of Computer Engineering, Nanyang Technological University, 639798, Singapore. E-mail: dctao@ntu.edu.sg.
- X. Li is with the State Key Laboratory of Transient Optics and Photonics, Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi'an 710119, China. E-mail: xuelong_li@opt.ac.cn.
- C.W. Chen is with the Department of Computer Science and Engineering, University at Buffalo, The State University of New York, Buffalo, NY 14260-2000. E-mail: chencw@cse.buffalo.edu.

Manuscript received 13 Aug. 2008; revised 18 Dec. 2008; accepted 17 Mar. 2009; published online 27 Mar. 2009.
Recommended for acceptance by Q. Ji, J. Luo, D. Metaxas, A. Torralba, T.S. Huang, and E.B. Sudderth.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMISI-2008-08-0496.
Digital Object Identifier no. 10.1109/TPAMI.2009.74.

0162-8828/10/$26.00 © 2010 IEEE Published by the IEEE Computer Society

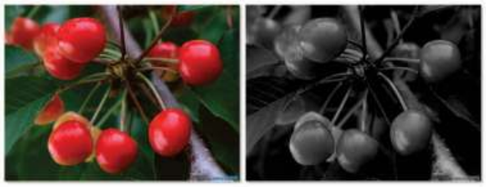


Fig. 1. The color appearance variation is lost by linearly combining R, G, and B channels [4] of a color image. Leaves and fruit have an almost identical gray level in the transformed gray-scale image, although they are visually distinct in the original color image.

well for color images with simple structures, e.g., graphics generated by Excel, which contain no more than 10 distinct colors. However, the algorithm is problematic for color images with high frequency, e.g., a rasterized graphic, without taking into account the visual cue from color spatial distribution in an image.

Bala and Eschbach [31] and Alsam and Kolås [32] have carried out the color to gray by applying the color difference preserved in the original color image to the initial gray-scale image, which is a linear combination of R, G, and B channels. Bala and Eschbach [31] have applied the high-frequency chrominance information to the luminance channel to preserve the difference between adjacent colors locally in the transferred gray-scale image. Alsam and Kolås [32] have enhanced the initial gray-scale image by utilizing a correction mask, which is the sum of the difference between each of the color channels. However, both schemes cannot distinguish between the contributions of hue and chroma, i.e., they deem hue and chroma have an equivalent importantance for color to gray transfer.

Socolinsky and Wolff [11] have applied image gradient information to model the contrast of a color image locally and used it as the visual cue to implement color to gray. However, because the contrast is built based on the maximal gradient from different channels at a short scale, i.e., every pixel and its nearest neighbors, this algorithm cannot deal well with long-scale contrast regions. Especially challenging for this algorithm are the pseudo-isochromatic plates for testing color blindness. In other words, it fails to transfer essential cues from the color image to the gray-scale image without counting visual cues from the spatial distribution of color and different effects of color channels, although we perceive different color channels in different ways. For example, it is possible to map red and green to an identical gray value by using this algorithm.

Rasche et al. [1] have defined an objective function that enforces the proportional color differences across the mapping between color to gray scale based on a subset of pixels in a color image. By minimizing the objective function, they obtained a linear mapping for the transformation. This method can also be used to map original R, G, and B channels into other color spaces. However, it ignores visual cues from the spatial arrangement and the different perceiving effects of colors channels. Therefore, it fails to deal well with images containing very small splashes of colors. To solve this problem, they [7] have introduced multidimensional scaling (MDS) [12] in the CIELAB color space to model the color to gray process. Unfortunately, the MDS-based algorithm is very slow when transforming images containing abundant colors, e.g., natural scenes.

Human-computer interaction (HCI) has been demonstrated to be helpful to further improve the performance of color to gray algorithms by allowing parameters to be tuned for a given image to obtain a visually reasonable transformation. Gooch et al. [2] have presented an interactive Color2Gray (ICG) algorithm that tuned three newly defined parameters: the difference in color chrominance, the contrast between colors, and the size of a pixel's neighborhood. Users use this algorithm for Color2Gray and preserve their preferred visual cues via tuning the above parameters manually. Therefore, it is difficult to find the optimal parameters for a given image to achieve the visual cues preservation. In addition, it is unsuitable for high-resolution images because of its high time cost. Motivated by the color image perceptual framework [30], Smith et al. [29] presented a simple and fast color to gray scheme by combining the global Helmholtz-Kohlrausch color appearance mapping and the multiscale local contrast enhancement. The second step requires HCIs, which makes the scheme well suited for natural images, photographs, artistic reproductions, and business graphics.

Besides the methods mentioned above, a number of color transfer algorithms [13], [14], [15] which calculate color mappings based on color correspondences between a target color image and a source color image can also be applied for color to gray. Because it is difficult to construct correspondences for color images with complex structures, it is unsuitable for the practical color to gray application. In addition, some of them may require heavy manual work.

In summary, a good color to gray algorithm should be able to transform a color image into a gray-scale image efficiently while preserving visual cues. In this paper, we define three visual cues via discovering different ways of color image understanding from painters, vision researchers, and psychologists. Based on these well-defined visual cues, we then develop a new VCP algorithm for color to gray based on a probabilistic graphical model. There are two main contributions:

- *Visual cues*: Different people may have different viewpoints to understand an identical color image, e.g., painters revealed the color spatial consistency of almost identical colors in a color image, computer vision researchers suggested employing image structure information for image understanding, and psychologists found the human perception priority of hue, chroma, and lightness. We define three visual cues quantitatively to reflect these different understandings of color images. These three visual cues are, namely, color spatial consistency, image structure information, and color channel perception priority.
- *Probabilistic graphical model for color to gray transformation*: To transfer the captured visual cues in the color image to a gray-scale image as much as possible, we construct a probabilistic graphical model based on the well mathematically defined visual cues by treating the target gray-scale image as a Markov random field. By using the Bayes' theorem, this probabilistic

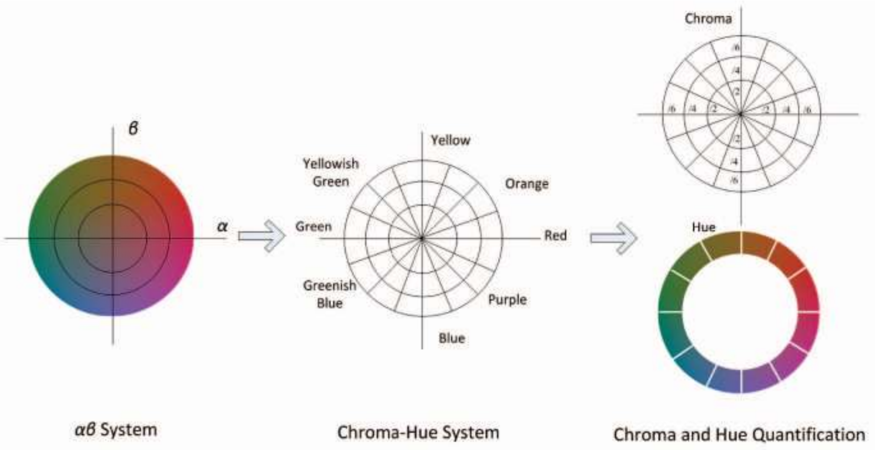


Fig. 2. The hue-chroma system for color representation

graphical model is further transformed to a Maximum a Posteriori (MAP) problem, which is identical to an integral minimization procedure. Furthermore, according to the Euler-Lagrange equation, the integral minimization procedure can be rewritten as a linear equation system.

Advantages of the proposed algorithm are the following: 1) Visual cues are defined based on different understandings of color images to reflect human visual cognition; 2) the probabilistic graphical model transfers the most important visual cues from a color image to the corresponding gray-scale image; and 3) the proposed method is automatic with limited time complexity. In comparison with competitive approaches, i.e., PhotoShop [3], Interactive Color2Gray [2], and Multidimensional scaling color to gray [1], the proposed one usually outperforms other algorithms based on preference matrix driven by paired comparison and can process a color image with much less computational cost without HCIs.

The rest of the paper is organized as follows: In Section 2, we first introduce a modified CIELCH color space and then define three visual cues based on different ways of color image understanding. Finally, we cast the color to gray process as a probabilistic graphical model associated with a fast optimization procedure. Experimental results in Section 3 thoroughly demonstrate the effectiveness and the efficiency of the new algorithm, and Section 4 concludes.

2 Visual Cue Preservation for Color to Gray

A visual cue is a statistic or signal that can be extracted from a color image by a perceiver that indicates the states of some properties of the image that the perceiver is interested in sensing. It brings to mind knowledge from previous experiences by providing a framework for its own interpretation.

In this paper, we define three visual cues, which are the color spatial consistency, the image structure information, and the color channel perception priority, by briefing three typical understandings of what people cognize color images.

We hold that color to gray preserves the aforementioned three visual cues through a process of inference in which cues are used to make probabilistic best guesses about what should be presented in the transformed gray-scale image. Here, we apply a probabilistic graphical model to implement color to gray and manifest visual cues in the transformed gray-scale image.

Before we define the proposed three visual cues, we first introduce a modified CIELCH color space. This is because definitions of these cues are based on this color space.

2.1 A Modified CIELCH Color Space

A suitable color space is the foundation to carry out a successful color image parsing. In this paper, it is essential to choose a suitable color space to define visual cues. According to color vision theory [4], to model the human color perception, RGB is not as suitable as CIELCH or CIELAB color spaces. In addition, because CIELAB color space does not define hue and chroma explicitly, but CIELCH does, we employ CIELCH. Because color spectrums are traditionally parsed (or analyzed) and formatted in RGB (red, green, and blue) channels, we need to transform pixels to CIELCH.

To transform pixels from RGB color space to CIELCH, we first transform pixels from RGB space to $l\alpha\beta$ space [16], which was obtained by a statistical learning process based on a large number of natural color images. The highly independent parameters l, α, and β, respectively, represent the lightness, the red-green, and the blue-yellow components. Fig. 2 shows that α and β span a disk which covers red to yellow, yellow to green, green to blue, and blue to red. The figure also shows that the hue can be quantified as a ring while the chroma can be viewed as the radius of the ring. Consequently, we can use α and β to define the hue and chroma in a modified CIELCH,

$$C^*_{\alpha\beta} = [\alpha^2 + \beta^2]^{1/2} \quad \text{and} \tag{1}$$

$$H^*_{\alpha\beta} = \begin{cases} \arccos\left[\frac{\alpha^\gamma}{(\alpha^2+\beta^2)^{\gamma/2}}\right], & C^*_{\alpha\beta} \neq 0, \beta \geq 0, \\ 2\pi - \arccos\left[\frac{\alpha^\gamma}{(\alpha^2+\beta^2)^{\gamma/2}}\right], & C^*_{\alpha\beta} \neq 0, \beta < 0, \end{cases} \tag{2}$$

where γ is a factor to control the distribution of the hue value. In [17], $\arctan$ is employed to measure the hue difference. The range of $\arctan$ is $(-\pi/2, \pi/2)$ and $\arctan$

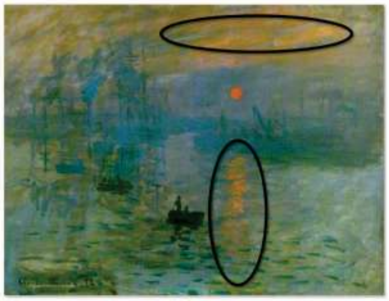


Fig. 3. The painting “Impression, Sunrise” by Monet.

fails to distinguish the difference between the left semicircle and the right semicircle of the hue representation; thus, a piecewise operator is applied in (2) to replace the original acttan operator in [17]. The range of the piecewise $\arccos$ is $[0, 2\pi)$, so it is convenient to distinct the difference between the left semicircle and the right semicircle of the hue representation. Note that both the hue and chroma will be undistinguishable when $C^*_{\alpha\beta} = 0$. Once all pixels are transformed to the modified CIELCH, we can define three visual cues, respectively.

2.2 Visual Cue 1: Color Spatial Consistency

> *When you go out and paint, try to forget what objects you have before you, a tree, a house, a field, or whatever, merely think, here is a little square of blue, here an oblong of pink, here is a streak of yellow, and paint it just as it looks to you, the exact color and shape, until it gives you own naïve impression of scene before you.*
>
> —Claude Oscar Monet (1840-1926), Impressionist Painter, French.

The first visual cue is termed the color spatial consistency, initially revealed by painters, which means that an object in a color scene has almost identical colors. Claude Oscar Monet, the 19th century French Impressionist, grouped almost-identical colors together in patches, as indicated by two elliptical regions in Fig. 3 to demonstrate the sky and the reflection stripe of the sun on the sea in spite of some subtle “burrs” (small strokes of color brushing) in these regions. This is why painters can depict a scene concisely with only a few brushes. Therefore, to capture the correct structural information instead of the subtle “burr” of an image, the color spatial consistency should be taken into account.

The color spatial consistency measures the smoothness of a window. The smoothness is reflected by the difference between the color (i.e., hue and chroma) value expectation and the Gaussian blurred result within the window. The larger the difference is, the less the smoothness will be.

In this paper, the color spatial consistency is defined on hue and chroma but not on lightness because lightness is independent of color. Therefore, we have two color spatial consistencies, $U_H(x,y)$ and $U_C(x,y)$, defined on hue and chroma, respectively. Both $U_H(x,y)$ and $U_C(x,y)$ are defined in (5) and (6), respectively. Fig. 4 shows the procedure for computing $U_H(x,y)$ and $U_C(x,y)$ of a sampling window. To define $U_H(x,y)$ and $U_C(x,y)$, we need to calculate the hue value expectation $E_{M_H}(x,y)$ and the chroma value expectation $E_{M_C}(x,y)$.

For a square window $W_{M\times M}$ with a side length M and centered at (x,y), its hue value expectation $E_{M_H}(x,y)$ is:

$$E_{M_H}(x,y) = \frac{1}{|D_M|} \sum_{i=x-M/2}^{x+M/2} \sum_{j=y-M/2}^{y+M/2} R \begin{bmatrix} \cos \mathbf{I}_H(i,j) \\ \sin \mathbf{I}_H(i,j) \end{bmatrix}, \tag{3}$$

and its chroma value expectation $E_{M_C}(x,y)$ is:

$$E_{M_C}(x,y) = \frac{1}{|D_M|} \sum_{i=x-M/2}^{x+M/2} \sum_{j=y-M/2}^{y+M/2} \mathbf{I}_C(i,j), \tag{4}$$

where $\mathbf{I}_H(i,j)$ and $\mathbf{I}_C(i,j)$ are, respectively, the hue and chroma values at (i,j) obtained via (1) and (2), $|D_M| = M \times M$, and the constant scale R is insensitive to various color images (usually, we set $R = 100$).

Then, for each pixel, $U_H(x,y)(U_C(x,y))$ is obtained by computing the distance between the hue (chroma) value of the Gaussian $G(\cdot)$ blurred pixel (x,y) and the corresponding color (hue or chroma) value expectation $E_{M_H}(x,y)(E_{M_C}(x,y))$, i.e.,

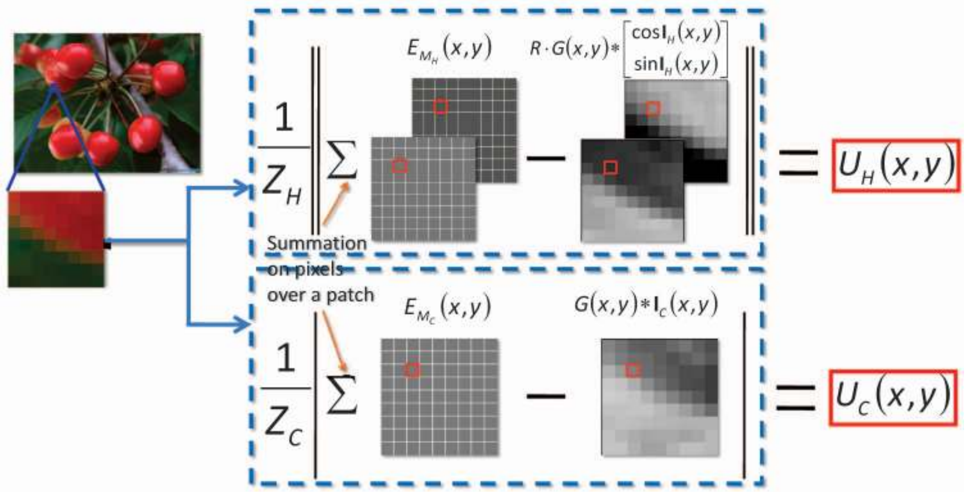

$E_{M_H}(x,y)$
$R\cdot G(x,y)*\begin{bmatrix}\cos\mathbf{I}_H(x,y)\\ \sin\mathbf{I}_H(x,y)\end{bmatrix}$
$\frac{1}{Z_H}$
$\sum$
$=$
$U_H(x,y)$
Summation on pixels over a patch
$E_{M_C}(x,y)$
$G(x,y)*\mathbf{I}_C(x,y)$
$\frac{1}{Z_C}$
$\sum$
$=$
$U_C(x,y)$

Fig. 4. The procedure for computing two color spatial consistencies $U_H(x,y)$ and $U_C(x,y)$.

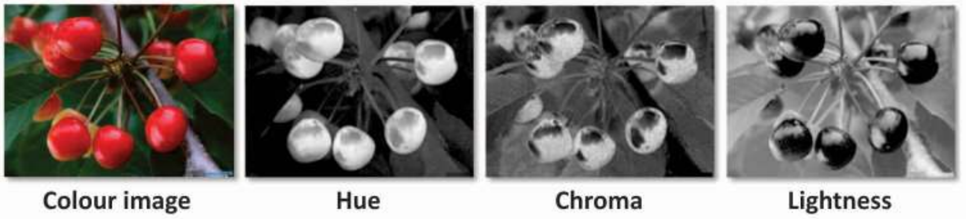


Fig. 5. Distributions of the hue, chroma, and lightness channels in a color image.

$$U_H(x,y) = \frac{1}{Z_H}\left\| E_{M_H}(x,y) - R\cdot G(x,y) * \begin{bmatrix}\cos \mathbf{I}_H(x,y)\\ \sin \mathbf{I}_H(x,y)\end{bmatrix}\right\| \quad \text{and} \tag{5}$$

$$U_C(x,y) = \frac{1}{Z_C}|E_{M_C}(x,y) - G(x,y) * \mathbf{I}_C(x,y)|, \tag{6}$$

where Z_H and Z_C are normalization factors in the sampling window $W_{M\times M}$ for hue and chroma, respectively. After calculating all $U_H(x,y)$ and $U_C(x,y)$ in an image based on (5) and (6), we need to rescale them to $[0,1]$ based on min-max normalization.

In hue and chroma, the color spatial consistencies $U_H(x,y)$ and $U_C(x,y)$ for pixels around an edge are larger than those for pixels in a flat region. As shown in Fig. 4, pixels in fruit have smaller color spatial consistencies, while pixels around the boundary between fruits and leaves have larger color spatial consistencies.

In summary, the computation of the color spatial consistency consists of three steps, i.e., computing color (hue or chroma) value expectation, Gaussian blurring the sampled window, and computing the difference between color (hue or chroma) value expectation and Gaussian blurring result.

2.3 Visual Cue 2: Image Structure Information

Computer vision researchers have suggested that the understanding of a color image requires an understanding of the structure of the image [19], [20], [21], [22]. The image structure information is defined by how hue, chroma, and lightness vary across an image. As illustrated by Fig. 5, dramatic variations in channels of hue, chroma, and lightness show three different presentations of the same image. Usually, the image structural information can be obtained by integrating gradients over three channels, i.e., $\nabla\mathbf{I}_H(x,y)$, $\nabla\mathbf{I}_C(x,y)$, and $\nabla\mathbf{I}_L(x,y)$.

For the hue channel, we have two problems to calculate its gradient.

First, it is difficult to distinguish different hues when corresponding chroma values are small, though $\nabla\mathbf{I}_H(x,y)$ is independent of chroma. In other words, as shown by the $\alpha\beta$ system in Fig. 2, human eyes are insensitive to hue variations when the chroma is near the center of the circle. It is necessary to take this phenomenon into account so that a weighting factor $[\mathbf{I}_C(x,y)/C_0]^{\lambda_H}$ is imposed to adjust $\nabla\mathbf{I}_H(x,y)$, where $\mathbf{I}_C(x,y)$ is the chroma value at (x,y), C_0 is a constant to define the dynamic range of chroma, and $\lambda_H = 2$ is a constant coefficient for scale adjustment. With this weighting factor, we can extract the gradients in hue like human eyes do.

Second, it is impossible to distinguish the direction of $\nabla\mathbf{I}_H(x,y)$ because hue is represented by a cyclic function, as shown in Fig. 2. To tackle the second problem, we cut the hue ring (as shown at the bottom right of Fig. 2) and transform it into a line segment. To implement this procedure, we first find a starting point, an angle θ, on the ring, and then quantize the ring into 256 bins. To avoid a bad cut that makes two close points on the hue circle to be the two terminals, θ is chosen to be the hue value that is farthest from all possible hue values in the original color image, i.e.,

$$\begin{aligned}\theta &= \arg\max_{\theta} \sum_{(x,y)\in\mathbf{I}} \left\| \begin{bmatrix}\cos \mathbf{I}_H(x,y) - \cos\theta\\ \sin \mathbf{I}_H(x,y) - \sin\theta\end{bmatrix}\right\|,\\ \theta &\in \left\{0, 1\times\frac{2\pi}{256}, \ldots, 255\times\frac{2\pi}{256}\right\}.\end{aligned} \tag{7}$$

Finally, the hue value for pixel (x,y) is redefined on the line segment as below:

$$\mathbf{I}_{\tilde{H}}(x,y) = \begin{cases}\mathbf{I}_H(x,y) - \theta, & \mathbf{I}_H(x,y) - \theta \geq 0,\\ \mathbf{I}_H(x,y) - \theta + 2\pi, & \mathbf{I}_H(x,y) - \theta < 0.\end{cases} \tag{8}$$

In summary, by combining the weighting factor $[\mathbf{I}_C(x,y)/C_0]^{\lambda_H}$, the original gradient $\nabla\mathbf{I}_{\tilde{H}}(x,y)$, and the corresponding direction $\mathrm{sign}(\nabla\mathbf{I}_{\tilde{H}}(x,y))$, we update the gradient on hue as:

$$\nabla\mathbf{I}_H(x,y) \leftarrow \mathrm{sign}(\nabla\mathbf{I}_{\tilde{H}}(x,y))\left[\frac{\mathbf{I}_C(x,y)}{C_0}\right]^{\lambda_H}|\nabla\mathbf{I}_{\tilde{H}}(x,y)|, \tag{9}$$

where $\lambda_H = 2$ and $\mathrm{sign}(x)$ is $+1$ if $x \geq 0$; otherwise, -1.

2.4 Visual Cue 3: Color Channel Perception Priority

As reported in [17, p. 95], psychologists have found a human's perception priority of hue, chroma, and lightness based on extensive experiments in psychology on visual color and tolerance of human. That is, for a normal observer, the cue of a color image is first provided by variations in hue, then in chroma, and least of all in lightness [17]. That is why we impose ordering and sequential dependencies among hue, chroma, and lightness. The algorithm pays attention to hue variations first. If hue is constant, attention is paid to chroma variations. If hue and chroma are constant, attention is then paid to lightness variations.

In this paper, the perception priority of hue, chroma, and lightness is quantized by a series of priority factors, which will be utilized to weight $\nabla\mathbf{I}_H(x,y)$ defined in (9), $\nabla\mathbf{I}_C(x,y)$ and $\nabla\mathbf{I}_L(x,y)$ for integration. In this paper, the hue priority factor $P_H(x,y) = 1$ and priority factors for

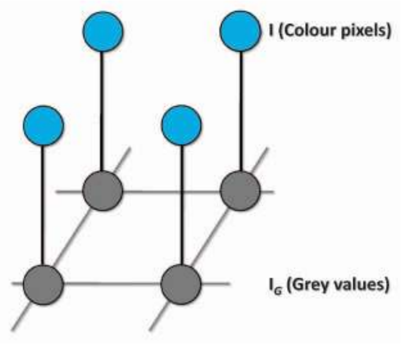


Fig. 6. An undirected graphical model representing a Markov random field for color to gray in which $\mathbf{I}_G(x,y)$ is a gray value denoting the state of a pixel at (x,y) in the unknown gray-scale image and $\mathbf{I}(x,y)$ denotes the corresponding color pixel ($\mathbf{I}_H(x,y)$, $\mathbf{I}_C(x,y)$, and $\mathbf{I}_L(x,y)$) at (x,y) in the observed color image.

chroma and lightness are defined based on color spatial consistencies $U_H(x,y)$ and $U_C(x,y)$. According to the perception priorities of hue, chroma, and lightness, the visual information perception in chroma channel is affected by that in hue channel. Similarly, the visual information perception in lightness channel is affected by that in both hue and chroma channels.

By adopting the influence of the color spatial consistency of hue $U_H(x,y)$, we model the chroma priority factor as:

$$P_C(x,y) = \exp(-\lambda_C U_H(x,y)), \tag{10}$$

where λ_C is the smallest distinguishable variation of hue. Given $0 \leq U_H \leq 1$, to separate chroma into 256 scales, we need to set λ_C as $-\ln(1/256)$ or approximately 5.

By imposing the influence of the color spatial consistency of hue and chroma, the lightness priority factor is given by:

$$P_L(x,y) = \exp(-\lambda_C U_H(x,y)) \exp(-\lambda_L U_C(x,y)), \tag{11}$$

where λ_L is the smallest distinguishable variation of lightness. Given $0 \leq U_C \leq 1$ and $0 \leq \exp(-\lambda_C U_H(x,y)) \leq 1$, to separate lightness into 256 scales, we need to set λ_L as $-\ln(1/256)$ or approximately 5.

2.5 A Probabilistic Graphical Model for Color to Gray

For a given color image $\mathbf{I}$, the corresponding gray-scale image $\mathbf{I}_G$ should preserve visual cues as much as possible. There is a strong correlation between $\mathbf{I}_G$ and $\mathbf{I}$, and neighboring pixels in $\mathbf{I}_G$ are strongly correlated. Thus, we choose to use the Markov random fields, a popular probabilistic graphical model, to make use of this prior knowledge. Fig. 6 shows an undirected graphical model representing a Markov random field for color to gray in which $\mathbf{I}_G(x,y)$ is a gray value denoting the state of a pixel at (x,y) in the unknown gray-scale image and $\mathbf{I}(x,y)$ denotes the corresponding color pixels ($\mathbf{I}_H(x,y)$, $\mathbf{I}_C(x,y)$, and $\mathbf{I}_L(x,y)$) at (x,y) in the observed color image. This model contains two types of nodes: *observable nodes* (blue for pixels in the color image) and *hidden nodes* (gray for values in the corresponding gray-scale image). Edges are used to describe the relationships between/among nodes. These two types of nodes, respectively, form two layers: the *observable layer* and the *hidden layer*. The relationship between the hidden node and the observable node is represented by $p(\mathbf{I}(x,y)|\mathbf{I}_G(x,y))$, and the relationship between the neighboring hidden nodes is represented by $p(\mathbf{I}_G(x,y))$.

The color to gray model can be regarded as a process that transfers the extracted visual cues from the color image $\mathbf{I}$ to the gray-scale image $\mathbf{I}_G$ as much as possible. And this process can be formulated by the following MAP framework:

$$\mathbf{I}_G = \arg\max_{\mathbf{I}_G} p(\mathbf{I}_G \mid \mathbf{I}). \tag{12}$$

The probability function $p(\mathbf{I}_G \mid \mathbf{I})$ is given by

$$\begin{aligned} p(\mathbf{I}_G \mid \mathbf{I}) &= p(\mathbf{I}_G \mid \mathbf{I}_H, \mathbf{I}_C, \mathbf{I}_L) \\ &= p(\mathbf{I}_H, \mathbf{I}_C, \mathbf{I}_L \mid \mathbf{I}_G) p(\mathbf{I}_G) / p(\mathbf{I}_H, \mathbf{I}_C, \mathbf{I}_L) \\ &\propto p(\mathbf{I}_H, \mathbf{I}_C, \mathbf{I}_L \mid \mathbf{I}_G) p(\mathbf{I}_G), \end{aligned} \tag{13}$$

where $\mathbf{I}_H$, $\mathbf{I}_C$, and $\mathbf{I}_L$ represent hue, chroma, and lightness channels, respectively. The color image $\mathbf{I}$ is a combination of $\mathbf{I}_H$, $\mathbf{I}_C$, and $\mathbf{I}_L$, so we have the first line of (13). By using the Bayes' theorem to $p(\mathbf{I}_G \mid \mathbf{I}_H, \mathbf{I}_C, \mathbf{I}_L)$, we obtain the second line. Finally, because $p(\mathbf{I}_H, \mathbf{I}_C, \mathbf{I}_L)$ is a constant, we get the third line.

The relationship between hidden nodes and observable nodes is described by $p(\mathbf{I}_H, \mathbf{I}_C, \mathbf{I}_L \mid \mathbf{I}_G)$, which is usually given by the conditional Gaussian to handle visual cues extracted from three different channels in the original color image. The relationship between the neighbor nodes is described by $p(\mathbf{I}_G)$, which is utilized to keep the gradient direction consistency in the proposed approach.

In an imaging system, chroma and lightness are correlated. For example, during the imaging process of a scene (or an object), the high lightness will cause the low chroma because of the mirror reflection of illumination. However, to perceive a picture, a visual system perceives the chroma (and the lightness) from pixels directly. Therefore, the correlation between the chroma and the lightness does away and we should model them independently. Based on the color channel perception priority (the third visual cue observed by psychologist), defined in (10) and (11), $\mathbf{I}_H$ is independent of $\mathbf{I}_C$ and $\mathbf{I}_L$, $\mathbf{I}_C$ is independent of $\mathbf{I}_L$ but depends on $\mathbf{I}_H$, and $\mathbf{I}_L$ depends on both $\mathbf{I}_H$ and $\mathbf{I}_C$. That is $\mathbf{I}_H$, $\mathbf{I}_C$, and $\mathbf{I}_L$ form a Markov chain and the color to gray model defined in (13) can be transformed to:

$$\begin{aligned} \max_{\mathbf{I}_G} p(\mathbf{I}_G \mid \mathbf{I}) &\propto \max_{\mathbf{I}_G} p(\mathbf{I}_H, \mathbf{I}_C, \mathbf{I}_L \mid \mathbf{I}_G) p(\mathbf{I}_G) \\ &= \max_{\mathbf{I}_G} p(\mathbf{I}_H \mid \mathbf{I}_G) p(\mathbf{I}_C \mid \mathbf{I}_G, \mathbf{I}_H) \\ &\quad p(\mathbf{I}_L \mid \mathbf{I}_G, \mathbf{I}_H, \mathbf{I}_C) p(\mathbf{I}_G). \end{aligned} \tag{14}$$

To obtain $\mathbf{I}_G$ that maximizes $p(\mathbf{I}_G \mid \mathbf{I})$, it is necessary to compute $p(\mathbf{I}_H \mid \mathbf{I}_G)$, $p(\mathbf{I}_C \mid \mathbf{I}_G, \mathbf{I}_H)$, $p(\mathbf{I}_L \mid \mathbf{I}_G, \mathbf{I}_H, \mathbf{I}_C)$, and $p(\mathbf{I}_G)$ in advance. Here, $p(\mathbf{I}_H \mid \mathbf{I}_G)$ is the probability of transferring the extracted visual cues in hue channel of color image to the gray-scale image, $p(\mathbf{I}_C \mid \mathbf{I}_G, \mathbf{I}_H)$ is the probability of transferring the extracted visual cues in chroma channel of color image to the gray-scale image by taking into account the priority of hue, and $p(\mathbf{I}_L \mid \mathbf{I}_G, \mathbf{I}_H, \mathbf{I}_C)$ is the probability of transferring the extracted visual cues in lightness channel of color image to the gray-scale image by taking into account priorities of hue and chroma. The extracted visual cue of each channel is defined by the gradient with the corresponding priority in terms of the definitions in Sections 2.2-2.4. Note that the directions of gradients $\nabla \mathbf{I}_H(x,y)$, $\nabla \mathbf{I}_C(x,y)$, and $\nabla \mathbf{I}_L(x,y)$ are important to infer the direction of $\nabla \mathbf{I}_G(x,y)$ but consistency among them is unnecessary. Therefore, it is essential to discuss the directions of $\nabla \mathbf{I}_H(x,y)$, $\nabla \mathbf{I}_C(x,y)$, $\nabla \mathbf{I}_L(x,y)$, and $\nabla \mathbf{I}_G(x,y)$ before we define $p(\mathbf{I}_H \mid \mathbf{I}_G)$, $p(\mathbf{I}_C \mid \mathbf{I}_G, \mathbf{I}_H)$, $p(\mathbf{I}_L \mid \mathbf{I}_G, \mathbf{I}_H, \mathbf{I}_C)$, and $p(\mathbf{I}_G)$.

According to the color channel perception priority, we will use the direction of hue $\nabla \mathbf{I}_H(x,y)$, or we will use the direction of chroma $\nabla \mathbf{I}_C(x,y)$ if that of hue is unavailable, or we will use the direction of lightness $\nabla \mathbf{I}_L(x,y)$ if either the direction of hue or that of chroma is not available. As a consequence, we define a set of sign functions, which will be utilized to define the direction of $\nabla \mathbf{I}_G(x,y)$:

$$\mathrm{sign}^\circ\big(\nabla \mathbf{I}_L^{(\cdot)}(x,y)\big) = \begin{cases} +1, & \nabla \mathbf{I}_L^{(\cdot)}(x,y) > 0, \\ -1, & \nabla \mathbf{I}_L^{(\cdot)}(x,y) < 0, \end{cases} \tag{15}$$

$$\begin{aligned} &\mathrm{sign}^\circ\big(\nabla \mathbf{I}_C^{(\cdot)}(x,y)\big) \\ &= \begin{cases} +1, & \nabla \mathbf{I}_C^{(\cdot)}(x,y) > 0, \\ -1, & \nabla \mathbf{I}_C^{(\cdot)}(x,y) < 0, \quad \text{and} \\ \mathrm{sign}^\circ\big(\nabla \mathbf{I}_L^{(\cdot)}(x,y)\big), & \nabla \mathbf{I}_C^{(\cdot)}(x,y) = 0, \end{cases} \end{aligned} \tag{16}$$

$$\mathrm{sign}^\circ\big(\nabla \mathbf{I}_H^{(\cdot)}(x,y)\big) = \begin{cases} +1, & \nabla \mathbf{I}_H^{(\cdot)}(x,y) > 0, \\ -1, & \nabla \mathbf{I}_H^{(\cdot)}(x,y) < 0, \\ \mathrm{sign}^\circ\big(\nabla \mathbf{I}_C^{(\cdot)}(x,y)\big), & \nabla \mathbf{I}_H^{(\cdot)}(x,y) = 0, \end{cases} \tag{17}$$

where $\nabla \mathbf{I}_L^{(\cdot)}(x,y)$ means x or y direction of $\nabla \mathbf{I}_L(x,y)$ and this is same for $\nabla \mathbf{I}_C^{(\cdot)}(x,y)$ and $\nabla \mathbf{I}_H^{(\cdot)}$. According to the color channel perception priority, $\mathrm{sign}^\circ(\nabla \mathbf{I}_H^{(\cdot)}(x,y))$ should be determined by $\mathrm{sign}^\circ(\nabla \mathbf{I}_C^{(\cdot)}(x,y))$ only if $\nabla \mathbf{I}_H^{(\cdot)}(x,y) = 0$, i.e., (17). And only if $\nabla \mathbf{I}_C^{(\cdot)}(x,y) = 0$, should $\mathrm{sign}^\circ(\nabla \mathbf{I}_C^{(\cdot)}(x,y))$ be given by $\mathrm{sign}^\circ(\nabla \mathbf{I}_L^{(\cdot)}(x,y))$, i.e., (16).

With $\mathrm{sign}^\circ(\nabla \mathbf{I}_H^{(\cdot)}(x,y))$, $\mathrm{sign}^\circ(\nabla \mathbf{I}_C^{(\cdot)}(x,y))$, and $\mathrm{sign}^\circ\big(\nabla \mathbf{I}_L^{(\cdot)}(x,y)\big)$, we can define $p(\mathbf{I}_H \mid \mathbf{I}_G)$, $p(\mathbf{I}_C \mid \mathbf{I}_G, \mathbf{I}_H)$, and $p(\mathbf{I}_L \mid \mathbf{I}_G, \mathbf{I}_H, \mathbf{I}_C)$ as

$$\begin{aligned} &p(\mathbf{I}_H(x,y) \mid \mathbf{I}_G(x,y)) \\ &\quad \propto \exp\big(-\sigma_H^{-2} \| \nabla \mathbf{I}_G(x,y) - \mathrm{sign}^\circ(\nabla \mathbf{I}_H(x,y)) \\ &\qquad P_H(x,y) \mid \nabla \mathbf{I}_H(x,y)| \|^2\big), \end{aligned} \tag{18}$$

$$\begin{aligned} &p(\mathbf{I}_C(x,y) \mid \mathbf{I}_G(x,y), \mathbf{I}_H(x,y)) \\ &\quad \propto \exp\big(-\sigma_C^{-2} \| \nabla \mathbf{I}_G(x,y) - \mathrm{sign}^\circ(\nabla \mathbf{I}_C(x,y)) \\ &\qquad P_C(x,y) \mid \nabla \mathbf{I}_C(x,y)| \|^2\big), \end{aligned} \tag{19}$$

and

$$\begin{aligned} &p(\mathbf{I}_L(x,y) | \mathbf{I}_G(x,y), \mathbf{I}_H(x,y), \mathbf{I}_C(x,y)) \\ &\quad \propto \exp(-\sigma_L^{-2} \| \nabla \mathbf{I}_G(x,y) - \mathrm{sign}^\circ(\nabla \mathbf{I}_L(x,y)) \\ &\qquad P_L(x,y) | \nabla \mathbf{I}_L(x,y)| \|^2), \end{aligned} \tag{20}$$

where the item $\| \nabla \mathbf{I}_G(x,y) - \mathrm{sign}^\circ(\nabla \mathbf{I}_H(x,y)) P_H(x,y) \mid \nabla \mathbf{I}_H(x,y)| \|$ in (18) reflects the visual difference between $\mathbf{I}_G$ and $\mathbf{I}_H$ wherein the magnitude and direction of $\nabla \mathbf{I}_H(x,y)$ is, respectively, adjusted by the hue priority factor $P_H(x,y)$ and $\mathrm{sign}^\circ(\nabla \mathbf{I}_H(x,y))$, the item $\| \nabla \mathbf{I}_G(x,y) - \mathrm{sign}^\circ(\nabla \mathbf{I}_C(x,y)) P_C(x,y) | \nabla \mathbf{I}_C(x,y)| \|$ in (19) reflects the visual difference between $\mathbf{I}_G$ and $\mathbf{I}_C$ wherein the magnitude and direction of $\nabla \mathbf{I}_C(x,y)$ are, respectively, adjusted by the chroma priority $P_C(x,y)$ and $\mathrm{sign}^\circ(\nabla \mathbf{I}_H(x,y))$, and $\| \nabla \mathbf{I}_G(x,y) - \mathrm{sign}^\circ(\nabla \mathbf{I}_H(x,y)) P_L | \nabla \mathbf{I}_L(x,y)| \|$ in (20) reflects the visual difference between $\mathbf{I}_G$ and $\mathbf{I}_L$ wherein the magnitude and the direction of $\nabla \mathbf{I}_L(x,y)$ are, respectively, adjusted by the lightness priority factor $P_L(x,y)$ and $\mathrm{sign}^\circ(\nabla \mathbf{I}_H(x,y))$.

In addition, it is natural to align the direction of $\nabla \mathbf{I}_G(x,y)$ to the direction of $\nabla \mathbf{I}_H(x,y)$ ($\nabla \mathbf{I}_C(x,y)$ or $\nabla \mathbf{I}_L(x,y)$) according to the color perception priority defined by (15)-(17) to achieve the direction consistency which leads to a good visual effect of the transformed gray-scale image. Therefore, we can define the prior $p(\mathbf{I}_G)$ as

$$p(\mathbf{I}_G(x,y)) = \begin{cases} 1, & \mathrm{sign}(\nabla \mathbf{I}_G(x,y)) \mathrm{sign}^\circ(\nabla \mathbf{I}_H(x,y)) \geq 0, \\ 0, & \mathrm{sign}(\nabla \mathbf{I}_G(x,y)) \mathrm{sign}^\circ(\nabla \mathbf{I}_H(x,y)) < 0. \end{cases} \tag{21}$$

Equation (21) is a constraint to represent the construction of a gray-scale image via $\mathrm{sign}^\circ(\nabla \mathbf{I}_H(x,y))$, and exactly reflects the color channel perception priority defined by the Visual Cue 3. Based on (18)-(21), the MAP for color to gray defined in (14) can be furthered as

$$\max_{\mathbf{I}_G} \sum_{(x,y)\in \mathbf{I}} \begin{bmatrix} -\sigma_L^{-2} \| \nabla \mathbf{I}_G(x,y) - \mathrm{sign}^\circ(\nabla \mathbf{I}_H(x,y)) \\ \hfill P_L(x,y) \ |\nabla \mathbf{I}_L(x,y)| \|^2 \\ -\sigma_C^{-2} \| \nabla \mathbf{I}_G(x,y) - \mathrm{sign}^\circ(\nabla \mathbf{I}_H(x,y)) \\ \hfill P_C(x,y) \ |\nabla \mathbf{I}_C(x,y)| \|^2 \\ -\sigma_H^{-2} \| \nabla \mathbf{I}_G(x,y) - \mathrm{sign}^\circ(\nabla \mathbf{I}_H(x,y)) \\ \hfill P_H(x,y) \ |\nabla \mathbf{I}_H(x,y)| \|^2 \end{bmatrix}, \tag{22}$$

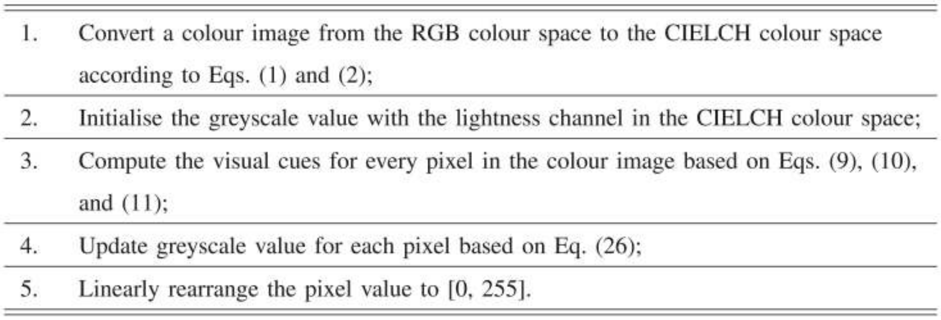
where the detailed procedure to obtain (22) is given in Appendix A.

The MAP for color to gray obtained in (22) can be formed as an integral minimization-based optimization procedure [23], [24]:

$$\begin{aligned} &\int\!\!\int F(\nabla \mathbf{I}_G(x,y), \mathbf{H}'(x,y), \mathbf{C}'(x,y), \mathbf{L}'(x,y)) dxdy \\ &= \int\!\!\int \begin{bmatrix} \| \nabla \mathbf{I}_G(x,y) - \mathbf{H}'(x,y) \|^2 + \\ \lambda_1 \| \nabla \mathbf{I}_G(x,y) - \mathbf{C}'(x,y) \|^2 + \lambda_2 \| \nabla \mathbf{I}_G(x,y) \\ -\mathbf{L}'(x,y) \|^2 \end{bmatrix} dxdy \\ &= \int\!\!\int \begin{bmatrix} \left(\frac{\partial \mathbf{I}_G(x,y)}{\partial x} - \mathbf{H}'_x \right)^2 + \left(\frac{\partial \mathbf{I}_G(x,y)}{\partial y} - \mathbf{H}'_y \right)^2 \\ + \lambda_1 \left(\left(\frac{\partial \mathbf{I}_G(x,y)}{\partial x} - \mathbf{C}'_x \right)^2 + \left(\frac{\partial \mathbf{I}_G(x,y)}{\partial y} - \mathbf{C}'_y \right)^2 \right) \\ + \lambda_2 \left(\left(\frac{\partial \mathbf{I}_G(x,y)}{\partial x} - \mathbf{L}'_x \right)^2 + \left(\frac{\partial \mathbf{I}_G(x,y)}{\partial y} - \mathbf{L}'_y \right)^2 \right) \end{bmatrix} dxdy. \end{aligned} \tag{23}$$

The detailed procedure for reformulating (22) to (23) is given in Appendix B. Based on our experiences, we can set λ_1 and λ_2 as one. Experimental results show that this setting is fine to get good results for various types of images.

TABLE 1
Visual Cue Preservation Algorithm for Color to Gray



1.	Convert a colour image from the RGB colour space to the CIELCH colour space according to Eqs. (1) and (2);
2.	Initialise the greyscale value with the lightness channel in the CIELCH colour space;
3.	Compute the visual cues for every pixel in the colour image based on Eqs. (9), (10), and (11);
4.	Update greyscale value for each pixel based on Eq. (26);
5.	Linearly rearrange the pixel value to [0, 255].

∗Because this algorithm depends on local measures of gradient and smoothness, it is possible that the same color at two different locations in the original image could be converted into two different scalar values in the final gray-scale image.

According to the Variational Principle, a function $\mathbf{I}_G$ that minimizes the integral in (23) must satisfy the Euler-Lagrange equation [24],

$$\frac{\partial F}{\partial \mathbf{I}_G} - \frac{d}{dx}\frac{\partial F}{\partial (\mathbf{I}_G)_x} - \frac{d}{dy}\frac{\partial F}{\partial (\mathbf{I}_G)_y} = 0, \tag{24}$$

which is a partial differential equation in $\mathbf{I}_G$. By substituting F, we obtain:

$$\begin{bmatrix} -2\left(\frac{\partial^2 \mathbf{I}_G}{\partial x^2} - \frac{\partial \mathbf{H}'_x}{\partial x}\right) - 2\lambda_1\left(\frac{\partial^2 \mathbf{I}_G}{\partial x^2} - \frac{\partial \mathbf{C}'_x}{\partial x}\right) - 2\lambda_2\left(\frac{\partial^2 \mathbf{I}_G}{\partial x^2} - \frac{\partial \mathbf{L}'_x}{\partial x}\right) \\ -2\left(\frac{\partial^2 \mathbf{I}_G}{\partial y^2} - \frac{\partial \mathbf{H}'_y}{\partial y}\right) - 2\lambda_1\left(\frac{\partial^2 \mathbf{I}_G}{\partial y^2} - \frac{\partial \mathbf{C}'_y}{\partial y}\right) - 2\lambda_2\left(\frac{\partial^2 \mathbf{I}_G}{\partial y^2} - \frac{\partial \mathbf{L}'_y}{\partial y}\right) \end{bmatrix} = 0. \tag{25}$$

Dividing (25) by 2 and rearranging terms, we obtain

$$(1 + \lambda_1 + \lambda_2)\nabla^2 \mathbf{I}_G = \mathrm{div}\mathbf{H}' + \lambda_1 \mathrm{div}\mathbf{C}' + \lambda_2 \mathrm{div}\mathbf{L}', \tag{26}$$

where $\nabla^2 \mathbf{I}_G(x,y) = \partial^2_{x^2}\mathbf{I}_G(x,y) + \partial^2_{y^2}\mathbf{I}_G(x,y)$ is the Laplacian operator;

$$\mathrm{div}\mathbf{H}' = \frac{\partial \mathbf{H}'_x}{\partial x} + \frac{\partial \mathbf{H}'_y}{\partial y}$$

is the divergence of gradient field and the same for $\mathrm{div}\mathbf{C}'$ and $\mathrm{div}\mathbf{L}'$. In the following part, we show how to solve (26) and implement the color to gray.

The detailed implementation procedure of the proposed color to gray algorithm is given in Table 1. Specifically, for step 4, we compute Laplacian and divergence value for pixel (x, y) based on the following two numerical approximations, respectively:

$$\nabla^2 \mathbf{I}_G(x,y) = \mathbf{I}_G(x+1,y) + \mathbf{I}_G(x-1,y) + \mathbf{I}_G(x,y+1) + \mathbf{I}_G(x,y-1) - 4\mathbf{I}_G(x,y), \tag{27}$$

$$\mathrm{div}\mathbf{H}'(x,y) = \mathbf{H}'_x(x+1,y) - \mathbf{H}'_x(x,y) + \mathbf{H}'_y(x,y+1) - \mathbf{H}'_y(x,y), \tag{28}$$

where $\mathbf{H}'_x(x,y)$ and $\mathbf{H}'_y(x,y)$ are the x component and y component of $\mathbf{H}'(x,y)$, respectively. The computations of $\mathrm{div}\mathbf{C}'(x,y)$ and $\mathrm{div}\mathbf{L}'(x,y)$ are the same as $\mathrm{div}\mathbf{H}'(x,y)$ by substituting $\mathbf{C}'(x,y)$ and $\mathbf{L}'(x,y)$ into (28), respectively.

At the boundary of an image, we assume derivatives around the boundary are 0. Based on (27) and (28), (26) can be transformed to a linear equation system $\mathbf{A}\mathbf{x} = \mathbf{b}$, where $\mathbf{A}$ is a sparse coefficient matrix, $\mathbf{x}$ consists of unknown gray-scale values, and $\mathbf{b}$ comes from $\mathrm{div}\mathbf{H}' + \lambda_1\mathrm{div}\mathbf{C}' + \lambda_2\mathrm{div}\mathbf{L}'$. The detailed procedure to construct the linear system is given in Appendix C. For implementation, the sparse LU decomposition solver [25] or Multigrid solver [26] can be utilized to form the solution for (26). To further speed up this step, the GPU solver [27] can be applied to replace the sparse LU decomposition solver.

In Fig. 7, we show visual cue maps. In detail, the first column is the original color image. The second column shows the color maps in the CIELCH color space. According to the discussion at the beginning of Section 2.1, e.g., (3) and (5), there are two maps for the hue channel, so there are two rows for the hue channel. The third column shows the hue value expectation E_{M_H} and that of E_{M_C}, defined in (3) and (4), and the associated Gaussian blurred maps are shown in the fourth column. Columns 5-7, respectively, show the maps of visual cue 1, i.e., color spatial consistencies U_H and U_C; those of the visual cue 3, i.e., color channel perception priorities P_C and P_L, and those of the visual cue 2, i.e., image structure information. The last column shows the transferred gray-scale image. In the original color image shown in Fig. 7, based on the row-column notation, we set (1, 2) and (2, 2) with different hue values but the same chroma and lightness, (2, 1) and (2, 2) with different lightness values but the same hue and chroma, and (2, 3) and (2, 2) with different chroma values but the same hue and lightness. In the output gray-scale image, the differences between (1, 2) and (2, 2), between (2, 3) and (2, 2), and between (2, 1) and (2, 2) are decreasing. This figure also shows that the same color at two different locations (e.g., the (1, 1) block and the (4, 4) block) in the original image are converted into two different scalar values (255 and 228, respectively) in the final gray-scale image.

3 Experiment and Result Analysis

In this section, we first evaluate the proposed VCP algorithm for color to gray in comparison with well-known techniques, which are the color to gray function in the Adobe Photoshop (Photoshop) [3], the Interactive Color2Gray (ICG) [2], the

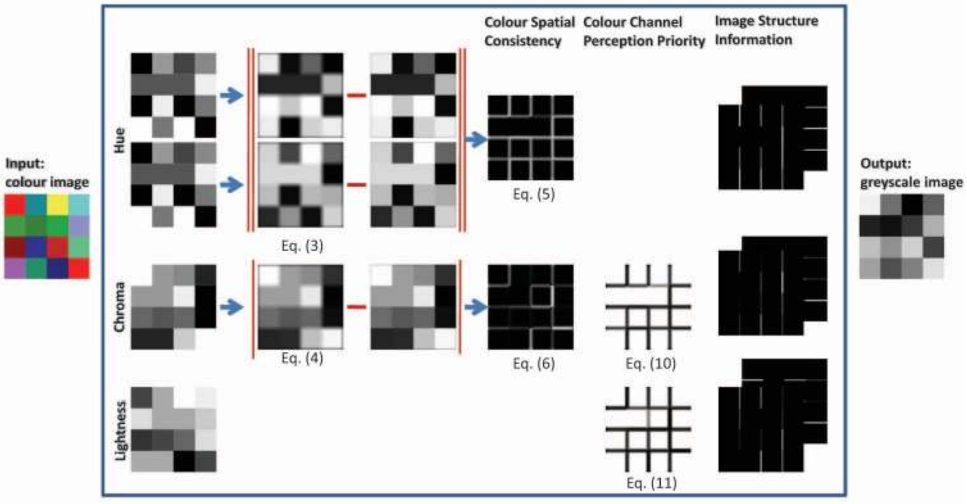
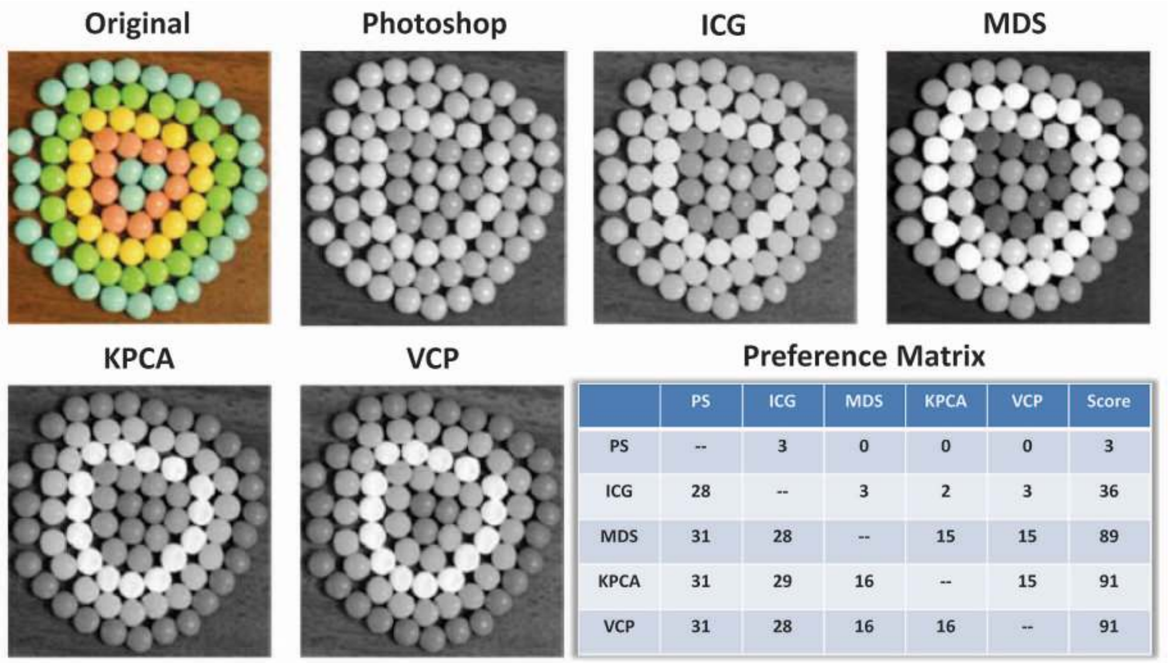


Fig. 7. Visual cue maps.



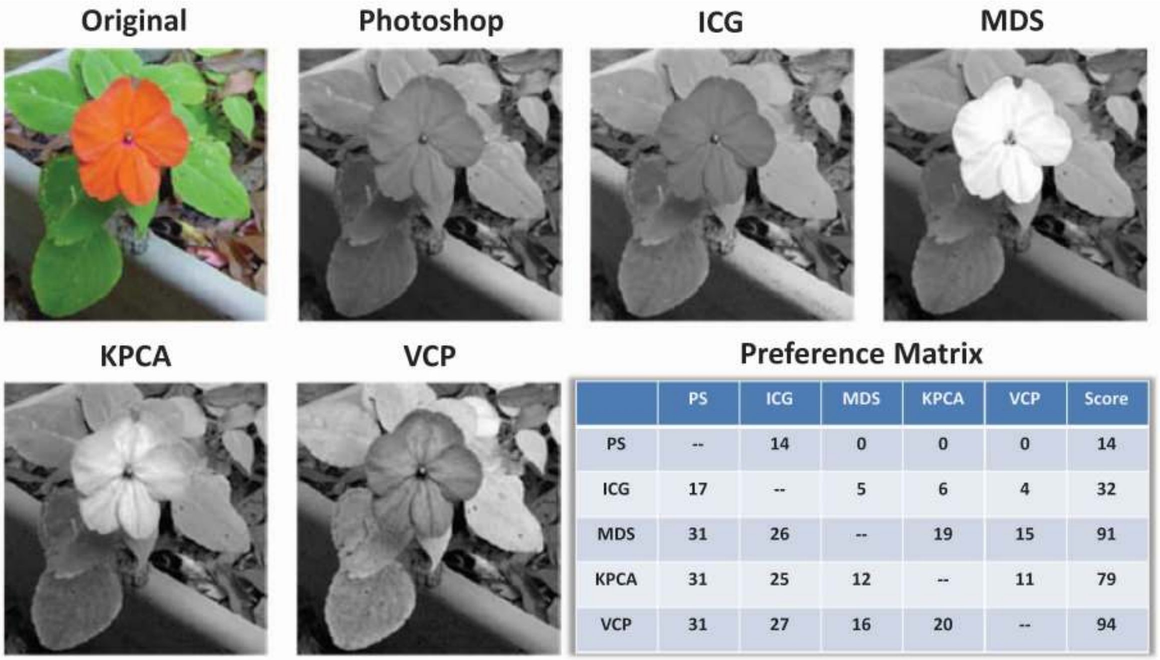
	PS	ICG	MDS	KPCA	VCP	Score
PS	--	3	0	0	0	3
ICG	28	--	3	2	3	36
MDS	31	28	--	15	15	89
KPCA	31	29	16	--	15	91
VCP	31	28	16	16	--	91

Fig. 8. For this case, KPCA and the proposed VCP achieved the best performance, and MDS performed comparably to the KPCA and VCP. The background texture obtained by MDS was dark and turbid. In the preference matrix, an entry means the number of subjects determined the method corresponding to the row performs better than the method corresponding to the column for this color image.

multidimensional scaling color to gray (MDS) [7], and the kernel principal components analysis (KPCA) [9], in terms of effectiveness and efficiency. Then, we discuss the influence of the sampling window size, the only free parameter in the proposed VCP, defined in Section 2.2 for color to gray. Finally, we provide additional examples to show the effectiveness of VCP.

3.1 Comparison with Well-Known Algorithms

Results of a subjective evaluation are directly given by human observers, so it is probably the best way to assess the performance of color to gray. This is because human observers are the ultimate receivers of the visual information contained in an image. Therefore, we choose to adopt the *paired comparison* [28] based user study to evaluate the effectiveness of the proposed VCP. It is worth emphasizing that both rating and ranking are not suitable here. This is because both of them are complex for an observer to perform and also would be an unnatural task for the observer leading to distorted results. The paired comparison is a method to present each subject with a pair of (color to gray) transformed images yielded by two different algorithms based on a same original color image. Participants are required to indicate a preference image for one of the two transformed gray-scale images compared to the reference. Evaluation results are then stored in the preference matrix. For example, considering the preference matrix in Fig. 8, the entry in the column "PS" (stands for the Color2Gray function in Photoshop) and the row "ICG" (stands for the Interactive Color2Gray [2]) is 28, which

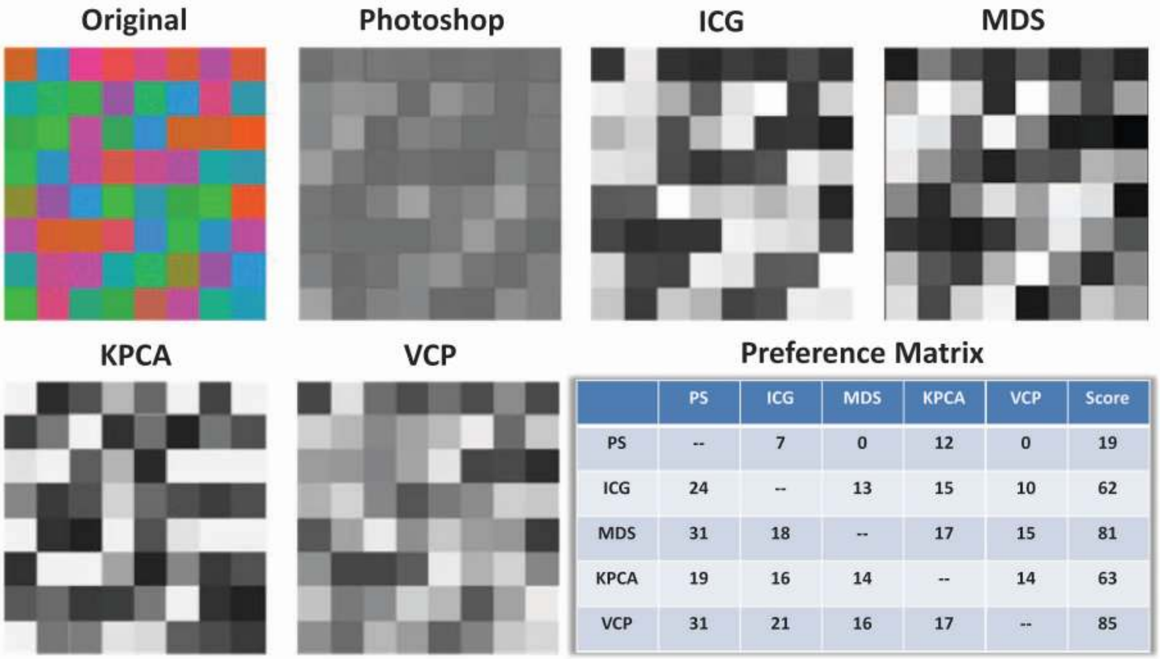


	PS	ICG	MDS	KPCA	VCP	Score
PS	--	14	0	0	0	14
ICG	17	--	5	6	4	32
MDS	31	26	--	19	15	91
KPCA	31	25	12	--	11	79
VCP	31	27	16	20	--	94

Fig. 9. For this case, VCP performed best and MDS performed comparably to VCP. KPCA provided the best visual effects for the flower but failed to distinguish green leaves and brown leaves. MDS transformed the flower brightly but subjects did not like this effect. In the preference matrix, an entry means the number of subjects who determined the method corresponding to the row performs better than the method corresponding to the column for this color image.

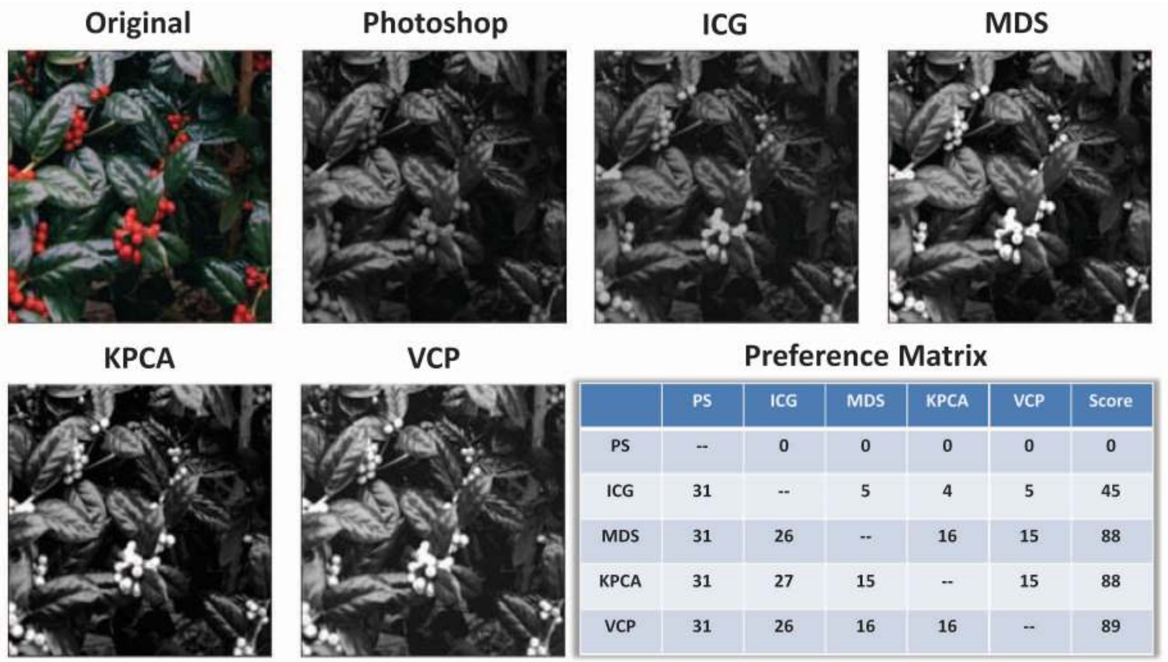
means 28 subjects considered ICG performed better than Photoshop for this color image.

In all experiments, 31 participants, including 9 females and 22 males, made the paired comparison and filled the preference matrix. The column of "Score" is the summation over columns. The higher the value of a corresponding algorithm is, the better the algorithm is, e.g., both KPCA and our method were deemed as the best methods to transform the top-left color image in Fig. 8. In Figs. 8, 9, 10, and 11, "Photoshop," "ICG," "MDS," "KPCA," and "VCP" in each table stand for the color to gray function in Adobe Photoshop, the Interactive Color2Gray [2], the Multidimen-



	PS	ICG	MDS	KPCA	VCP	Score
PS	--	7	0	12	0	19
ICG	24	--	13	15	10	62
MDS	31	18	--	17	15	81
KPCA	19	16	14	--	14	63
VCP	31	21	16	17	--	85

Fig. 10. For this case, VCP outperformed other algorithms and MDS performed comparably to our method. MDS provided good contrast but failed to distinguish some neighbor cells well, e.g., the cells (1,7) and (2,7). KPCA and ICG mixed some neighbor cells. VCP failed to provide good contrast but can distinguish neighbor cells well. In the preference matrix, an entry means the number of subjects who determined the method corresponding to the row performs better than the method corresponding to the column for this color image.



Original

Photoshop

ICG

MDS

KPCA

VCP

Preference Matrix

	PS	ICG	MDS	KPCA	VCP	Score
PS	--	0	0	0	0	0
ICG	31	--	5	4	5	45
MDS	31	26	--	16	15	88
KPCA	31	27	15	--	15	88
VCP	31	26	16	16	--	89

Fig. 11. For this case, VCP outperformed others while MDS and KPCA were comparable to VCP. Note the bottom-right dark green leaves in the original color image could not be conveniently observed in the MDS and KPCA transformed gray-scale images. That is, VCP can preserve more details in the color image than that of MDS and KPCA. In the preference matrix, an entry means the number of subjects who determined the method corresponding to the row performs better than the method corresponding to the column for this color image.

sional scaling color to gray [7], KPCA for color to gray [9], and the proposed algorithm. All of the color images in Figs. 8, 9, 10, and 11 are adopted from [7].

For the proposed VCP-based color to gray, there are three parameters but only one free parameter (i.e., the size of sampling windows), which is set automatically for different color images. In detail, for all experiments, we set $\lambda_1 = \lambda_2 = 1$ (please see Section 3.2) because it is simple and, meanwhile, VCP also performs well with this setting for various images. The size of sampling windows is the mean size of all segments, which can be obtained by an image segmentation algorithm, e.g., the meanshift used in this paper. In summary, we need not to tune parameters for VCP-based color to gray.

For fair comparison, parameters in algorithms are tuned to the best setting respectively to achieve best performance for ICG and KPCA-based color to gray, and the results of MDS-based color to gray are obtained from [7] directly. In addition, the original color images and results of MDS-based color to gray were obtained from [7] directly.

As shown in Figs. 8, 9, 10, and 11, with the color to gray function in Photoshop, the objects cannot be distinguished in transformed gray-scale images, although they are distinguishable in the original color images. With multidimensional scaling color to gray [7], in comparing with Photoshop color to gray function, although more visual information is preserved in the transformed gray-scale image, a lot of subtle details are lost, e.g., the background texture in the first color image in Fig. 8 and the drape of the flower in the second color image in Fig. 9. Interactive Color2Gray [2] fails to distinguish different colors in the first color image in Fig. 8. Although KPCA [9] produces reasonable results for large-scale objects in original color images, it makes subtle visual appearance vague, e.g., the background in the first color image in Fig. 8 and the difference between the flower and the leaves (or the difference between the green leaves and the brown leaves) in the second color image in Fig. 9. For all cases, the new VCP method works better than others, e.g., for the third color image in Fig. 10, it distinguishes all different color blocks well.

Both the time and the space costs are low for the color to gray function in Photoshop, which is a linear transformation method. However, its reproducing results are usually poor because important visual cues in original color images cannot be preserved in the transformed gray-scale images. KPCA can usually produce reasonable results, but it fails to manipulate high-resolution images because both time and space costs are unacceptable for today's computers. Interactive Color2Gray [2] requires complex human-computer interactions to adjust parameters to obtain a reasonable transformation result, which preserves user preferred visual cues. Additionally, this algorithm is inefficient for high-resolution images. Multidimensional scaling color to gray [7] usually produces reasonable results, as shown in Figs. 8, 9, 10, and 11, but its time cost is very heavy for images with a large number of colors. In Table 2, we evaluate the average time costs of KPCA, Interactive Color2Gray, multidimensional scaling color to gray, and the proposed VCP (by Multigrid solver) for color images in different sizes. As shown in this table, VCP is efficient in comparison with other algorithms. Experiments were carried out on a personal computer with Pentium IV 3.0 GHz CPU and 4 GB memory.

3.2 Experiments with Different Sampling Window Sizes

In this experiment, we study how free parameters affect the performance of VCP and how to set suitable parameters to

TABLE 2
Average Time Costs of Different Algorithms with Color Images in Different Sizes

Size	KPCA	ICG	MDS	VCP
30×30	5.3s	0.16s	$\approx$ 1s	0.012s
60×60	366.47s	2.94s	$\approx$ 4s	0.042s
100×60	3h	21.00s	$\approx$ 20s	0.134s
300×300	$\gg$ 3h	878.16s	$\approx$ 200s	0.925s
800×600	$\gg$ 3h	$\gg$ 3h	$\approx$ 1400s	3.251s
2400×1600	$\gg$ 3h	$\gg$ 3h	$\gg$ 3h	35.987s

$*$The time cost stands for transforming a set of color images to gray-scale ones, excluding the costs for interaction, parameter adjustment, and image preprocessing.

achieve a reasonable transformation result for a specific image. In the proposed algorithm, we have three free parameters only, which are the sampling windows size M, and λ_1, λ_2, as shown in Appendix B. Based on our experiences, we can always achieve reasonable results by setting both λ_1 and λ_2 to be 1 for images with different resolutions, i.e., for all experiments $\lambda_1 = \lambda_2 = 1$. We acknowledge it is possible to achieve a better performance by using λ_1 and λ_2 for a specific image. In the future, we will design a strategy to tune λ_1 and λ_2 automatically for different images. At the current stage, the only free parameter is the size of sampling windows.

To analyze the impact of the size of sampling windows on the performance of VCP for color to gray, we set up a set of experiments by varying the window size M continuously. In Fig. 12, we display three transformed gray-scale images corresponding to different M. Based on this figure, we have the following observations: 1) According to the second gray-scale image in Fig. 12, when M is comparable to M^* (the size of a main object, i.e., the yellow flower in the original color image), the transformed gray-scale image can preserve important visual cues in the original color image; 2) according to the first gray-scale image in Fig. 12, when M is much smaller than M^*, detailed visual cues with image noises will be transferred to the transformed gray-scale image; and 3) according to the third gray-scale image in Fig. 12, when M is much larger than M^*, important visual cues cannot be found in the transformed gray-scale image. Therefore, we can set M as M^* to achieve a good visual effect. Practically, we use the meanshift algorithm to segment the original color image to obtain a set of segments. The value of M is the mean size of these segments.

3.3 Further Examples for the Proposed Approach

Further experimental results are given in Fig. 13. In the first column (from the left), the difference between the red booth and the green trees is successfully transferred to the gray-scale image by the proposed method shown in the bottom row. The results shown in the middle row, obtained by Photoshop, cannot distinguish the color difference although this difference is an important visual cue in the color image. In the second column, the red fruit and the green leaves are distinguished well in the gray-scale result obtained by the proposed VCP. Moreover, flowers and leaves are assigned with different gray-scale values. We also test the new method on an artificial color dot pattern, shown in the third column for color blindness test. The result shows that our method transfers visual differences in the color image to the transformed gray-scale one, so VCP can be applied to the color blindness related applications. In the last column, we test our method on the famous impressionist painting by Monet. As the result in the third row shows, the sun, the boat, and the sea are well distinguishable. In summary, the proposed VCP for color to gray is effective, robust, and flexible to various color images. Additional results are given in Fig. 14.

4 Conclusion

Conventional color to gray algorithms achieved much but share one or more of three particular drawbacks: First, some of them do not produce gray-scale images that satisfactorily render the levels of contrasts and detail that can be perceived in the original color images, i.e., they may not preserve important visual cues because there are no clear definitions of visual cues; second, some may have very heavy computational costs and are not ready for practical applications; and finally, better quality results may depend upon adjustments to free parameters by human operators, which is not user friendly.

This paper proposed an algorithm that preserves newly well-defined visual cues in the transformed gray-scale image. In detail, the algorithm conveys visual cues which are constructed based on three color image understandings and are obtained in the CIELCH color space, under a

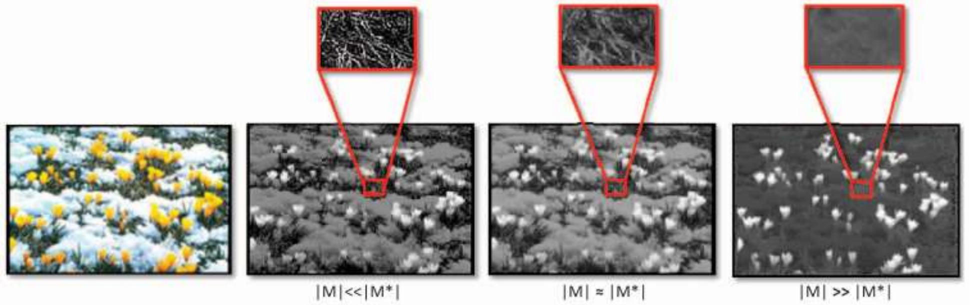


Fig. 12. The impact of the size of sampling windows for the proposed color to gray algorithm: The source image was obtained from the Internet.

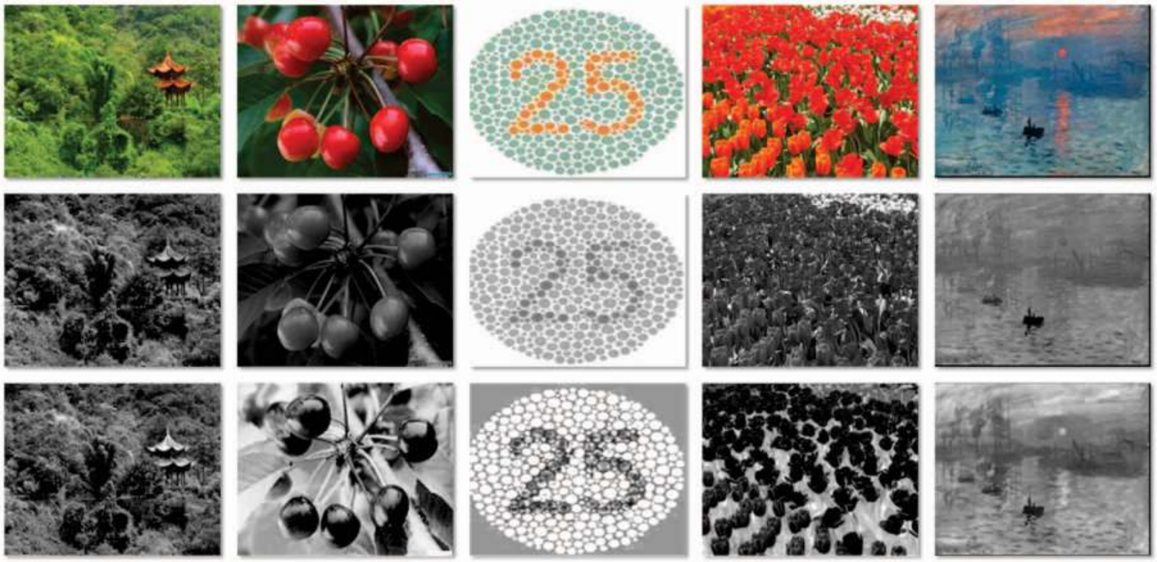


Fig. 13. Further examples for the proposed approach: The first row shows original color images, the second row shows gray-scale images transformed by the Photoshop color to gray function and the last row shows the gray-scale images transformed by the proposed algorithm. Source images were obtained from the Internet and courtesy of Monet (in the last two columns).

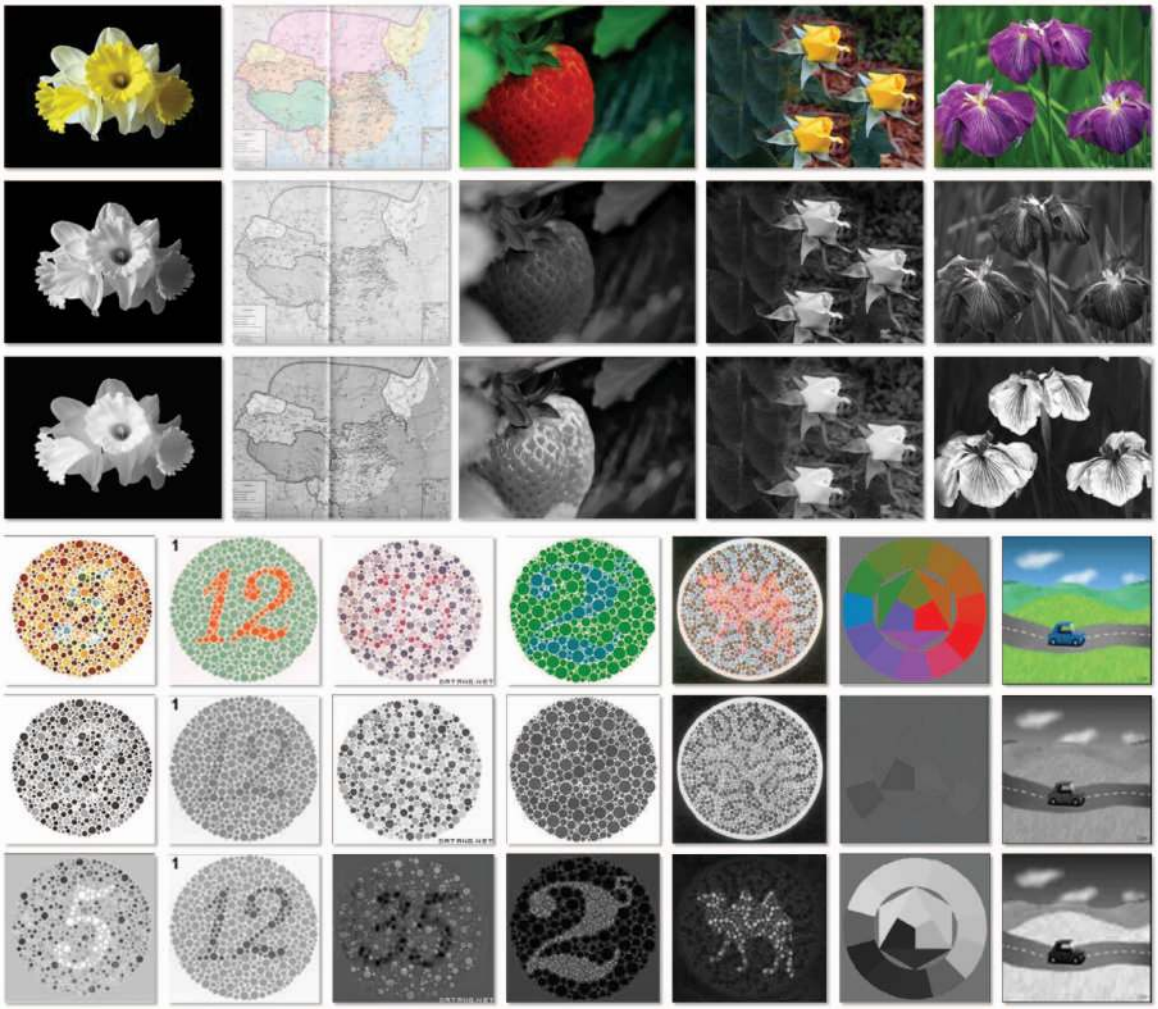


Fig. 14. Additional results: The first row shows original color images, the second row shows gray-scale images transformed by the Photoshop color to gray function, and the last row shows the gray-scale images transformed by the proposed algorithm. Source images were obtained from the Internet and courtesy of Bli (in the last two columns).

probabilistic graphical model. In addition, it is very fast and requires no human-computer interactions. Thorough empirical studies over a wide range of color images demonstrate the effectiveness and the efficiency of the new algorithm in reproducing gray-scale images from color images in comparison with popular algorithms.

Appendix A

The detailed procedure to obtain (22) is given below. Based on (18)-(21), the MAP framework for color to gray defined in (14) is given by:

$$
\begin{aligned}
&\max_{\mathbf{I}_G} p(\mathbf{I}_H \mid \mathbf{I}_G) p(\mathbf{I}_C \mid \mathbf{I}_G, \mathbf{I}_H) p(\mathbf{I}_L \mid \mathbf{I}_G, \mathbf{I}_H, \mathbf{I}_C) p(\mathbf{I}_G) \\
&= \max_{\mathbf{I}_G} \prod_{(x,y)\in\mathbf{I}} \begin{bmatrix} p(\mathbf{I}_H(x,y)|\mathbf{I}_G(x,y)) \\ \times\, p(\mathbf{I}_C(x,y) \mid \mathbf{I}_G(x,y), \mathbf{I}_H(x,y)) \\ \times\, p(\mathbf{I}_L(x,y)|\mathbf{I}_G(x,y), \mathbf{I}_H(x,y), \mathbf{I}_C(x,y)) \\ \times\, p(\mathbf{I}_G(x,y)) \end{bmatrix} \\
&\propto \max_{\mathbf{I}_G} \sum_{(x,y)\in\mathbf{I}} \begin{bmatrix} \ln p(\mathbf{I}_H(x,y)|\mathbf{I}_G(x,y)) \\ + \ln p(\mathbf{I}_C(x,y)|\mathbf{I}_G(x,y), \mathbf{I}_H(x,y)) \\ + \ln p(\mathbf{I}_L(x,y)|\mathbf{I}_G(x,y), \mathbf{I}_H(x,y), \mathbf{I}_C(x,y)) \end{bmatrix} \\
&= \max_{\mathbf{I}_G} \sum_{(x,y)\in\mathbf{I}} \begin{bmatrix} -\sigma_L^{-2} \|\nabla \mathbf{I}_G(x,y) - \mathrm{sign}^\circ(\nabla \mathbf{I}_H(x,y)) \\ P_L(x,y)\ |\nabla \mathbf{I}_L(x,y)|\|^2 \\ -\sigma_C^{-2} \|\nabla \mathbf{I}_G(x,y) - \mathrm{sign}^\circ(\nabla \mathbf{I}_H(x,y)) \\ P_C(x,y)\ |\nabla \mathbf{I}_C(x,y)|\|^2 \\ -\sigma_H^{-2} \|\nabla \mathbf{I}_G(x,y) - \mathrm{sign}^\circ(\nabla \mathbf{I}_H(x,y)) \\ P_H(x,y)\ |\nabla \mathbf{I}_H(x,y)|\|^2 \end{bmatrix},
\end{aligned}
$$

s.t. $p(\mathbf{I}_G) = 1$.

Appendix B

The detailed procedure for reformulating (22) to (23) is given below:

$$
\begin{aligned}
&\arg\max_{\mathbf{I}_G} \sum_{(x,y)\in\mathbf{I}} \begin{bmatrix} -\sigma_L^{-2} \|\nabla \mathbf{I}_G(x,y) - \mathrm{sign}^\circ(\nabla \mathbf{I}_H(x,y)) \\ P_L(x,y)\ |\nabla \mathbf{I}_L(x,y)|\|^2 \\ -\sigma_C^{-2} \|\nabla \mathbf{I}_G(x,y) - \mathrm{sign}^\circ(\nabla \mathbf{I}_H(x,y)) \\ P_C(x,y)\ |\nabla \mathbf{I}_C(x,y)|\|^2 \\ -\sigma_H^{-2} \|\nabla \mathbf{I}_G(x,y) - \mathrm{sign}^\circ(\nabla \mathbf{I}_H(x,y)) \\ P_H(x,y)\ |\nabla \mathbf{I}_H(x,y)|\|^2 \end{bmatrix} \\
&= \arg\min_{\mathbf{I}_G} \sum_{(x,y)\in\mathbf{I}} \begin{bmatrix} \sigma_L^{-2} \|\nabla \mathbf{I}_G(x,y) - \mathrm{sign}^\circ(\nabla \mathbf{I}_H(x,y)) \\ P_L(x,y)\ |\nabla \mathbf{I}_L(x,y)|\|^2 \\ +\sigma_C^{-2} \|\nabla \mathbf{I}_G(x,y) - \mathrm{sign}^\circ(\nabla \mathbf{I}_H(x,y)) \\ P_C(x,y)\ |\nabla \mathbf{I}_C(x,y)|\|^2 \\ +\sigma_H^{-2} \|\nabla \mathbf{I}_G(x,y) - \mathrm{sign}^\circ(\nabla \mathbf{I}_H(x,y)) \\ P_H(x,y)\ |\nabla \mathbf{I}_H(x,y)|\|^2 \end{bmatrix} \\
&= \arg\min_{\mathbf{I}_G} \sum_{(x,y)\in\mathbf{I}} \begin{bmatrix} \|\nabla \mathbf{I}_G(x,y) - \mathrm{sign}^\circ(\nabla \mathbf{I}_H(x,y)) \\ P_H(x,y)\ |\nabla \mathbf{I}_H(x,y)|\|^2 \\ +\lambda_1 \|\nabla \mathbf{I}_G(x,y) - \mathrm{sign}^\circ(\nabla \mathbf{I}_H(x,y)) \\ P_C(x,y)\ |\nabla \mathbf{I}_C(x,y)|\|^2 \\ +\lambda_2 \|\nabla \mathbf{I}_G(x,y) - \mathrm{sign}^\circ(\nabla \mathbf{I}_H(x,y)) \\ P_L(x,y)\ |\nabla \mathbf{I}_L(x,y)|\|^2 \end{bmatrix}.
\end{aligned}
$$

Let

$$
\begin{aligned}
\mathbf{H}'(x,y) &= \mathrm{sign}^\circ(\nabla \mathbf{I}_H(x,y)) P_H(x,y)\ |\nabla \mathbf{I}_H(x,y)|, \\
\mathbf{C}'(x,y) &= \mathrm{sign}^\circ(\nabla \mathbf{I}_H(x,y)) P_C(x,y)\ |\nabla \mathbf{I}_C(x,y)|, \\
\mathbf{L}'(x,y) &= \mathrm{sign}^\circ(\nabla \mathbf{I}_L(x,y)) P_L(x,y)\ |\nabla \mathbf{I}_L(x,y)|.
\end{aligned}
$$

By treating the above equations continuously, they can be finally formulated as:

$$
\int\!\!\int F(\nabla \mathbf{I}_G(x,y), \mathbf{H}'(x,y), \mathbf{C}'(x,y), \mathbf{L}'(x,y))\,dxdy.
$$

Appendix C

The detailed procedure for reformulating (26) to a linear equation system $\mathbf{Ax} = \mathbf{b}$ is given below. By substituting (27) and (28) into (26), for each pixel, we have

$$
\begin{aligned}
&(1 + \lambda_1 + \lambda_2)(\mathbf{I}_G(x+1,y) + \mathbf{I}_G(x-1,y) + \mathbf{I}_G(x,y+1) \\
&\quad + \mathbf{I}_G(x,y-1) - 4\mathbf{I}_G(x,y)) \\
&= \begin{bmatrix} \mathbf{H}'_x(x+1,y) - \mathbf{H}'_x(x,y) + \mathbf{H}'_y(x,y+1) - \mathbf{H}'_y(x,y) \\ +\lambda_1(\mathbf{C}'_x(x+1,y) - \mathbf{C}'_x(x,y) + \mathbf{C}'_y(x,y+1) - \mathbf{C}'_y(x,y)) \\ +\lambda_2(\mathbf{L}'_x(x+1,y) - \mathbf{L}'_x(x,y) + \mathbf{L}'_y(x,y+1) - \mathbf{L}'_y(x,y)) \end{bmatrix}.
\end{aligned}
$$

By reorganizing the left-hand side of the above equation, we have

$$
\begin{aligned}
&(1 + \lambda_1 + \lambda_2)[1 \quad 1 \quad 1 \quad 1 \quad -4][\mathbf{I}_G(x+1,y) \quad \mathbf{I}_G(x-1,y) \\
&\quad \mathbf{I}_G(x,y+1) \quad \mathbf{I}_G(x,y-1) \quad \mathbf{I}_G(x,y)]^T \\
&= \begin{bmatrix} \mathbf{H}'_x(x+1,y) - \mathbf{H}'_x(x,y) + \mathbf{H}'_y(x,y+1) - \mathbf{H}'_y(x,y) \\ +\lambda_1(\mathbf{C}'_x(x+1,y) - \mathbf{C}'_x(x,y) + \mathbf{C}'_y(x,y+1) - \mathbf{C}'_y(x,y)) \\ +\lambda_2(\mathbf{L}'_x(x+1,y) - \mathbf{L}'_x(x,y) + \mathbf{L}'_y(x,y+1) - \mathbf{L}'_y(x,y)) \end{bmatrix}.
\end{aligned}
$$

Considering an image with $W \times H$ pixels, for each pixel a similar equation can be constructed and indexed from 1 to $W \times H$. According to [24], by integrating these equations together, a sparse square matrix can be formed on the left, which is $\mathbf{A}$ in $\mathbf{Ax} = \mathbf{b}$.

Acknowledgments

The authors thank all six guest editors: Professors Qiang Ji, Jiebo Luo, Dimitris Metaxas, Antonio Torralba, Thomas Huang, and Erik Sudderth and all of the anonymous reviewers for their constructive comments on this manuscript. The author also thanks K. Rasche, A. Gooch, and Y. Zhang for their kind sharing of data and code. This project was partially supported by the the Natural Science Foundation of China (60873124), Nanyang Technological University SUG Grant (M58020010), the Natural Science Foundation of Zhejiang Provice (Y1090516), the Chinese Universities Scientific Fund (2009QNA5015), and the National Key Technology R&D Program (2008BAH26B02).

REFERENCES

[1] K. Rasche, R. Geist, and J. Westall, "Detail Preserving Reproduction of Colour Images for Monochromats and Dichromats," *IEEE Computer Graphics and Applications,* vol. 25, no. 3, pp. 22-30, May/June 2005.

[2] A. Gooch, J. Tumblin, and B. Gooch, "Color2gray: Salience-Preserving Colour Removal," *ACM Trans. Graphics,* vol. 24, no. 3, pp. 634-639, 2005.

[3] R. Brown, "Photoshop Tips: Converting Colour to Black-and-White," http://www.russellbrown.com/tips_tech.html, 2006.

[4] G. Wyszecki and W.S. Stiles, *Colour Science: Concepts and Methods, Quantitative Data and Formulae.* Wiley Interscience, 2000.

[5] H.R. Wu and K.R. Rao, *Digital Video Image Quality and Perceptual Coding.* CRC Press, 2001.

[6] R. Bala and K. Braun, "Color-to-Grayscale Conversion to Maintain Discriminabillity," *Proc. SPIE,* pp. 196-202, 2004.

[7] K. Rasche, R. Geist, and J. Westall, "Re-Colouring Images for Gamuts of Lower Dimension," *Computer Graphics Forum,* vol. 24, no. 3, pp. 423-432, 2005.

[8] I.T. Jolliffe, *Principle Component Analysis.* Springer, 2002.

[9] S. Mika, B. Schölkopf, A. Smola, K.-R. Müller, M. Scholz, and G. Rätsch, "Kernel PCA and De-Noising in Feature Spaces," *Advances in Neural Information Processing Systems,* vol. 11, pp. 536-542, MIT Press, 1999.

[10] K.I. Kim, M.O. Franz, and B. Scholkopf, "Iterative Kernel Principal Component Analysis for Image Modeling," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 27, no. 9, pp. 1351-1366, Sept. 2005.

[11] D. Socolinsky and L. Wolff, "Multispectral Image Visualization through First-Order Fusion," *IEEE Trans. Image Processing,* vol. 11, no. 8, pp. 923-931, Aug. 2002.

[12] I. Borg and P. Groenen, *Modern Multidimensional Scaling: Theory and Applications.* Springer-Verlag, 1997.

[13] E. Reinhard, M. Ashikhmin, B. Gooch, and P. Shirley, "Colour Transfer between Images," *IEEE Computer Graphics and Applications,* vol. 21, no. 5, pp. 34-41, Sept./Oct. 2001.

[14] Y.-W. Tai, J. Jia, and C.-K. Tang, "Soft Color Segmentation and Its Applications," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 29, no. 9, pp. 1520-1537, Sept. 2007.

[15] G.R. Greenfield and D.H. House, "Image Recolouring Induced by Palette Colour Associations," *J. Winter School of Computer Graphics and CAD Systems,* vol. 11, no. 1, pp. 3-7, 2003.

[16] D.L. Ruderman, T.W. Cronin, and C.-C. Chiao, "Statistics of Cone Responses to Natural Images: Implications for Visual Coding," *J. Optical Soc. of Am., A,* vol. 15, no. 8, pp. 2036-2045, 1998.

[17] D. Malacara, *Colour Vision and Colourimetry Theory and Applications.* SPIE Press, 2002.

[18] J. Vogel and B. Schiele, "Semantic Modeling of Natural Scenes for Content-Based Image Retrieval," *Int'l J. Computer Vision,* vol. 72, no. 2, pp. 133-157, 2007.

[19] C.E. Guo, S.C. Zhu, and Y.N. Wu, "Primal Sketch: Integrating Texture and Structure," *Computer Vision and Image Understanding,* vol. 106, no. 1, pp. 5-19, 2007.

[20] D. Slater and G. Healey, "The Illumination Invariant Matching of Deterministic Local Structure in Color Images," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 19, no. 10, pp. 1146-1151, Oct. 1997.

[21] M.A. Ruzon and C. Tomasi, "Edge, Junction, and Corner Detection Using Color Distributions," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 23, no. 11, pp. 1281-1295, Nov. 2001.

[22] D.R. Martin, C.C. Fowlkes, and J. Malik, "Learning to Detect Natural Image Boundaries Using Local Brightness, Color, and Texture Cues," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 26, no. 5, pp. 530-549, May 2004.

[23] R. Fattal, D. Lischinski, and M. Werman, "Gradient Domain High Dynamic Range Compression," *ACM Trans. Graphics,* vol. 21, no. 3, pp. 249-256, 2002.

[24] G. Sapiro, *Geometric Partial Differential Equation and Image Analysis.* Cambridge Univ. Press, 2001.

[25] T.A. Davis, "Umfpack Version 4.1 User Guide," Technical Report TR-03-008, Univ. of Florida, 2003.

[26] W.H. Press, S.A. Teukolsk, W.T. Vetterling, and B.P. Flannery, *Numerical Recipes in C: The Art of Scientific Computing,* second ed. Cambridge Univ. Press, 1992.

[27] J. Bolz, I. Farmer, E. Grinspun, and P. Schröder, "Sparse Matrix Solvers on the GPU: Conjugate Gradients and Multigrid," *ACM Trans. Graphics,* vol. 22, no. 3, pp. 917-924, 2003.

[28] H.A. David, *The Method of Paired Comparisons.* Oxford Univ. Press, 1988.

[29] K. Smith, P.-E. Landes, J. Thollot, and K. Myszkowski, "Apparent Grayscale: A Simple and Fast Conversion to Perceptually Accurate Images and Video," *Computer Graphics Forum,* vol. 27, no. 2, pp. 193-200, 2008.

[30] R. Mantiuk, K. Myszkowski, and H.-P. Seidel, "A Perceptual Framework for Contrast Processing of High Dynamic Range Images," *ACM Trans. Applied Perception,* vol. 3, no. 3, pp. 286-308, 2006.

[31] R. Bala and R. Eschbach, "Spatial Color-to-Grayscale Transformation Preserving Chrominance Edge Information," *Proc. IS&T/SID 12th Color Imaging Conf.,* pp. 82-86, 2004.

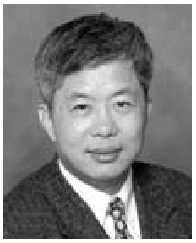
[32] A. Alsam and Ø. Kolås, "Grey Colour Sharpening," *Proc. IS&T/SID 14th Color Imaging Conf.,* pp. 263-267, 2006.

Mingli Song received the PhD degree in computer science from Zhejiang University, China, in 2006. He is currently an associate professor in the College of Computer Science and Microsoft Visual Perception Laboratory, Zhejiang University. His research interests include visual perception analysis, image enhancement, and face modeling. He is a member of the IEEE and the IEEE Computer Society.

Dacheng Tao received the BEng degree from the University of Science and Technology of China (USTC), the MPhil degree from the Chinese University of Hong Kong (CUHK), and the PhD degree from the University of London. Currently, he is a Nanyang assistant professor with the School of Computer Engineering at the Nanyang Technological University, a research associate professor at the University of London, a visiting professor at Xi'Dian University and a guest professor at Wuhan University. He mainly applyies statistics and mathematics for data analysis problems in cognitive science, computer vision, human computer interactions, multimedia, machine learning, and compressed sensing. He has authored or coauthors more than 150 scientific articles in top venues, including *IEEE TPAMI*, *TIP*, *TKDE*, NIPS, AISTATS, CVPR, ECCV, ICDM, *ACM TKDD*, KDD, and *Cognitive Computation*, with several best paper awards. His Erdös number is 3 and he holds the K.C. Wong Education foundation Award. He is an associate editor of the *IEEE Transactions on Knowledge and Data Engineering*, as well as of several Elsevier and Springer journals. He has served on committees for more than 100 major international conferences and more than 40 prestigious international journals. He is a member of the IEEE and the IEEE Computer Society.

Chun Chen is a professor in the College of Computer Science at Zhejiang University, China. His research interests include computer vision, computer graphics, and embedded technology.

Xuelong Li is a Resarcher (full professor) with the State Key Laboratory of Transient Optics and Photonics, Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, China. He is a senior member of the IEEE.



Chang Wen Chen received the BS degree from the University of Science and Technology of China in 1983, the MSEE degree from the University of Southern California in 1986, and the PhD degree from the University of Illinois at Urbana-Champaign in 1992. He has been a professor of computer science and engineering at the University at Buffalo, State University of New York, since 2008. Previously, he was Allen S. Henry distinguished professor in the Department of Electrical and Computer Engineering, Florida Institute of Technology, from 2003 to 2007. He was on the Faculty of Electrical and Computer Engineering at the University of Missouri-Columbia from 1996 to 2003 and at the University of Rochester, New York, from 1992 to 1996. From 2000 to 2002, he served as the head of the Interactive Media Group at the David Sarnoff Research Laboratories, Princeton, New Jersey. He has also consulted with Kodak Research Labs, Microsoft Research, Mitsubishi Electric Research Labs, NASA Goddard Space Flight Center, and the US Air Force Rome Laboratories. He has been the editor-in-chief for the *IEEE Transactions on Circuits and Systems for Video Technology* (*CSVT*) from January 2006 to December 2009. He has served as an editor for the *Proceedings of the IEEE*, the *IEEE Transactions on Multimedia*, the *IEEE Journal of Selected Areas in Communications*, *IEEE Multimedia*, the *Journal of Wireless Communication and Mobile Computing*, the *EUROSIP Journal of Signal Processing: Image Communications*, and the *Journal of Visual Communication and Image Representation*. He has also chaired and served on numerous technical program committees for the IEEE and other international conferences. He was elected a fellow of the IEEE for his contributions in digital image and video processing, analysis, and communications, and a fellow of the SPIE for his contributions in electronic imaging and visual communications.