Incorporating Higher-Order Cues in Image Colorization

Tae Hoon Kim
thkim@diehard.snu.ac.kr
Kyoung Mu Lee
kyoungmu@snu.ac.kr
Sang Uk Lee
sanguk@ipl.snu.ac.kr

Dept. of EECS, ASRI
Seoul National University
Seoul, Korea

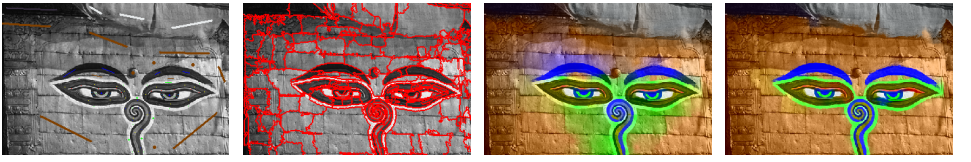
Abstract

We consider the colorization problem of grayscale images when some pixels, called scribbles, with initial colors are given. In this paper, we propose a new multi-layer graph model and an energy formulation that can incorporate higher-order cues for reliable colorization of natural images. In contrast to most existing energy functions with unary and pairwise constraints, we address the problem of imposing a high-order constraint whereby pixels constituting each region tend to have similar colors to the representative color of the region they belong to. The representative colors of the regions that are generated by unsupervised image segmentation algorithms, act as higher-order cues, and they are automatically obtained by a nonparametric learning technique that estimates them from the resulting pixel colors in a recursive fashion. We formulate this problem in terms of two quadratic energy functions of pixel and region colors, that are supplementary to each other, in our proposed multi-layer graph model and estimate them by a simple optimization technique that minimizes both functions simultaneously. Since our higher-order constraint enforces the color consistency among regions, we can easily obtain good colorization results with less dependence on the size and position of each user-given scribble. Experiments on several natural images demonstrate that our colorization method achieves much high-quality results compared with the conventional state-of-the-art methods.

1 Introduction

Colorization problem is to add colors to a grayscale image in which some pixels, called scribbles, with initial user-defined colors are given. There are two important difficulties for reliable colorizations in natural images. The first problem is how to correctly identify fuzzy or complex region boundaries with robust color distinction. The second one is how to easily add a user's scribbles. The main reason for why these problems are difficult to solve is that general colorizations tend to be largely dependent on the scribble properties: size and position. In accordance with these scribble properties, the resulted image can look oversmoothed and suffer from color artifacts visible especially near edges. In this case, the user is often left with the task of manually drawing the additional scribbles for delineating complicated

BMVC 2009 doi:10.5244/C.23.97



(a) (b) (c) (d)

Figure 1: Introducing our colorization framework. (a) Grayscale image with color scribbles. (b) Regions generated by an unsupervised image segmentation algorithm [3]. (Red curves represent the boundaries of the regions.) (c) Conventional colorization result by Levin et al. [10]. (d) Our higher-order colorization result by using regions in (b).

boundaries between a subject and the background. It is an expensive and time-consuming process. Therefore we need an efficient system which produces reasonable colorizations by using less human effort.

Recently, several colorization approaches [7][8][14][10][2][13][4] have been proposed. The work of Yatziv et al. [14] is based on the concept of the weighted color blending. This blending is derived from a weighted distance function computed from the intensity channel. By using the basic concept of the shortest distance as the blending weight, it offers a fast colorization technique. However, it did not consider the global color-continuity relationship between neighboring pixels. The method by Levin et al. [10] colorizes an image by minimizing a quadratic energy function derived from the color differences between a pixel and the weighted average of its neighborhood colors. However, enough scribbles that are tightly positioned along the object boundaries must be given for producing high-quality color images, especially for thin elongated objects. For solving this difficulty, it is helpful to use the region information obtained by over-segmentations. Previous region-based colorization methods [2][13][4] are inspired by the hard constraint whereby pixels constituting a particular region should have similar colors. If, however, the initial regions are not consistent with boundaries in the image, there are radical difficulties in obtaining exact solutions. In fact, such situations often arise in natural images.

Our technique automatically propagates the scribble colors to the remaining pixels by the grayscale intensity difference, as in [10]. However, unlike [10], for the color estimation of each pixel, our method additionally utilizes the representative colors of the over-segmented regions as higher-order cues in terms of a soft color consistency constraint whereby the pixels in each region tend to have similar colors to the representative color of the region they belong to. The region-based higher-order cues have been proposed for other tasks in computer vision such as labeling problems [6][12][9]. Especially, Kohli et al. [9] first learn higher-order cues as parametric models, and then new higher-order potentials in an energy function are designed based on them for the label estimation. In our work, we simultaneously estimate the region colors, namely higher-order cues, and the final pixel colors. Instead of parametric models, we address a nonparametric learning technique to recursively estimate the higher-order cues from the resulting pixel colors included in each region. By incorporating these higher-order cues into a quadratic energy function for partly enforcing the color consistency among regions, the final color image is also recursively produced. Therefore we design two quadratic energy functions of pixel and region colors that are supplementary to each other, and estimate them by a simple optimization technique that minimizes both functions simultaneously. Our colorization method has various advantages over conventional approaches

[14][10] as follows. First, it is less dependent on the size and position of each scribble. Even with a few scribbles that are roughly positioned, we can achieve good colorization results with clear object boundaries. For example, the gray-scale image that contains thin elongated objects is much better colorized with detailed boundaries than the conventional state-of-the-art method as shown in Fig. 1. This advantage can be attributed to the enforcement of the soft color consistency among regions. Second, the initial regions guide the suitable positions of the scribbles for a user. Therefore, the user can expect reasonable colorization according to the user-inputs. Finally, we can encode the long-range connections between regions that facilitate propagation of colors across larger image regions via a region-based quadratic energy function.

The paper is organized as follows. In Section 2, we introduce our proposed framework for colorization and explain in detail how to design the quadratic energy functions based on higher-order cues in a proposed graph model. The experimental results of our method are given in Section 3. Finally, we discuss our approach and give conclusions in Section 4.

2 Proposed Algorithm

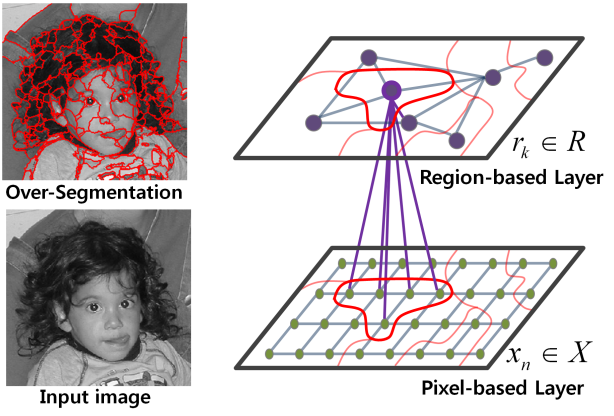
Given a grayscale image I with scribbles S with the desired colors, the colorization is to find the colors of all pixels $X = \{x_n\}_{n=1,\ldots,|X|}$. We work in the YUV color space where $Y = \{y_n\}_{n=1,\ldots,|X|}$ is the monochromatic luminance channel, which we will refer to simply as intensity, while $U = \{u_n\}_{n=1,\ldots,|X|}$ and $V = \{v_n\}_{n=1,\ldots,|X|}$ are the chrominance channels, encoding the color. Our goal is to complete both the U and V channels, given $Y = I$. We deal with the only U channel in this paper, since the V channel can be treated in the same manner.

Our colorization is based on the propagation of the scribble colors by the intensity differences. Unlike previous work [10] that generally consider only pairwise constraints that two neighboring pixels should have similar colors if their intensities are similar, we propose to impose a high-order constraint whereby pixels constituting each region tend to have similar colors to the representative color of the region they belong to. Our proposed algorithm works as follows. We first design a new multi-layer graph model with two different node types: pixels and regions in the image. We then formulate two quadratic energy functions for estimating the pixel and region colors, that are supplementary to each other, in this graph, and simultaneously optimize them in a simple way.

2.1 Graphical Model

Let us construct an undirected graph $G = (Q, E)$ where the nodes $Q = \{X, R\}$ consist of two types: pixels X and regions R, generated by an unsupervised segmentation algorithm such as Mean Shift [3], and the edges E are the links between two nodes as shown in Fig. 2. In a subset X^+ of X, each node $x_n \in X^+$ represents the pixel with user-given color $u_n^+ \in U^+$. In a similar way, we select a subset R^+ of R, where each node $r_k \in R^+$ is a region containing at least one pre-defined pixel ($\exists x_n \in X^+$ among $x_n \in r_k$), and it has the initial color $\bar{u}_k^+ \in U^+$, defined as the mean color of inner pre-defined pixels. We will refer to these subsets X^+ and R^+ simply as seeds. Each pixel $x_n \in X$ initially has an intensity $y_n \in Y$. For each region $r_k \in R$, we can generate its properties $\bar{y}_k$ as the mean intensity of the inner pixels $x_n \in r_k$: $\bar{y}_k = \frac{1}{|r_k|}\sum_{x_n \in r_k} y_n$, where $|r_k|$ is the number of the pixels inside the region r_k.

In this graph G, the pairwise edges $E = \{E^{\mathrm{X}}, E^{\mathrm{R}}, E^{\mathrm{H}}\}$ are determined by the neighborhood system. The weights $W = \{W^{\mathrm{X}}, W^{\mathrm{R}}, W^{\mathrm{H}}\}$ are assigned to E and classified by different



(a) (b)

Figure 2: Illustration of a proposed graph. A node $Q = \{X, R\}$ denotes a pixel $x_n \in X$ (green circle) or a region $r_k \in R$ (violet circle). The boundaries of the regions R, generated by over-segmentation [3], are drawn in red color overlaid on the image in (a). (b) shows an example of the edges between a region and its corresponding pixels with violet lines.

criteria according to the types of the two connected nodes. First, an edge $e^{\mathrm{X}}_{nm} \in E^{\mathrm{X}}$ between two neighboring pixels x_n and x_m has a weight $w^{x}_{nm} \in W^{\mathrm{X}}$ as a typical Gaussian weighting function given by

$$w^{\mathrm{X}}_{nm} = \begin{cases} exp\left(-\frac{(y_n - y_m)^2}{2\sigma_X^2}\right) & x_m \in \aleph_n \\ 0 & otherwise, \end{cases} \tag{1}$$

where $\aleph_n$ represents the local neighborhood pixels of x_n and σ_X is the variance of the total pixel intensities. This weight w^{X}_{nm} provides us with a numerical measure for the intensity similarity between two nodes x_n and x_m. And, for an edge $e^{\mathrm{R}}_{kl} \in E^{\mathrm{R}}$ between two adjacent regions r_k and r_l, a weight $w^{\mathrm{R}}_{kl} \in W^{\mathrm{R}}$ is similarly defined:

$$w^{\mathrm{R}}_{kl} = \begin{cases} exp\left(-\frac{(\bar{y_k} - \bar{y_l})^2}{2\sigma_R^2}\right) & r_l \in \bar{\aleph}_k \\ 0 & otherwise, \end{cases} \tag{2}$$

where $\bar{\aleph}_k$ represents the adjacent neighborhood regions of r_k and σ_R is the variance of the total region intensities. Finally, the edges E^{H} between pixels X and regions R are added using the fact that each pixel corresponds to only one region such as in Fig. 2(b). Therefore, an edge $e^{\mathrm{H}}_{nk} \in E^{\mathrm{H}}$ between a pixel x_n and a region r_k has a weight $w^{\mathrm{H}}_{nk} \in W^{\mathrm{H}}$ as follows:

$$w^{\mathrm{H}}_{nk} = \begin{cases} 1 & x_n \in r_k \\ 0 & otherwise. \end{cases} \tag{3}$$

By these connections, each region can get the supplementary information of the pixel colors. Simultaneously, it is possible to transfer the information of the region colors into the pixel one. Since each region color is estimated based on the colors of its inner pixels as well as its neighboring regions, it is called "higher-order cue".

2.2 Higher-Order Energy Model

The concept of our proposed framework is to estimate the pixel colors $U = \{u_n\}_{n=1,\dots,|X|}$ by defining the resulted region colors $\bar{U} = \{\bar{u}_k\}_{k=1,\dots,|R|}$ as the higher-order cues. By the definition of the relationship between all nodes $Q = \{X, R\}$ in the graph G, we propose a quadratic energy function J^{X} of pixel colors U as follows.

$$\begin{aligned} J^{\mathrm{X}} &= \mathscr{E}^{\mathrm{X}}_{pairwise} + \lambda \mathscr{E}^{\mathrm{X}}_{unary} + \tau \mathscr{E}^{\mathrm{X}}_{region} \\ &= \sum_{x_{n,m} \in X} \tilde{w}^{\mathrm{X}}_{nm} (u_n - u_m)^2 + \lambda \sum_{x_n \in X^+} (u_n - u^+_n)^2 + \tau \sum_{x_n \in X} (u_n - \sum_{r_k \in R} \tilde{w}^{\mathrm{XR}}_{nk} \bar{u}_k)^2, \end{aligned} \tag{4}$$

where $\tilde{w}^{\mathrm{X}}_{nm} = \frac{w^{\mathrm{X}}_{nm}}{\sum_{n'=1}^{|X|} w^{\mathrm{X}}_{nn'}}$ in (1) and $\tilde{w}^{\mathrm{XR}}_{nk} = \frac{w^{\mathrm{H}}_{nk}}{\sum_{k'=1}^{|R|} w^{\mathrm{H}}_{nk'}}$ in (3) are defined as the normalized weights. Two parameters λ and τ emphasize the energy term $\mathscr{E}^{\mathrm{X}}_{unary}$ and $\mathscr{E}^{\mathrm{X}}_{region}$, respectively, and they are initially fixed for our all experiments.

The common energy model [10] used for colorization is characterized by the energy function only defined in unary $\mathscr{E}^{\mathrm{X}}_{unary}$ and pairwise $\mathscr{E}^{\mathrm{X}}_{pairwise}$ terms. Note that in this work, we propose to use the additional higher-order term $\mathscr{E}^{\mathrm{X}}_{region}$ defined by the color differences between pixels and regions.

Pairwise Term. The first term $\mathscr{E}^{\mathrm{X}}_{pairwise}$ of the right-hand side in (4) is the color-continuity constraint that two neighboring pixels in the small neighborhood system, which is usually chosen to be either 4 or 8 neighborhoods, should have similar colors if their intensities are similar. This pairwise term is closely related to the algorithms proposed for other tasks in image processing such as image segmentation algorithms [1][5]. In this work, we use the normalized weight function $\tilde{w}^{\mathrm{X}}_{nm}$ that sums to one. It becomes large when an intensity y_n is similar to y_m, and small when the two intensities are different, as in [10].

Unary Term. The second term $\mathscr{E}^{\mathrm{X}}_{unary}$ in (4) is the unary constraint that each seed should maintain the user-given color. For the larger λ, each seed $x_n \in X^+$ gets more similar color to the user-given one $u^+_n \in U^+$. Therefore λ represents the authenticity of the user-inputs.

Higher-Order Term. Finally, the third term $\mathscr{E}^{\mathrm{X}}_{region}$ in (4) is the higher-order color consistency constraint whereby a pixel color should be similar to its corresponding region color. Unlike the hard color consistency constraints that were used in other conventional region-based methods, our soft constraint allows the pixels in a common region to have quite different colors. Since a region often contains multiple seed pixels with different colors, the color consistency between its inner pixels is partly enforced with a weight τ. Also, when some regions have the inconsistent boundaries under high-clutter scenes, this soft constraint is truly useful. This higher-order term $\mathscr{E}^{\mathrm{X}}_{region}$ is designed based on the region colors $\bar{U}$, estimated by nonparametric learning from the given image itself.

2.3 Learning Region Colors

To solve the formulation (4), we need to estimate the region colors $\bar{U}$ by referring the pixel colors U in the graph G. In a similar way to J^{X} in (4), the energy model J^{R} for the region colors $\bar{U}$ is formulated as follows.

$$\begin{aligned} J^{\mathrm{R}} &= \mathscr{E}^{\mathrm{R}}_{pairwise} + \lambda \mathscr{E}^{\mathrm{R}}_{unary} + \varepsilon \mathscr{E}^{\mathrm{R}}_{pixel} \\ &= \sum_{r_{k,l} \in R} \tilde{w}^{\mathrm{R}}_{kl} (\bar{u}_k - \bar{u}_l)^2 + \lambda \sum_{r_k \in R^+} (\bar{u}_k - \bar{u}^+_k)^2 + \varepsilon \sum_{r_k \in R} (\bar{u}_k - \sum_{x_n \in X} \tilde{w}^{\mathrm{RX}}_{kn} u_n)^2, \end{aligned} \tag{5}$$

where $\tilde{w}^{\mathrm{R}}_{kl} = \frac{w^{\mathrm{R}}_{kl}}{\sum_{k'=1}^{|R|} w^{\mathrm{R}}_{kk'}}$ in (2) and $\tilde{w}^{\mathrm{RX}}_{kn} = \frac{w^{\mathrm{H}}_{nk}}{\sum_{n'=1}^{|X|} w^{\mathrm{H}}_{n'k}}$ in (3) are the normalized weights. Two parameters λ and ε emphasize the energy term $\mathscr{E}^{\mathrm{R}}_{unary}$ and $\mathscr{E}^{\mathrm{R}}_{pixel}$, respectively, and they are fixed for our all experiments.

The first term $\mathscr{E}^{\mathrm{R}}_{pairwise}$ of the right-hand side in (5) is the pairwise constraint that both adjacent regions should have similar colors if their mean intensities are similar, similarly to $\mathscr{E}^{\mathrm{X}}_{pairwise}$ in (4). Namely, each region color $\bar{u}_k$ depends on the basic constraint that a good color image should not change too much between nearby regions. The second term $\mathscr{E}^{\mathrm{R}}_{unary}$ in (5) is the unary constraint that each seed region $r_k \in R^+$ should have the mean color $\bar{u}^+_k \in \bar{U}^+$ of the inner seed pixels, similarly to $\mathscr{E}^{\mathrm{X}}_{unary}$ in (4). Finally, the third term $\mathscr{E}^{\mathrm{R}}_{pixel}$ in (5) is another estimated unary constraint whereby a region color $\bar{u}_k$ should be similar to the mean of inner pixel colors u_n of $x_n \in r_k$. This term has the effect of refining the region colors from more informative pixel colors, when there is less color information from user-inputs.

2.4 Optimization Formulation

Since two energy functions J^{X} and J^{R} in (4) and (5) are related to each other, we should minimize them simultaneously. First, these energy functions can be reformulated in matrix forms with two vectors $\vec{u} = [u_n]_{|X|\times 1}$ and $\vec{\bar{u}} = [\bar{u}_k]_{|R|\times 1}$ as follows.

$$\begin{aligned} \mathbf{J}^{\mathrm{X}} &= \vec{u}^T(\mathbf{I}-\tilde{\mathbf{W}}^{\mathrm{X}})\vec{u} + (\vec{u}-\vec{u}^*)^T\Lambda^{\mathrm{X}}(\vec{u}-\vec{u}^*) + \tau(\vec{u}-\tilde{\mathbf{W}}^{\mathrm{XR}}\vec{\bar{u}})^T(\vec{u}-\tilde{\mathbf{W}}^{\mathrm{XR}}\vec{\bar{u}}) \\ \mathbf{J}^{\mathrm{R}} &= \vec{\bar{u}}^T(\mathbf{I}-\tilde{\mathbf{W}}^{\mathrm{R}})\vec{\bar{u}} + (\vec{\bar{u}}-\vec{\bar{u}}^*)^T\Lambda^{\mathrm{R}}(\vec{\bar{u}}-\vec{\bar{u}}^*) + \varepsilon(\vec{\bar{u}}-\tilde{\mathbf{W}}^{\mathrm{RX}}\vec{u})^T(\vec{\bar{u}}-\tilde{\mathbf{W}}^{\mathrm{RX}}\vec{u}), \end{aligned} \tag{6}$$

where $\tilde{\mathbf{W}}^{\mathrm{X}} = [\tilde{w}^{\mathrm{X}}_{nm}]_{|X|\times|X|}$, $\tilde{\mathbf{W}}^{\mathrm{XR}} = [\tilde{w}^{\mathrm{XR}}_{nk}]_{|X|\times|R|}$, $\tilde{\mathbf{W}}^{\mathrm{R}} = [\tilde{w}^{\mathrm{R}}_{kl}]_{|R|\times|R|}$, and $\tilde{\mathbf{W}}^{\mathrm{RX}} = [\tilde{w}^{\mathrm{RX}}_{kn}]_{|R|\times|X|}$ are the row-normalized matrices with weights in (4) and (5). The matrix $\Lambda^{\mathrm{X}} = diag([\kappa^{\mathrm{X}}_1,...,\kappa^{\mathrm{X}}_{|X|}])$ (or $\Lambda^{\mathrm{R}} = diag([\kappa^{\mathrm{R}}_1,...,\kappa^{\mathrm{R}}_{|R|}])$) is a diagonal form with $\kappa^{\mathrm{X}}_n = \lambda$ (or $\kappa^{\mathrm{R}}_k = \lambda$) if a seed $x_n \in X^+$ (or $r_k \in R^+$) and 0 otherwise. The vector $\vec{u}^* = [u^*_n]_{|X|\times 1}$ (or $\vec{\bar{u}}^* = [\bar{u}^*_k]_{|R|\times 1}$) is an initial color vector with $u^*_n = u^+_n$ (or $\bar{u}^*_k = \bar{u}^+_k$) if $x_n \in X^+$ (or $r_k \in R^+$) and 0 otherwise.

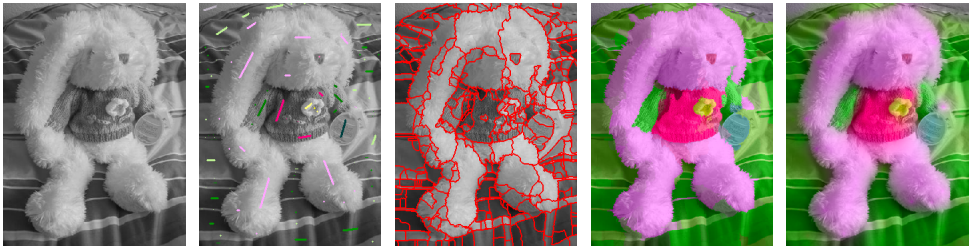
Since these two quadratic functions are convex, they can be solved jointly in a simple way. Differentiating their matrix formulations $\mathbf{J}^{\mathrm{X}}$ and $\mathbf{J}^{\mathrm{R}}$ in (6) with respect to $\vec{u}$ and $\vec{\bar{u}}$, respectively, and set to zero, we can get all colors $\vec{u}$ and $\vec{\bar{u}}$ simply by

$$\begin{aligned} \frac{\partial \mathbf{J}^{\mathrm{X}}}{\partial \vec{u}} &= \vec{u}-\tilde{\mathbf{W}}^{\mathrm{X}}\vec{u} + \Lambda^{\mathrm{X}}(\vec{u}-\vec{u}^*) + \tau(\vec{u}-\tilde{\mathbf{W}}^{\mathrm{XR}}\vec{\bar{u}}) &= 0 \\ \frac{\partial \mathbf{J}^{\mathrm{R}}}{\partial \vec{\bar{u}}} &= \vec{\bar{u}}-\tilde{\mathbf{W}}^{\mathrm{R}}\vec{\bar{u}} + \Lambda^{\mathrm{R}}(\vec{\bar{u}}-\vec{\bar{u}}^*) + \varepsilon(\vec{\bar{u}}-\tilde{\mathbf{W}}^{\mathrm{RX}}\vec{u}) &= 0, \end{aligned} \tag{7}$$

which can be jointly transformed into

$$\begin{bmatrix} \vec{u} \\ \vec{\bar{u}} \end{bmatrix} = (\mathbf{I}-\Omega)\Pi \begin{bmatrix} \vec{u} \\ \vec{\bar{u}} \end{bmatrix} + \Omega \begin{bmatrix} \vec{u}^* \\ \vec{\bar{u}}^* \end{bmatrix}, \tag{8}$$

where $\Omega = \begin{bmatrix} \frac{\Lambda^{\mathrm{X}}}{(1+\tau)\mathbf{I}+\Lambda^{\mathrm{X}}} & \\ & \frac{\Lambda^{\mathrm{R}}}{(1+\varepsilon)\mathbf{I}+\Lambda^{\mathrm{R}}} \end{bmatrix}$ and $\Pi = \begin{bmatrix} \dot{\tau}\tilde{\mathbf{W}}^{\mathrm{X}} & (1-\dot{\tau})\tilde{\mathbf{W}}^{\mathrm{XR}} \\ (1-\dot{\varepsilon})\tilde{\mathbf{W}}^{\mathrm{RX}} & \dot{\varepsilon}\tilde{\mathbf{W}}^{\mathrm{R}} \end{bmatrix}$ ($\dot{\tau} = \frac{1}{1+\tau}$ and $\dot{\varepsilon} = \frac{1}{1+\varepsilon}$) are the diagonal and row-normalized matrices, respectively. Since the matrix ($\mathbf{B} = \mathbf{I} - (\mathbf{I}-\Omega)\Pi$) is nonsingular [11], the matrix $\mathbf{B}$ is invertible. Therefore we easily



(a) (b) (c) (d) (e)

Figure 3: Overview of our proposed colorization algorithm. (a) Input image I. (b) Color scribbles S. (c) Regions: R, generated by Mean Shift [3]. (d) Our region-based colorization from region colors $\vec{u}$ in (9). (e) Our final colorization from pixel colors $\vec{u}$ in (9).

compute the equation (8) by a matrix inversion technique, and finally have the colors $\vec{u}$ and $\vec{u}$ of all pixels X and regions R for the colorization problem as follows:

$$\begin{bmatrix} \vec{u} \\ \vec{u} \end{bmatrix} = (\mathbf{I} - (\mathbf{I} - \Omega)\Pi)^{-1}\Omega \begin{bmatrix} \vec{u}^* \\ \vec{u}^* \end{bmatrix} = \mathbf{B}^{-1}\Omega \begin{bmatrix} \vec{u}^* \\ \vec{u}^* \end{bmatrix}. \tag{9}$$

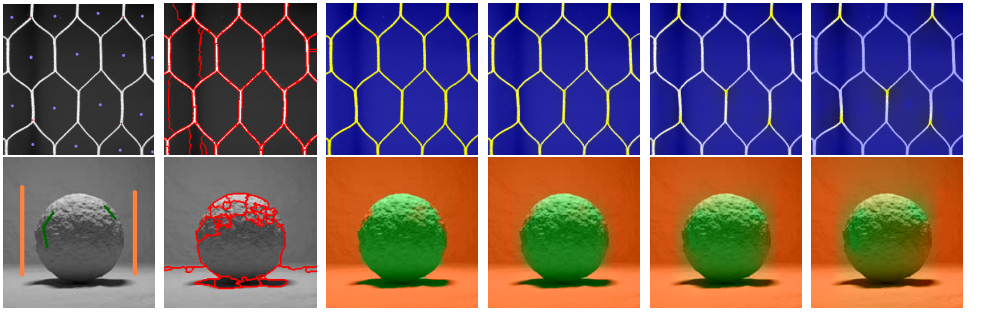
Since the edges E^{X} between two neighboring pixels are connected in a small neighborhood system, $\mathbf{B}$ in (9) is very large sparse $((|X|+|R|)\times(|X|+|R|))$ matrix $(|R| \ll |X|)$. Therefore its inversion typically has an efficient computation. The inversion method implemented by MATLAB division operator '\' (which we used in our experiments) is very efficient in finding the inversion of large sparse matrix. In case of the 364×273 image I in Fig. 3(a), $\mathbf{B}$ is on about 0.00004% full. It takes about 4.7 seconds to produce our colorization result in Fig. 3(e) by MATLAB 7.6 on a quad-core 2.4GHz desktop PC.

2.5 Overview of Our Colorization

Fig. 3 shows the overall process of our algorithm from the color scribbles to the estimation of the region colors, that are used as higher-order cues, and the resulting colorization. Our method starts with color scribbles S as shown in Fig. 3(b). After generating the regions R in Fig. 3(c) from the input image I in Fig. 3(a), we simultaneously estimate the colors $\vec{u}$ and $\vec{u}$ of all pixels X and regions R by solving the unified function (9). Fig. 3(d) presents the region-based colorization generated from $\vec{u}$ by exploiting the simple soft constraint whereby pixels in common region r_k have the same representative region color $\bar{u}_k$. We finally produce the colorization result from the estimated pixel colors $\vec{u}$ in Fig. 3(e). Our algorithm is based on the region information by the definition of the higher-order energy function in (4). Therefore the qualities of the regions exert influence on our colorization results. Note that even if some regions are incorrectly extracted for the higher-order cues as shown in Fig. 3(c), our method can infer highly accurate and natural colored object with precise boundaries as shown in Fig. 3(e).

3 Experiments

We evaluate our colorization method using several natural images. In our experiments, we initially obtain the over-segmented regions by Mean Shift segmentation algorithm [3]. The



(a) (b) (c) τ=0.1 (d) τ=0.01 (e) τ=0.001 (f) τ=0.0001

Figure 4: Examples of the colorizations with respect to the variation of τ in (4). From the images (a) with color scribbles, (c),(d),(e) and (f) give the resulting colorizations based on the regions (b) according to τ. ($\varepsilon = 0.1$)

Mean Shift algorithm uses two bandwidth parameters (h_s, h_r) for spatial and range domains, respectively, and we set $(h_s, h_r) = (7,7)$ in our tests. And, the 4-connected neighborhoods were used for the pixel neighborhood system in (1). The three parameters λ, τ and ε in (4) and (5) are fixed in all experiments, and chosen empirically. In this section, we first analyze the effect of parameters, and then, the performance of our algorithm is compared with the state-of-the-art methods [14][10].

3.1 Parameter Settings

In our experiments, we set λ in (4) and (5) to have a high value ($= 10^3$), since the estimated color of each seed should be similar to the initial one determined by the user. And, note that ε means the weight of the additional unary term estimated from the resulting pixel colors in (5). So, if there is enough color information from color scribbles, the resulting colorizations are less sensitive to ε, compared with τ in (4). Also, since the pixel and region colors are recursively estimated in this unified framework, two parameters τ and ε are mutually correlative. Therefore we select the only experiment with respect to the variation of τ, when ε is fixed. Fig. 4 shows the colorization results by varying the parameter τ. With larger τ, the color consistency inside each region is more emphasized and the object boundaries in the final colorizations are more consistent with the initial regions, as shown in Fig. 4. And with smaller τ, smoother colors are expressed around the seed pixels. It is important to find appropriate τ to reduce the dependence on the edges of inaccurate regions and to alleviate the over-smoothing effect. In this work, τ and ε were empirically chosen, and we set $\tau = 0.01$ and $\varepsilon = 0.1$ for all the test images.

3.2 Colorization Results

Now, let us examine the performance of our algorithm for the colorizations of natural images. We first examine the robustness of our proposed algorithm with respect to the the scribble properties: size and position. Fig. 5 shows an example of the colorization results with different sets of scribbles. In this experiment, we evaluate the robustness by the visual consistency of colorizations under different scribble settings as shown in Fig. 5(a). In Fig. 5(b), we observe that the shortest-path-based work of Yatziv et al. [14] is largely sensitive to

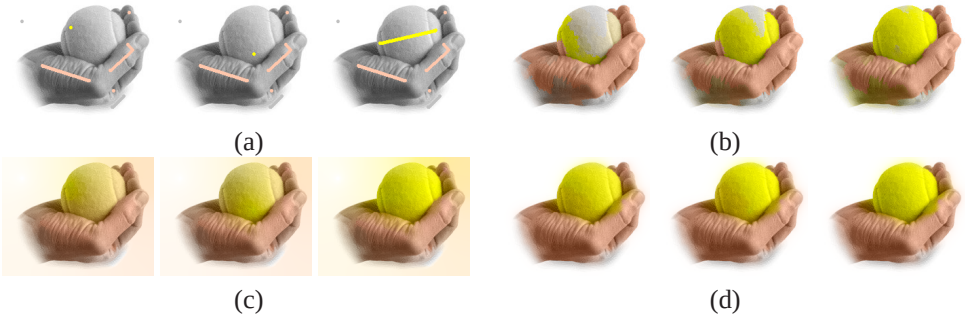


Figure 5: Testing the robustness of the higher-order colorization algorithm, compared with the state-of-the-art methods [14][10]. From different sets (a) of scribbles, (b),(c) and (d) are the colorized results by Yatziv et al. [14], Levin et al. [10] and our algorithm, respectively.

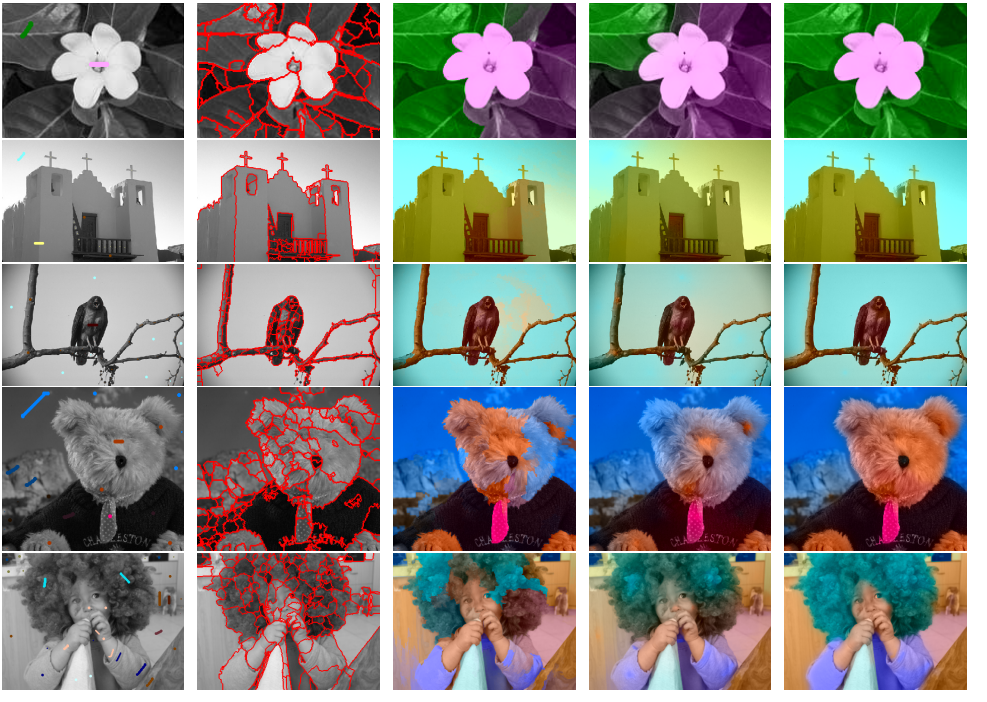
the size and position of each scribble, especially in the textured regions. Fig. 5(c) shows that the work of Levin et al. [10] produces the over-smoothed colorizations with color artifacts that are clearly visible especially near edges away from the scribbles. Since it deals with the only pairwise color consistency between the neighboring pixels, enough scribbles that are tightly positioned along the boundaries are needed. Compared with these two conventional methods, our method obtains visually more consistent and higher-quality results as shown in Fig. 5(d). Since our algorithm considers the global region information as well as the local relationship between the neighboring pixels, it is less dependent on the scribble changes.

Fig. 6 shows our final colorization results on several natural images, compared with those of the state-of-the-art methods [14] and [10]. In Fig. 6(c) and (d), the previous methods usually require for the users to delineate complicated boundaries between regions, since they are sensitive to the scribble size and position. In contrast, our approach provides superior performance with much higher-quality colorization results from even less number of scribbles (or few pixels) as shown in Fig. 6(e). These comparisons clearly demonstrate the robustness and accuracy of our algorithm.

4 Conclusions

In this paper, we propose a novel higher-order colorization framework with soft consistency constraint whereby the pixels in each region, generated by an unsupervised image segmentation algorithm, tend to have similar colors to the representative color of the region they belong to. Our work has several advantages. First, we automatically and recursively learn the region colors, defined as higher-order cues, from the resulting pixel colors. Simultaneously, we incorporate these high-order cues into the quadratic energy function for pixel colors, and minimize it in a simple way. Second, our method produces high-quality colorization results in natural images. Our experiments demonstrate that our higher-order cues significantly improve the colorization results. Finally, since our method is less sensitive to the size and position of each user-given scribble, it provides more convenient interaction for a user.

For the computation of two quadratic energy functions (4) and (5), two parameters τ and ε were chosen empirically. They are not, however, optimal for every image. Moreover, since some wrong regions can be usually extracted by the unsupervised segmentations, we can



(a) (b) (c) (d) (e)

Figure 6: Colorization results on the natural images. (a) Input images with color scribbles. (b) Regions generated by an unsupervised image segmentation algorithm [3]. Colorizations by (c) Yatziv et al. [14]; (d) Levin et al. [10]; (e) Our algorithm.

not guarantee that our method always produces the high-quality colorizations. If we use the well-controlled parameters τ and ε, and other methodologies for generating the regions with consistent boundaries, better results will be obtained. Therefore our future work will include the automatic parameter selection and the fusion of other region-based techniques.

Acknowledgement

This research was supported in part by the Defense Acquisition Program Administration and Agency for Defense Development, Korea, through IIRC (UD070007AD), and in part by the IT R&D program of MKE/IITA (2008-F-030-01, Development of Full 3D Reconstruction Technology for Broadcasting Communication Fusion).

References

[1] Y. Boykov and G. Funka-Lea. Graph cuts and efficient n-d image segmentation. *Int. J. Computer Vision*, 70(2):109–131, 2006.

[2] T. Chen, Y. Wang, V. Schillings, and C. Meinel. Grayscale image matting and colorization. In *Asian Conference on Computer Vision*, 2004.

[3] D. Comaniciu and P. Meer. Mean shift: a robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):603–619, 2002.

[4] O. Dalmau-Cedeno, M. Rivera, and P.P. Mayorga. Computing the α-channel with probabilistic segmentation for image colorization. In *International Conference on Computer Vision*, 2007.

[5] L. Grady. Random walks for image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(11):1768–1783, 2006.

[6] D. Hoiem, A.A. Efros, and M. Hebert. Geometric context from a single image. In *International Conference on Computer Vision*, 2005.

[7] T. Horiuchi. Estimation of color for gray-level image by probabilistic relaxation. In *International Conference on Pattern Recognition*, 2002.

[8] T. Horiuchi and S. Hirano. Colorization algorithm for grayscale image by propagating seed pixels. In *International Conference on Image Processing*, 2003.

[9] P. Kohli, L. Ladický, and P.H.S. Torr. Robust higher order potentials for enforcing label consistency. *Int. J. Computer Vision*, 82(3):302–324, 2009.

[10] A. Levin, D. Lischinski, and Y. Weiss. Colorization using optimization. *ACM Transactions on Graphics*, 23(3):689–694, 2004.

[11] C.D. Meyer. *Matrix Analysis and Applied Linear Algebra*. SIAM, ISBN: 0-89871-454-0, 2000.

[12] B.C. Russell, A.A. Efros, J. Sivic, W.T. Freeman, and A. Zisserman. Using multiple segmentations to discover objects and their extent in image collections. In *Computer Vision and Pattern Recognition*, 2006.

[13] D. Sýkora, J. Buriánek, and J. Žára. Unsupervised colorization of black-and-white cartoons. In *Non-Photorealistic Animation and Rendering*, 2004.

[14] L. Yatziv and G. Sapiro. Fast image and video colorization using chrominance blending. *IEEE Transactions on Image Processing*, 15(5):1120–1129, 2006.