Colorization in YCbCr Color Space and Its Application to JPEG Images

Hideki Noda [a,*], Michihiru Niimi [a]

[a] *Department of Systems Innovation and Informatics, Kyushu Institute of Technology, 680-4 Kawazu, Iizuka, 820-8502 Japan*

Abstract

This paper presents a colorization method in YCbCr color space, which is based on the maximum a posteriori estimation of a color image given a monochrome image as is our previous method in RGB color space. The presented method in YCbCr space is much simpler than that in RGB space and requires much less computation time, while both methods in YCbCr and RGB space produce color images with comparable PSNR values. The proposed colorization in YCbCr is applied to JPEG compressed color images aiming at better recovery of downsampled chrominance planes. Experimental results show that colorization in YCbCr is usually effective for quality improvement of JPEG color images.

Key words: Colorization, Monochrome image, MAP estimation, MRF, YCbCr, JPEG

---

1 Introduction

Colorization is usually a computer-aided process of adding color to monochrome images or movies. Colorization is now generally carried out manually using some drawing software tools. Obviously such manual work is very expensive and time-consuming.

Several colorization methods [1–3] have already been proposed which do not require intensive manual effort. Welsh et al. proposed a semi-automatic method to colorize a monochrome image by transferring color from a reference color image [1]. This method requires an appropriate reference color image prepared

---

* Corresponding author. Tel.: +81-948-29-7722; fax: +81-948-29-7709
Email address: `noda@mip.ces.kyutech.ac.jp` (Hideki Noda).

Preprint submitted to Elsevier Science 2 April 2007

by a user. Levin et al. have proposed an interactive method, where a user needs to give some color scribbles and the colors are automatically propagated to produce a fully colorized image [2]. Horiuchi [3] has proposed a method where a user gives colors for some pixels and colors for all other pixels are determined automatically by using the probabilistic relaxation [4].

We have also proposed a colorization method in red, green, blue (RGB) color space [5], where unlike previously proposed methods, the colorization problem is formulated as the maximum a posteriori (MAP) estimation of a color image given a monochrome image. Markov random field (MRF) [6] is used for modeling a color image which is utilized as a prior for the MAP estimation. In this paper, we consider colorization in luminance and chrominance (YCbCr) color space under the same formulation as in RGB space and derive a simpler and more efficient algorithm than that in RGB space. This is in principle due to the fact that in YCbCr space luminance component is already known from a given monochrome image and only the other two components have to be estimated.

Then we give a meaningful application of the proposed colorization in YCbCr space, i.e., its application to JPEG [7] compressed color images. JPEG is a commonly used standard to compress digital color images. In JPEG, Cb and Cr planes are usually downsampled by a factor of two at its compression stage, and afterward the downsampled chrominance planes are interpolated at its decompression stage. Aiming at better recover of the downsampled chrominance planes, the proposed colorization in YCbCr space is applied to JPEG color images. The proposed colorization algorithm has a structure that chrominance components are estimated considering luminance component which is not downsampled, and therefore we can expect a better recovery of them.

The rest of this paper is organized as follows. After color image estimation in RGB space is reviewed in Section 2, that in YCbCr space is described in Section 3. In Section 4, experimental results are given to compare colorization methods in RGB and YCbCr space. In Section 5, application of colorization in YCbCr to JPEG images is described and conclusions are addressed in Section 6.

2 Color Image Estimation in RGB Space

In this section, we review our previous colorization method in RGB space [5].

2.1 Estimation Algorithm

Let $\mathbf{x}_L = \{\mathbf{x}_{ij}; (i,j) \in L\}$ and $y_L = \{y_{ij}; (i,j) \in L\}$ [1] denote a color image and a monochrome image, respectively, defined on a two-dimensional lattice $L = \{(i,j); 1 \leq i \leq N_1, 1 \leq j \leq N_2\}$. In RGB color space, $\mathbf{x}_{ij} = (r_{ij}, g_{ij}, b_{ij})^T$, i.e., a color vector at (i,j) pixel is composed of red r_{ij}, green g_{ij} and blue b_{ij} components. We assume that a monochrome image $y_L = \{y_{ij}; (i,j) \in L\}$ is associated with a color image $\mathbf{x}_L = \{\mathbf{x}_{ij}; (i,j) \in L\}$ under the following relation:

$$\begin{aligned} y_{ij} = \mathbf{a}^T \mathbf{x}_{ij} &= 0.299 r_{ij} + 0.587 g_{ij} + 0.114 b_{ij}, \\ & 0 \leq y_{ij}, r_{ij}, g_{ij}, b_{ij} \leq 255. \end{aligned} \tag{1}$$

Given y_L, $\mathbf{x}_L$ can be estimated by maximizing the a posteriori probability $p(\mathbf{x}_L \mid y_L)$, i.e., by MAP estimation. The MAP estimate $\hat{\mathbf{x}}_L$ is written as

$$\hat{\mathbf{x}}_L = \arg\max_{\mathbf{x}_L} p(\mathbf{x}_L \mid y_L), \tag{2}$$

where the a posteriori probability $p(\mathbf{x}_L \mid y_L)$ is described as

$$p(\mathbf{x}_L \mid y_L) = \frac{p(y_L \mid \mathbf{x}_L) p(\mathbf{x}_L)}{\sum_{\mathbf{x}_L} p(y_L \mid \mathbf{x}_L) p(\mathbf{x}_L)}. \tag{3}$$

Considering (1), $p(y_L \mid \mathbf{x}_L)$ is described as

$$\begin{aligned} p(y_L \mid \mathbf{x}_L) &= 1(\{y_{ij} = \mathbf{a}^T \mathbf{x}_{ij}, (i,j) \in L\}) \\ &= \prod_{(i,j) \in L} 1(y_{ij} = \mathbf{a}^T \mathbf{x}_{ij}), \end{aligned} \tag{4}$$

where

$$1(y_{ij} = \mathbf{a}^T \mathbf{x}_{ij}) = \begin{cases} 1 & \text{if } y_{ij} = \mathbf{a}^T \mathbf{x}_{ij} \\ 0 & \text{otherwise.} \end{cases} \tag{5}$$

Assuming a Markov random field (MRF) for $\mathbf{x}_L$ and then using the mean field approximation, $p(\mathbf{x}_L)$ can be decomposed as

---

[1] In this paper, x_A and $f(x_A)$ denote the set $\{x_{a_1}, \ldots, x_{a_l}\}$ and the multivariable function $f(x_{a_1}, \ldots, x_{a_l})$ respectively, where $A = \{a_1, \ldots, a_l\}$.

$$p(\mathbf{x}_L) \simeq \prod_{(i,j)\in L} p(\mathbf{x}_{ij} \mid \langle \mathbf{x} \rangle_{\eta_{ij}}), \tag{6}$$

where η_{ij} denotes (i,j) pixel's neighborhood and $\langle \mathbf{x} \rangle_{\eta_{ij}}$ denotes the mean fields for $\mathbf{x}_{\eta_{ij}}$. Substituting (4) and (6) into (3) and replacing $\sum_{\mathbf{x}_L} \prod_{(i,j)\in L}$ by $\prod_{(i,j)\in L} \sum_{\mathbf{x}_{ij}}$, we obtain the following decomposition for $p(\mathbf{x}_L \mid y_L)$:

$$p(\mathbf{x}_L \mid y_L) \simeq \prod_{(i,j)\in L} p(\mathbf{x}_{ij} \mid y_{ij}, \langle \mathbf{x} \rangle_{\eta_{ij}}), \tag{7}$$

where

$$p(\mathbf{x}_{ij} \,|\, y_{ij}, \langle \mathbf{x} \rangle_{\eta_{ij}}) = \frac{1(y_{ij} = \mathbf{a}^T \mathbf{x}_{ij}) p(\mathbf{x}_{ij} \mid \langle \mathbf{x} \rangle_{\eta_{ij}})}{\sum_{\mathbf{x}_{ij}} 1(y_{ij} = \mathbf{a}^T \mathbf{x}_{ij}) p(\mathbf{x}_{ij} \mid \langle \mathbf{x} \rangle_{\eta_{ij}})}. \tag{8}$$

In the following, $\mathbf{x}_{\eta_{ij}}$ is simply used for $\langle \mathbf{x} \rangle_{\eta_{ij}}$. Then $p(\mathbf{x}_{ij} \mid y_{ij}, \mathbf{x}_{\eta_{ij}}) = p(\mathbf{x}_{ij} \mid y_{ij}, \langle \mathbf{x} \rangle_{\eta_{ij}})$ is considered as local a posteriori probability (LAP). Using these LAPs, the global optimization problem shown by Eq. (2) is approximately decomposed into the local optimization problems

$$\hat{\mathbf{x}}_{ij} = \arg\max_{\mathbf{x}_{ij}} p(\mathbf{x}_{ij} \mid y_{ij}, \mathbf{x}_{\eta_{ij}}). \tag{9}$$

In order to solve (9) for all (i,j) pixels, their neighboring color vectors $\mathbf{x}_{\eta_{ij}}$ should be given. Since such a problem as shown in (9) can be solved iteratively as is popular in numerical analysis, we rewrite Eq. (9) as

$$\mathbf{x}_{ij}^{(p+1)} = \arg\max_{\mathbf{x}_{ij}} p(\mathbf{x}_{ij} \mid y_{ij}, \mathbf{x}_{\eta_{ij}}^{(p)}), \tag{10}$$

where p represents the pth iteration.

Regarding $p(\mathbf{x}_{ij} \mid \mathbf{x}_{\eta_{ij}})$ in (8), a Gaussian MRF is here used whose local conditional probability density function (pdf) is given as

$$p(\mathbf{x}_{ij} \mid \mathbf{x}_{\eta_{ij}}) = \frac{1}{(2\pi)^{3/2} |\mathbf{\Sigma}|^{1/2}} \exp\{-\frac{1}{2}(\mathbf{x}_{ij} - \bar{\mathbf{x}}_{\eta_{ij}})^T \mathbf{\Sigma}^{-1} (\mathbf{x}_{ij} - \bar{\mathbf{x}}_{\eta_{ij}})\}, \tag{11}$$

$$\bar{\mathbf{x}}_{\eta_{ij}} = \frac{1}{|N|} \sum_{\tau \in N} \mathbf{x}_{ij+\tau}. \tag{12}$$

Here $\bar{\mathbf{x}}_{\eta_{ij}}$ is the mean of neighboring pixels' color vectors $\mathbf{x}_{\eta_{ij}} = \{\mathbf{x}_{ij+\tau}, \tau \in N\}$, where N denotes the neighborhood of $(0,0)$ pixel. For example, $N = \{(0,1), (0,-1), (1,0), (-1,0)\}$ for the first-order neighborhood, and if $\tau =$

$(0,1)$, $\mathbf{x}_{ij+\tau} = \mathbf{x}_{i,j+1}$. $\mathbf{\Sigma}$ is the covariance matrix of $\mathbf{x}_{ij} - \bar{\mathbf{x}}_{\eta_{ij}}$. Considering (1), (8), (11) and (12), the local MAP estimation (10) is rewritten as the following constrained quadratic programming problem:

$$\begin{aligned} &\text{minimize} \ \ (\mathbf{x}_{ij} - \bar{\mathbf{x}}_{\eta_{ij}})^T \mathbf{\Sigma}^{-1} (\mathbf{x}_{ij} - \bar{\mathbf{x}}_{\eta_{ij}}) \\ &\qquad\qquad \text{with } \bar{\mathbf{x}}_{\eta_{ij}} = \frac{1}{|N|} \sum_{\tau \in N} \mathbf{x}_{ij+\tau}^{(p)}, \end{aligned} \tag{13}$$

$$\text{subject to } \ \mathbf{a}^T \mathbf{x}_{ij} = y_{ij}, \quad 0 \le r_{ij}, g_{ij}, b_{ij} \le 255. \tag{14}$$

2.2 Initial Color Estimation

Since the color estimation shown by Eq. (10) is carried out iteratively, an initial color image is needed to start the iterative procedure. Assuming that color vectors for K pixels, $\mathbf{s}_{i_k j_k}, k = 1, \ldots, K$ are given, we consider an initial color estimation procedure which consists of two steps.

(1) Selection of a reference color vector
In order to estimate an initial color image $\mathbf{x}_L^{(0)} = \{\mathbf{x}_{ij}^{(0)}; (i,j) \in L\}$, a reference color vector for each pixel is selected from given K references, $\mathbf{s}_{i_k j_k}, k = 1, \ldots, K$. The used measure to select a reference for (i,j) pixel is

$$F_{ij}(k) = w \frac{\{(i - i_k)^2 + (j - j_k)^2\}^{1/2}}{(N_1 + N_2)/2} + (1 - w) \frac{|y_{ij} - \mathbf{a}^T \mathbf{s}_{i_k j_k}|}{255}, \tag{15}$$

where w is a weighting factor, and the first term measures a spatial distance from a reference $\mathbf{s}_{i_k j_k}$ and the second term measures a difference between (i,j) pixel's brightness y_{ij} and that of $\mathbf{s}_{i_k j_k}$. The reference $\mathbf{s}_{i_k j_k}$ which minimizes $F_{ij}(k)$ is selected for the (i,j) pixel. An appropriate value of w is determined experimentally.

(2) Color estimation using a reference
Once a reference $\mathbf{s}_{i_k j_k}$ is selected for (i,j) pixel, an initial estimate $\mathbf{x}_{ij}^{(0)}$ can be determined as the closest point to $\mathbf{s}_{i_k j_k}$ within the plane $\mathbf{a}^T \mathbf{x}_{ij} = y_{ij}$. Such a point is usually the projection vector of $\mathbf{s}_{i_k j_k}$ onto the plane, which is derived as

$$\mathbf{x}_{ij}^{(0)} = \mathbf{s}_{i_k j_k} + \frac{y_{ij} - \mathbf{a}^T \mathbf{s}_{i_k j_k}}{\mathbf{a}^T \mathbf{a}} \mathbf{a}. \tag{16}$$

However the derived projection point $\mathbf{x}_{ij}^{(0)} = (r_{ij}^{(0)}, g_{ij}^{(0)}, b_{ij}^{(0)})^T$ is sometimes out of the range of $0 \le r_{ij}, g_{ij}, b_{ij} \le 255$. Regarding how to determine the closest point for such a case, see Ref [5].

3 Color Image Estimation in YCbCr Space

Let $\mathbf{x}_{ij} = (y^c_{ij}, c^b_{ij}, c^r_{ij})^T$ denote a color vector at (i, j) pixel in YCbCr space, where y^c_{ij} is a luminance component and c^b_{ij} and c^r_{ij} are two chrominance components. In the following, let $\mathbf{c}_{ij} = (c^b_{ij}, c^r_{ij})^T$ for notational convenience. Considering that y^c_{ij} in a color image is equal to y_{ij} in its monochrome image and using the same kind of Gaussian MRF in YCbCr space as shown in (11) and (12), the local MAP estimation for $\mathbf{c}_{ij}$ becomes the following minimization problem:

$$\begin{aligned} &\text{minimize} \quad (\mathbf{x}_{ij} - \bar{\mathbf{x}}_{\eta_{ij}})^T \mathbf{\Sigma}^{-1} (\mathbf{x}_{ij} - \bar{\mathbf{x}}_{\eta_{ij}})|_{y^c_{ij}=y_{ij}} \\ &\qquad\qquad \text{with } \bar{\mathbf{x}}_{\eta_{ij}} = \frac{1}{|N|} \sum_{\tau \in N} \mathbf{x}^{(p)}_{ij+\tau}. \end{aligned} \tag{17}$$

Note that the local MAP estimation in YCbCr space becomes a simple unconstrained optimization problem, whereas in RGB space it is a constrained one.

The solution of (17) is explicitly described as follows. Let the covariance matrix in the Gaussian MRF in YCbCr space, $\mathbf{\Sigma} = \begin{pmatrix} \sigma_{11} & \sigma_{12} & \sigma_{13} \\ \sigma_{21} & \sigma_{22} & \sigma_{23} \\ \sigma_{31} & \sigma_{32} & \sigma_{33} \end{pmatrix} = \begin{pmatrix} \sigma_y & \mathbf{\Sigma}_{yc} \\ \mathbf{\Sigma}_{cy} & \mathbf{\Sigma}_c \end{pmatrix}$, where $\sigma_y = \sigma_{11}$, $\mathbf{\Sigma}_c = \begin{pmatrix} \sigma_{22} & \sigma_{23} \\ \sigma_{32} & \sigma_{33} \end{pmatrix}$, and $\mathbf{\Sigma}_{yc} = (\sigma_{12}, \sigma_{13}) = \mathbf{\Sigma}^T_{cy}$. The Gaussian MRF in YCbCr space $p(\mathbf{x}_{ij} \mid \mathbf{x}_{\eta_{ij}})$ can be decomposed as

$$p(\mathbf{x}_{ij} \mid \mathbf{x}_{\eta_{ij}}) = p(y^c_{ij} \mid y^c_{\eta_{ij}}) p(\mathbf{c}_{ij} \mid y^c_{ij}, \mathbf{x}_{\eta_{ij}}), \tag{18}$$

$$p(y^c_{ij} \mid y^c_{\eta_{ij}}) = \frac{1}{(2\pi)^{1/2}\sigma_y^{1/2}} \exp\{-\frac{1}{2\sigma_y}(y^c_{ij} - \bar{y}^c_{\eta_{ij}})^2\}, \tag{19}$$

$$p(\mathbf{c}_{ij} \mid y^c_{ij}, \mathbf{x}_{\eta_{ij}}) = \frac{1}{(2\pi)|\mathbf{\Sigma}_{c|y}|^{1/2}} \exp\{-\frac{1}{2}(\mathbf{c}_{ij} - \mathbf{m}_{c|y})^T \mathbf{\Sigma}^{-1}_{c|y} (\mathbf{c}_{ij} - \mathbf{m}_{c|y})\}, \tag{20}$$

where

$$\mathbf{m}_{c|y} = \bar{\mathbf{c}}_{\eta_{ij}} + \mathbf{\Sigma}_{cy}\sigma_y^{-1}(y^c_{ij} - \bar{y}^c_{\eta_{ij}}), \tag{21}$$

$$\mathbf{\Sigma}_{c|y} = \mathbf{\Sigma}_c - \mathbf{\Sigma}_{cy}\sigma_y^{-1}\mathbf{\Sigma}_{yc}. \tag{22}$$

Considering that $y^c_{ij} = y_{ij}$ and the maximum of (18) is derived at $\mathbf{c}_{ij} = \mathbf{m}_{c|y}$, the reestimate of $\mathbf{c}_{ij}$, $\mathbf{c}^{(p+1)}_{ij}$ is derived as

$$\mathbf{c}_{ij}^{(p+1)} = \bar{\mathbf{c}}_{\eta_{ij}} + \mathbf{\Sigma}_{cy}\sigma_y^{-1}(y_{ij} - \bar{y}_{\eta_{ij}}), \tag{23}$$

where

$$\bar{\mathbf{c}}_{\eta_{ij}} = \frac{1}{|N|}\sum_{\tau \in N} \mathbf{c}_{ij+\tau}^{(p)}, \tag{24}$$

$$\bar{y}_{\eta_{ij}} = \frac{1}{|N|}\sum_{\tau \in N} y_{ij+\tau}. \tag{25}$$

In initial color estimation, chrominance components of a selected reference for (i, j) pixel are used as those of $\mathbf{x}_{ij}^{(0)}$, i.e., $\mathbf{c}_{ij}^{(0)}$. Therefore no additional computation is required in YCbCr space, while in RGB space, calculation of (16) is necessary and furthermore complex exceptional processing described in Ref [5] is required.

4 Experimental Results

In order to compare colorization performance in YCbCr space with that in RGB space, experiments were carried out using four standard color images (Lena, Milkdrop, Peppers, Mandrill). These images are 256 × 256 pixels in size and 24 bit per pixel (bpp) full color images. Their monochrome images were produced by the transform shown in (1) from the original color images and used for colorization experiments. For initial color estimation, several numbers of reference color vectors were given from each original image, whose positions in the image were randomly selected. It is fair to select reference positions randomly because colorization performance depends on positions of given references. The weight value w in (15) was set as follows. After optimal weight values, which depend on the number of references as well as image, were determined experimentally, the average of optimal weight values for four images was used; the used w was 0.3.

The local MAP estimation in RGB space, i.e., the constrained quadratic programming problem in (13) and (14), was here directly solved using a quadratic programming solver [8]. In YCbCr space, the solution of (17) is given in (23) with (24) and (25). In the calculation of $\bar{\mathbf{x}}_{\eta_{ij}}$ in (13) and in (17), the third-order neighborhood[2] was used and $\mathbf{x}_{ij+\tau}^{(p)}$ whose luminance value $y_{ij+\tau}$ is far from y_{ij} was excluded from the calculation. In the following experiments, if $|y_{ij+\tau} - y_{ij}| > 0.5s$, where s is the standard deviation of luminance values

---

[2] For the third-order neighborhood, $N = \{(0,1), (0,-1), (1,0), (-1,0), (1,1), (-1,-1), (1,-1), (-1,1), (0,2), (0,-2), (2,0), (-2,0)\}$

averaged over four images, $\mathbf{x}_{ij+\tau}^{(p)}$ was excluded from the calculation of $\bar{\mathbf{x}}_{\eta_{ij}}$. For the covariance matrix $\mathbf{\Sigma}$ in (13) and (17), the average of normalized covariance matrices (normalized by their maximum components) for four images was used.

Colorization performance using 25 references measured by PSNR value and CIELAB distance is shown in Table 1 and Table 2, respectively. Experiments were carried out 20 times using randomly selected references and each result is shown as mean value ± standard deviation of 20 experimental values in the tables. For each image, the upper row shows performance of initial color estimation and the lower row shows the final result after the iterative MAP estimation. Iterations were stopped when the difference of estimated color components averaged over all pixels at a current and the previous iteration became less than 0.5. Mean of the number of iterations is also given in Table 1. It is seen that colorization performance in YCbCr measured by PSNR is comparable to that in RGB and that in YCbCr measured by CIELAB distance is a little bit better than that in RGB. Regarding computation time, colorization in YCbCr took approximately only one fourth the computation time in RGB, though even in RGB space it took only 6 seconds at most to colorize one image. This time reduction is due to the aforementioned unconstrained optimization in YCbCr space resulting in the simple computation shown in (23). Note that in RGB space, a certain amount of numerical computation is needed to solve the constrained quadratic programming problem.

From Table 1, it is seen that improvement on PSNR value by MAP estimation is not very significant. However, it is still effective to a certain extent and in fact considerable improvement can be visually perceived (see Fig. 1). Fig. 1 shows colorization results in YCbCr space, where an average result among 20 experiments is shown for each image. PSNR values of initial and final results are 25.3 and 26.4 dB for Lena, 23.2 and 24.2 dB for Milkdrop, 20.7 and 23.0 dB for Peppers, 17.2 and 19.5 dB for Mandrill.

5 Application to JPEG Color Images

We address a meaningful application of the proposed colorization in YCbCr space, i.e., its application to JPEG [7] compressed color images. JPEG is a commonly used standard to compress digital color images. In JPEG compression, R, G, B color components are converted to Y, Cb, Cr components and each of the three color planes is processed independently. The chrominance planes, Cb and Cr planes are usually downsampled by a factor of two. Each plane is divided into 8 × 8 blocks and the discrete cosine transform (DCT) is then applied to each block. Derived DCT coefficients are quantized and quantized DCT coefficients are then entropy coded usually by Huffman coding. JPEG decompression is performed in a reverse way of compression except for the following point. After the inverse DCT, the downsampled Cb and Cr planes are interpolated by repetition followed by spatial smoothing with a low-pass filter.

The proposed colorization in YCbCr space is applied to JPEG compressed color images, where the interpolated and smoothed chrominance components are used as initial values $\mathbf{c}_{ij}^{(0)}$s for the iterative MAP estimation. The iterative MAP estimation shown in (23) has a structure that chrominance components are estimated considering luminance component which is not downsampled, and therefore we can expect a better recovery of them. In the application to JPEG color images, one or two iterations were enough for the MAP estimation and the following experimental results are those by one iteration. The covariance matrix in (17) for each image was computed using each JPEG compressed image.

Experimental results are shown in Fig. 2 and Fig. 3, where results by a restoration method using curvature-preserving PDE [9] are also shown for performance comparison. In these figures, PSNR values, CIELAB and S-CIELAB[3] distances are plotted for four different quality factor (qf) images: qf=60, 70, 80, and 90, and the leftmost and the rightmost point of each line correspond to qf=60 and qf=90, respectively. Larger quality factor image has higher quality, larger PSNR value and smaller CIELAB distance with larger bit rate (larger file size). Fig. 2 shows results for Lena of three different pixel sizes: 128×128, 256×256, and 512×512 pixels. From Fig. 2(a), it is seen that the proposed colorization method makes improvement on PSNR value better than the PDE-based restoration method and significant improvement is achieved particularly in small size cases. From Fig. 2(b) on CIELAB distance, an obvious advantage of the proposed method is observed. From Fig. 2(c) on S-CIELAB distance, effectiveness of the proposed method is observed in smaller size cases. Fig. 3 shows colorization results applied to JPEG compressed four color images of 256×256 pixels in size. From Fig. 3(a), it is seen that colorization is effective to improve PSNR value of JPEG compressed images except for Peppers with qf=90 and Mandrill. From Fig. 3(b), it is seen that the colorization method works better (usually much better) than the restoration method except for Peppers with qf=90, and it works even for Mandrill to some extent. From Fig. 3(c), it is seen that the colorization method works better than the restoration method for Lena and Peppers but worse for Milkdrop. Fig. 4 shows closeup results for Lena and Milkdrop, where quality improvement can be visually perceived by careful inspection. From these results, it could be addressed that colorization in YCbCr space is usually effective for quality improvement of

---

[3] S-CIELAB is a spatial extension of CIELAB proposed by Zhang and Wandel [10] to account for how spatial pattern influences color appearance and color discrimination.

JPEG color images except for very complex and/or textured images such as Mandrill.

6 Conclusions

This paper presented a colorization method in YCbCr space, which is in principle based on the MAP estimation of a color image given a monochrome image as is our previous method in RGB space. The presented method in YCbCr space is much simpler than that in RGB space and requires much less computation time: about one fourth the computation time in RGB space. As for quality of estimated color image, both methods in YCbCr and RGB space produce color images with comparable PSNR values.

The proposed colorization in YCbCr space was applied to JPEG compressed color images aiming at better recovery of downsampled chrominance planes. Experimental results show that colorization in YCbCr space is usually effective for quality improvement of JPEG color images and works better than a PDE-based restoration method.

References

[1] T. Welsh, M. Ashikhmin, K. Mueller, Transferring color to greyscale images, ACM Transactions on Graphics 21 (2002) 277-280.

[2] A. Levin, D. Lischinski, Y. Weiss, Colorization using optimization, ACM Transactions on Graphics 23 (2004) 689-694.

[3] T. Horiuchi, Colorization algorithm using probabilistic relaxation, Image and Vision Computing 22 (2004) 197-202.

[4] A. Rosenfeld, R.A. Hummel, S.W. Zucker, Scene labeling using relaxation operation, IEEE Transactions on Systems, Man and Cybernetics SMC-6 (1976) 420-433.

[5] H. Noda, J. Korekuni, M. Niimi, A colorization algorithm based on local MAP estimation, Pattern Recognition, 39 (2006) 2212-2217.

[6] S. Geman, D. Geman, Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images, IEEE Trans. Pattern Anal. & Machine Intell., PAMI-6 (1984) 721-741.

[7] W.B. Pennebaker, J.L. Mitchell, JPEG: Still Image Data Compression Standard, Van Nostrand Reinhold, 1993.

[8] H.D. Mittelmann, P. Spellucci, Decision tree for optimization software Web site (http://plato.asu.edu/guide.html).

[9] D. Tschumperle, Fast anisotropic smoothing of multi-valued images using curvature-preserving PDE's, International Journal of Computer Vision, 68 (2006) 65-82. Web site (http://www.greyc.ensicaen.fr/~dtschump/greycstoration/index.html).

[10] X. Zhang and B.A. Wandell, A spatial extension of CIELAB for digital color image reproduction, Society for Information Display Symposium Technical Digest, 27 (1996) 731-734. Web site (http://white.stanford.edu/~brian/scielab/scielab.html).

Captions

Table 1 Colorization performance (PSNR(dB)) using 25 references in YCbCr and RGB space

Table 2 Colorization performance (CIELAB distance) using 25 references in YCbCr and RGB space

Fig. 1. Colorization results in YCbCr space: (a) original color image, (b) monochrome image with randomly selected 25 references, (c) initially estimated color image, (d) finally estimated color image.

Fig. 2. Experimental results for JPEG compressed color image Lena of 128 × 128, 256 × 256, and 512 × 512 pixels in size. Performance is measured by (a) PSNR, (b) CIELAB distance, and (c) S-CIELAB distance.

Fig. 3. Experimental results for JPEG compressed four color images of 256 × 256 pixels in size. Performance is measured by (a) PSNR, (b) CIELAB distance, and (c) S-CIELAB distance.

Fig. 4. Colorization results applied to JPEG compressed color images: (a) original color image, (b) JPEG compressed image with qf=60, (c) colorization applied image to (b).

Table 1
Colorization performance (PSNR(dB)) using 25 references in YCbCr and RGB space

image		YCbCr	RGB
	initial	25.1 ± 1.0	24.9 ± 0.9
Lena	final (♯ iterations)	26.4 ± 1.1 (4.1)	26.3 ± 1.0 (4.2)
	initial	23.4 ± 1.1	23.6 ± 1.5
Milkdrop	final (♯ iterations)	24.3 ± 1.1 (5.4)	24.2 ± 1.5 (5.2)
	initial	20.8 ± 0.5	20.7 ± 0.5
Peppers	final (♯ iterations)	23.0 ± 0.9 (7.9)	22.6 ± 0.8 (7.9)
	initial	17.3 ± 0.7	17.1 ± 0.7
Mandrill	final (♯ iterations)	19.4 ± 1.0 (9.1)	19.3 ± 1.0 (9.5)

Table 2
Colorization performance (CIELAB distance) using 25 references in YCbCr and RGB space

image		YCbCr	RGB
	initial	8.5 ± 0.8	11.3 ± 1.7
Lena	final	7.3 ± 0.8	8.7 ± 1.8
	initial	21.2 ± 2.1	22.4 ± 3.2
Milkdrop	final	20.5 ± 1.7	21.2 ± 2.6
	initial	31.8 ± 2.5	32.2 ± 2.1
Peppers	final	24.0 ± 3.4	24.4 ± 3.1
	initial	22.7 ± 3.1	25.0 ± 4.0
Mandrill	final	17.4 ± 2.9	19.2 ± 3.4

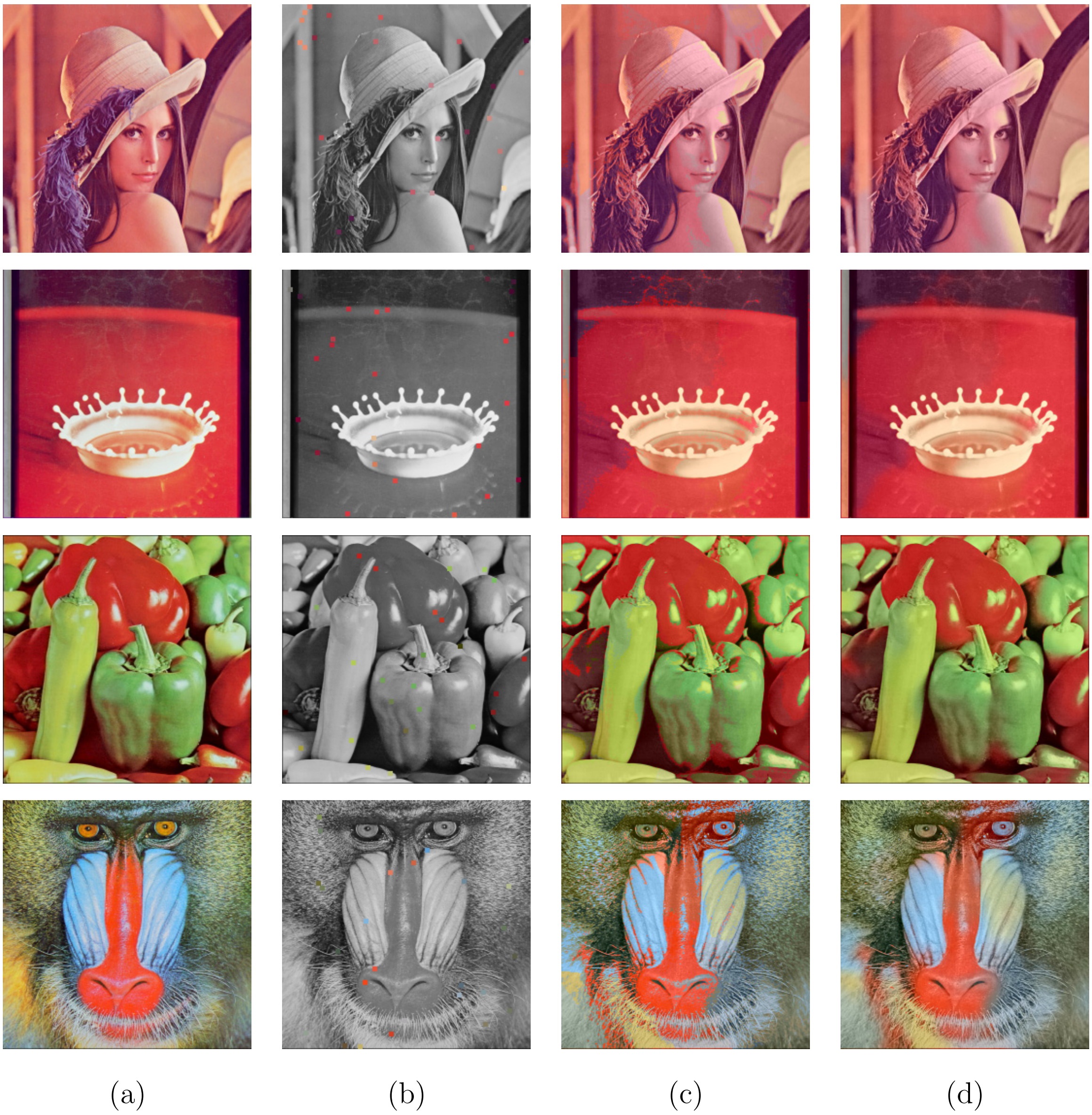


Fig. 1. Colorization results in YCbCr space: (a) original color image, (b) monochrome image with randomly selected 25 references, (c) initially estimated color image, (d) finally estimated color image.

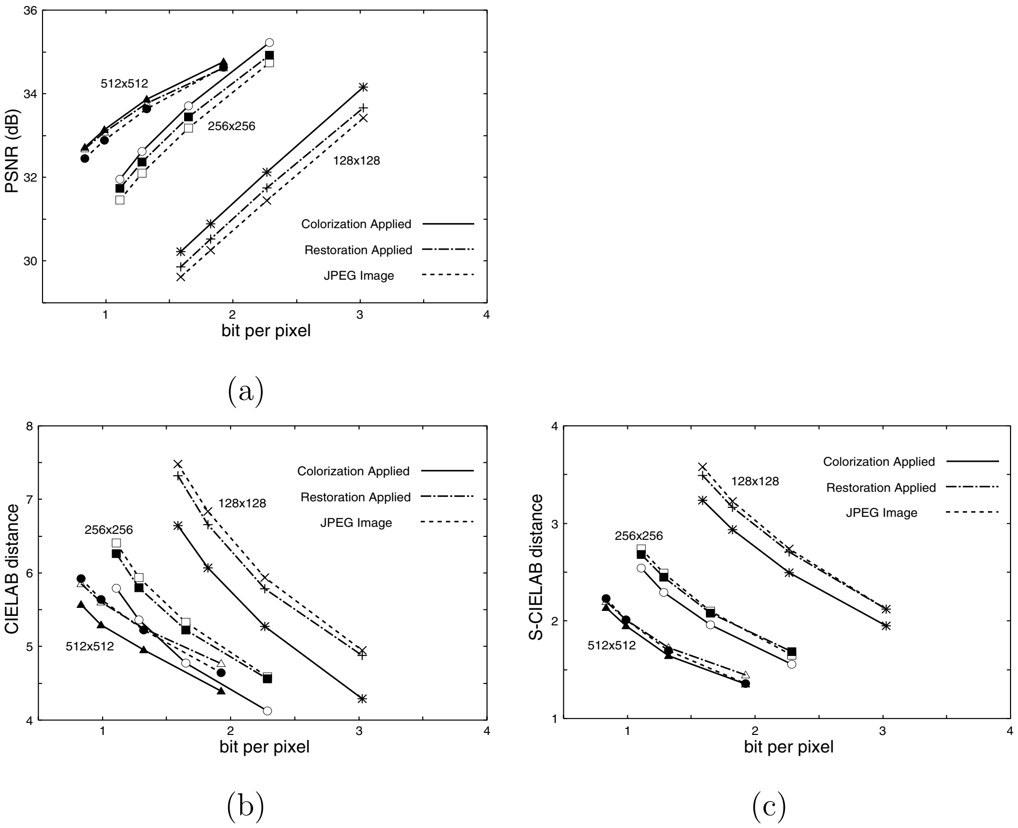
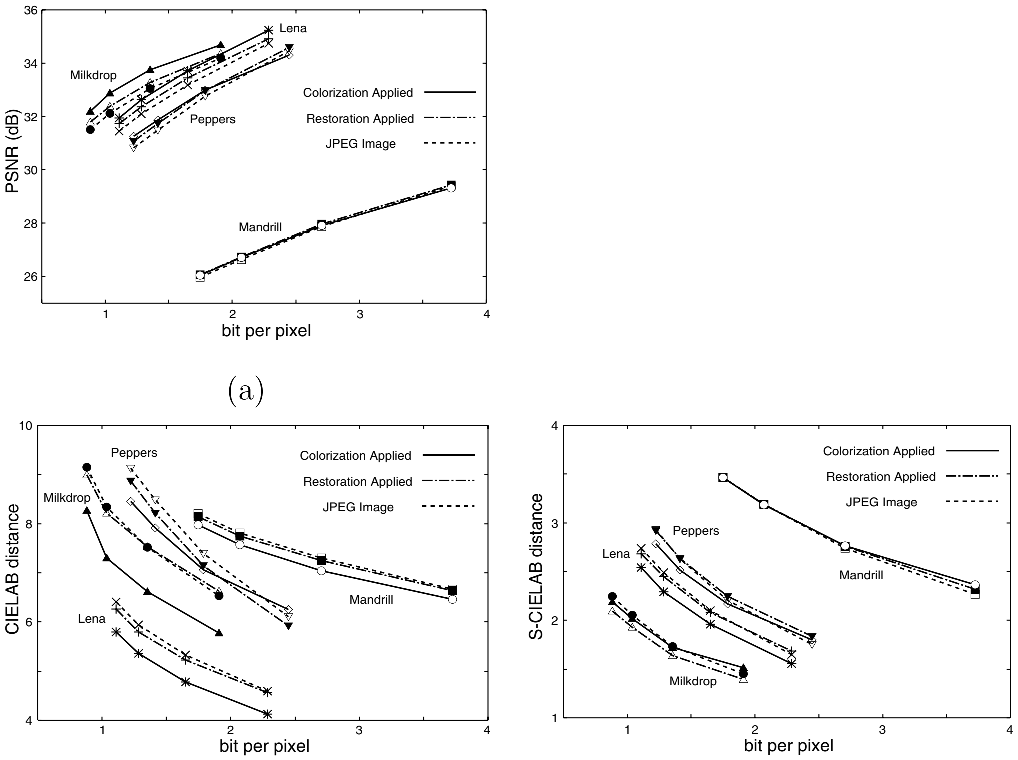


Fig. 2. Experimental results for JPEG compressed color image Lena of 128 × 128, 256 × 256, and 512 × 512 pixels in size. Performance is measured by (a) PSNR, (b) CIELAB distance, and (c) S-CIELAB distance.

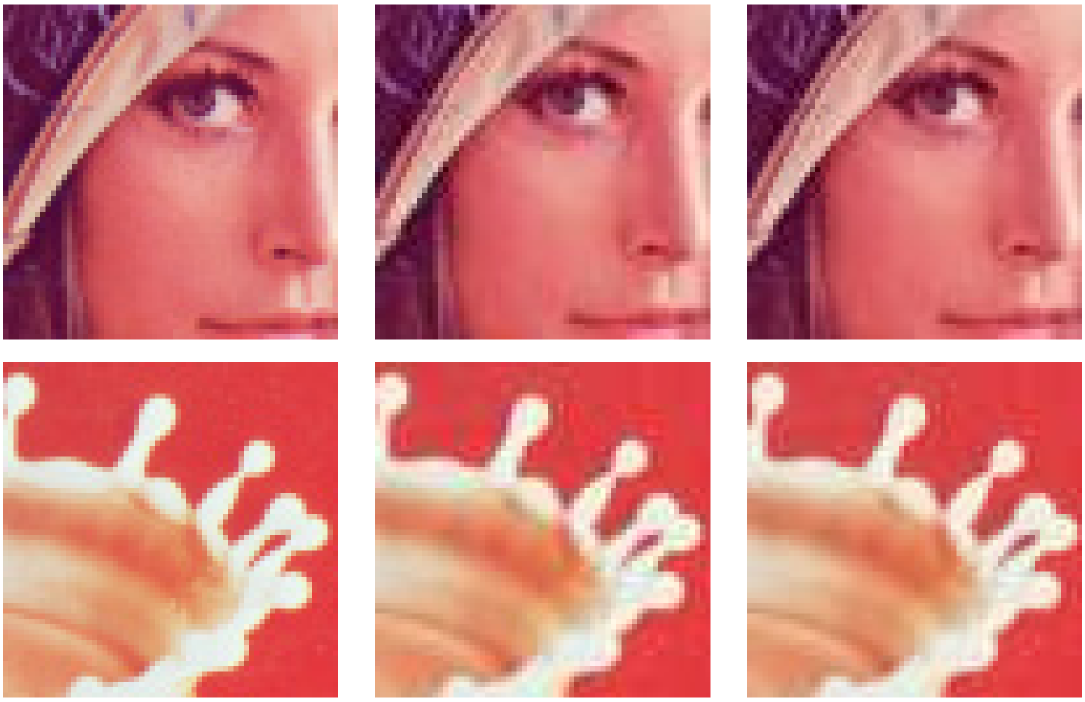


(a)

(b)

(c)

Fig. 3. Experimental results for JPEG compressed four color images of 256 × 256 pixels in size. Performance is measured by (a) PSNR, (b) CIELAB distance, and (c) S-CIELAB distance.



(a) (b) (c)

Fig. 4. Colorization results applied to JPEG compressed color images: (a) original color image, (b) JPEG compressed image with qf=60, (c) colorization applied image to (b).