Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

A two-stage parametric subspace model for efficient contrast-preserving decolorization*

Hong-yang LU[1,2], Qie-gen LIU[†‡1], Yu-hao WANG[1,2], Xiao-hua DENG[1,2]
([1]Department of Electronic Information Engineering, Nanchang University, Nanchang 330031, China)
([2]Institute of Space Science and Technology, Nanchang University, Nanchang 330031, China)
[†]E-mail: liuqiegen@ncu.edu.cn

Received Jan. 12, 2016; Revision accepted May 22, 2016; Crosschecked Nov. 26, 2017

Abstract: The RGB2GRAY conversion model is the most popular and classical tool for image decolorization. A recent study showed that adapting the three weighting parameters in this first-order linear model with a discrete searching solver has a great potential in its conversion ability. In this paper, we present a two-step strategy to efficiently extend the parameter searching solver to a two-order multivariance polynomial model, as a sum of three subspaces. We show that the first subspace in the two-order model is the most important and the second one can be seen as a refinement. In the first stage of our model, the gradient correlation similarity (Gcs) measure is used on the first subspace to obtain an immediate grayed image. Then, Gcs is applied again to select the optimal result from the immediate grayed image plus the second subspace-induced candidate images. Experimental results show the advantages of the proposed approach in terms of quantitative evaluation, qualitative evaluation, and algorithm complexity.

Key words: Color-to-gray conversion; Subspace modeling; Two-order polynomial model; Gradient correlation similarity; Discrete searching
https://doi.org/10.1631/FITEE.1600017 **CLC number:** TM346

1 Introduction

Color-to-gray conversion is a fundamental problem for many real-world applications in image processing and computer vision. It aims to convert a color image into a grayscale one. One advantage of this conversion is that it enables the application of single-channel algorithms to the processing of color images, like the Canny operator for edge detection. Other applications include monochrome printing and photograph rendering. All these applications have prompted the development of various color-to-gray conversion methods in the past decade. Unfortunately, since color-to-gray conversion involves mapping a three-dimentional (3D) vector to a 1D scalar, it is essentially a dimensionality-reduction process, and hence inevitably suffers from information loss. As a result, advanced algorithms have been developed to exploit the limited range in gray scales to present the contrasts and details of the input color image.

If the source image is in a red, green, and blue (RGB) format, the most well-known method to decolorize an input image is to linearly sum its R, G, and B channels with a fixed weight (e.g., the rgb2gray() in Matlab). Nevertheless, using a luminance channel image alone cannot faithfully represent structures and contrasts in some color images, such as those having iso-luminant regions. In recent years, significant advances have been made in theories and algorithms for perception-driven decolorization, resulting in a large

‡ Corresponding author
* Project supported by the National Basic Research Program (973) of China (No. 2013CB035600), the National Natural Science Foundation of China (Nos. 61261010, 61362001, and 61503176), Jiangxi Provincial Advanced Projects for Post-Doctoral Research Funds of China (No. 2014KY02), the International Postdoctoral Exchange Fellowship Program, and the International Scientific and Technological Cooperation Projects of Jiangxi Province, China (No. 20141BDH80001)
ORCID: Qie-gen LIU, http://orcid.org/0000-0003-4717-2283

number of new methods. These methods can be classified broadly into two categories: (1) local adjustment methods exploiting the local distribution of color pixel values (Bala and Eschbach, 2004; Neumann *et al*., 2007; Smith *et al*., 2008); (2) global adjustment methods, including objective minimization methods using differences between the original color values and mapped gray values.

In the first category, color-to-gray mapping usually treats pixels of the same color differently to enhance local contrast, according to different attributes such as local chrominance edges. Although they may have advantages in accurately preserving local features, constant color regions could be converted homogeneously if the mapping changes in the regions. Bala and Eschbach (2004) presented a method to enhance chromatic edges, adding high-frequency components of chromaticity to the luminance channel. Neumann *et al.* (2007) reconstructed a grayscale image from the gradients of a color image by measuring the color and luminance contrasts as a gradient contrast of the coloroid space. Smith *et al.* (2008) decomposed the image into several frequency components and adjusted combination weights using chromatic channels. These methods can preserve local features; however, they might occasionally distort the appearance of constant color regions, which will not happen in global mapping.

Unlike local adjustment methods, global color-to-gray conversion methods consider the image as a whole, and apply some consistent mapping functions to all the pixels in the image. Thus, pixels of the same color will be converted to the same grayscale values. Gooch *et al.* (2005) proposed the color2gray algorithm to generate grayscale values that best match the differences between pixel values of the input color image. Rasche *et al.* (2005) minimized a linear color mapping function to achieve an optimal conversion. The function was defined by constraints imposed directly on different color pairs. Grundland and Dodgson (2007) proposed a fast algorithm to compute a linear mapping that adds the chromatic channel information to that of the luminance channel. Kim *et al.* (2009) proposed a fast color-to-gray conversion algorithm which preserves the color ordering according to the original lightness of colors. Kuk *et al.* (2010) extended the method proposed by Gooch *et al.* (2005) by considering both local and global contrasts. Ancuti *et al.* (2010) proposed a decolorization strategy based on image fusion principles and using three different weight maps to control saliency, exposure, and saturation. To maximize the variance of the output grayscale image, Jin *et al.* (2014) presented a regularization term to make the resulting objective function convex and obtained a stable combination of R, G, and B pixel values. Wang and Yau (2014) used image decolorization to enhance the color contrast when preserving the grayscale in a real-time moustache detection.

Unfortunately, all these approaches suffer from two disadvantages: lack of robustness and high computational cost. To alleviate these difficulties, researchers have revisited the simple and conventional rgb2gray model. This model assumes that the grayscale output g is a constrained linear combination of the R, G, and B channels of the color image I, i.e., $g=\sum_{w_c\in Z} \boldsymbol{w}_c \boldsymbol{I}_c$, $Z=\left\{\boldsymbol{w}_c \middle| \sum_c \boldsymbol{w}_c=1, \boldsymbol{w}_c\geq 0, c=\{r,g,b\}\right\}$, and I_r, I_g, I_b stand for the R, G, and B channels, respectively. In the classical rgb2gray() in Matlab, the weights are fixed as (w_r, w_g, w_b)=(0.2928, 0.5870, 0.1140) for all images. Recently, a few researchers have adaptively chosen the channel weights by applying some measures to the discretized candidate images. Real-time contrast preserving (RTCP) decolorization proposed by Lu *et al.* (2012b) discretizes the solution space of a linear parametric model with 66 candidates, and then identifies one candidate that achieves the smallest gradient error based energy value as the optimal solution. This is currently the fastest algorithm. Song *et al.* (2013) discussed the robustness of the existing methods, and investigated multi-scale contrast preservation on the discretized candidate images by taking advantages of the (joint) bilateral filter. Liu *et al.* (2015) presented a gradient correlation similarity (Gcs) model, and identified the solution with the maximum Gcs from the linear parametric model induced discretized candidate images.

All the three methods described by Lu *et al.* (2012b), Song *et al.* (2013), and Liu *et al.* (2015) are based on the rgb2gray model, yet apply different measures to identify the ‘good’ results from finite candidate images. Motivated by their impressive performance, we aim to extend the parametric discrete searching technique from the linear model to a two-order multivariance polynomial model. In this paper, we present a two-stage strategy to achieve this goal: after dividing the whole space into three subspaces, Gcs measures are used to obtain an immediate image from the first subspace (i.e., the linear parametric model). Then, based on the immediate image, we select the final image by using the Gcs measures again to determine the parameter weights in the second subspace.

2 The proposed model

In this section, we first investigate the two-order multivariance polynomial model based on subspace modeling. Then the differences between the three subspaces are revealed. Finally, a discrete-searching solver is presented for a real-time and effective color-to-gray conversion.

2.1 Two-order multivariance polynomial model

To retain a better feature discrimination in a color-to-gray conversion, we aim to extend the rgb2gray model used by Lu *et al.* (2012b), Song *et al.* (2013), and Liu *et al.* (2015) to a two-order multivariance polynomial model. As in the CP model of Lu *et al.* (2012a), we assume that grayscale g is a two-order multivariate polynomial function for mapping:

$$\begin{aligned} &g=\sum_{\boldsymbol{m}_c\in \boldsymbol{Z}} \boldsymbol{w}_c\boldsymbol{m}_c, \\ &Z=\{I_r,I_g,I_b,I_rI_g,I_rI_b,I_gI_b,I_{r^2},I_{g^2},I_{b^2}\}, \end{aligned} \tag{1}$$

which is a parametric color-to-gray model. A major drawback in the RTCP work of Lu *et al.* (2012b) is that they treated all bases in Z equally. Also, they used an iterative optimization technique to solve the derived model, leading to high execution time. In this study, we view Eq. (1) as the sum of three subspaces:

$$g=\sum_{\boldsymbol{m}_{c_1}\in Z_1} \boldsymbol{w}_{c_2}\boldsymbol{m}_{c_1}+\sum_{\boldsymbol{m}_{c_2}\in Z_2} \boldsymbol{w}_{c_2}\boldsymbol{m}_{c_2}+\sum_{\boldsymbol{m}_{c_3}\in Z_3} \boldsymbol{w}_{c_3}\boldsymbol{m}_{c_3},$$

$$\begin{cases} Z_1=\{I_r,I_g,I_b\}, \\ Z_2=\{I_rI_g,I_rI_b,I_gI_b\}, \\ Z_3=\{I_{r^2},I_{g^2},I_{b^2}\}, \end{cases} \tag{2}$$

where Z_1, Z_2, and Z_3 are the subspaces spanned by a family of monomials. Since they are predefined by the linear combination of channel images, the pursuit of the mapping function in Eqs. (1) and (2) turns to be the determining of weight parameters $\{\boldsymbol{w}_c\}$. Section 2.2, the differences among the three subspaces will be revealed.

2.2 Survey of the three subspaces

The differences among the three subspaces are enormous. On one hand, the weight combination in subspace Z_1 provides impressive results (Lu *et al.*, 2012b; Song *et al.*, 2013; Liu *et al.*, 2015). On the other hand, if we search the candidate images from all three subspaces, the search time is huge. Therefore, an alternative strategy is needed for a tradeoff between the effectiveness and the computation cost. Basis images in subspace Z_1 are whiter, while those in subspace Z_2 (which contain cross channels) and those in subspace Z_3 are darker and more similar (Fig. 1).

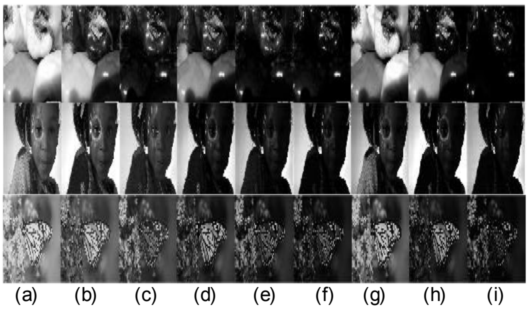

Fig. 1 Demonstration of the nine channels of three color images from Ĉadík's dataset (Ĉadík, 2008)
(a)–(i) refer to I_r, I_g, I_b, I_rI_g, I_rI_b, I_gI_b, I_{r^2}, I_{g^2}, and I_{b^2} channel images, respectively

First, we investigated the importance of the three subspaces. We took 24 color images from Ĉadík's dataset (Ĉadík, 2008) as examples for illustration. Quantitatively, the average gradient entropy of the basis images from subspaces Z_1, Z_2, and Z_3 are 1.0005, 0.9627, and 0.9147, respectively. The results of the 24 test images are shown in Fig. 2a. Except for the 15th image of the dataset, the images from subspace Z_1 contain more information than those in the other two subspaces. From a visual comparison, even the 15th image looks better (the third line in Fig. 1).

Second, we compare the similarities of the three subspaces. The average structure similarity (SSIM) (Wang *et al.*, 2004) between subspaces Z_1 and Z_3 is

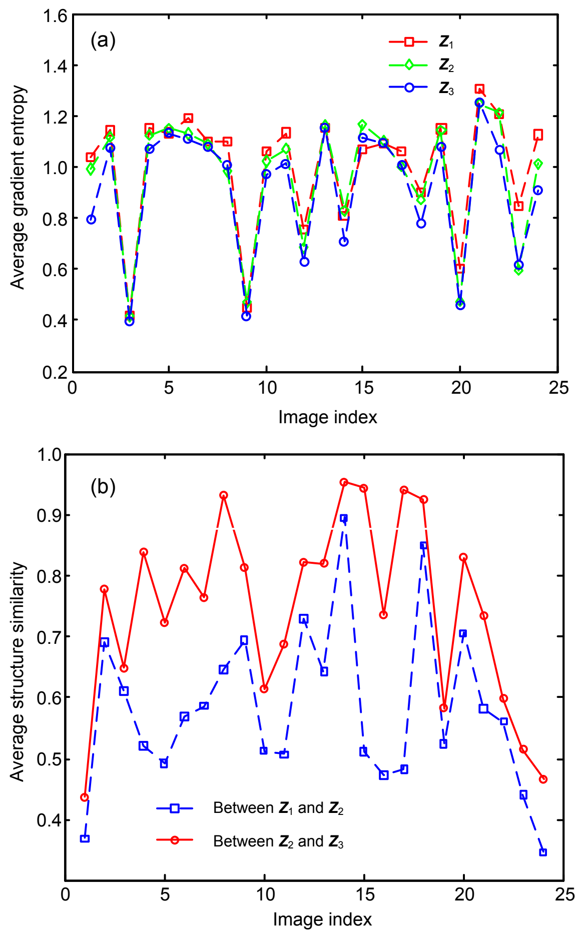

Fig. 2 The average gradient entropies of basis images in Z_1, Z_2, and Z_3 from Čadík's dataset (Čadík, 2008) (a) and the average structure similarities between Z_1 and Z_2, and Z_2 and Z_3 (b)

0.5813, and that between subspaces Z_2 and Z_3 is 0.7470. The results from the 24 images are shown in Fig. 2b. Note that Z_2 and Z_3 are better corrected.

In summary, the above discussions yield two conclusions: (1) The first subspace is the most important, followed by the second and then the third; (2) Subspaces Z_2 and Z_3 are better corrected.

2.3 Two-stage parametric subspace derived discrete searching solver

Based on the fact that the first subspace Z_1 in the two-order linear parametric model is the most important and subspaces Z_2 and Z_3 are secondary important and similarly related, we propose a method applying a two-step approach to decolorization:

$$\begin{cases} g = g_1 + g_2, \\ g_1 = \sum_{\boldsymbol{m}_{c_1} \in Z_1} \boldsymbol{w}_{c_1} \boldsymbol{m}_{c_1}, \\ g_2 = \sum_{\boldsymbol{m}_{c_2} \in Z_2} \boldsymbol{w}_{c_2} \boldsymbol{m}_{c_2}. \end{cases} \tag{3}$$

An overview of the two-stage parametric subspace (TPS) method is shown in Fig. 3. In the TPS method, a measure is applied to the first subspace to obtain the immediate grayed image. Then the measure is applied again to select the optimal result from the immediate grayed image plus the second subspace-induced candidate images. Intuitively, this approach can be recognized as a refinement method. Clearly, model (3) provides a set containing only the possible grayscale intensity images. To optimize the decolorization process, a criterion should be built such that the optimal solution g, which preserves as much color image contrast as possible, can be determined. In this study, we use a discrete searching strategy to determine parameters $\{\boldsymbol{w}_c\}$.

Liu *et al.* (2015) proposed a novel Gcs measure in the rgb2gray model as follows:

$$\begin{aligned} &\min_g \mathrm{Gcs}(g) = -\sum_{(x,y)\in \boldsymbol{P}} \sum_c \frac{2(|I_{c,x} - I_{c,y}| + \varepsilon)|g_x - g_y|}{(|I_{c,x} - I_{c,y}| + \varepsilon)^2 + |g_x - g_y|^2} \\ &\text{s.t.}\ \ g = \sum_{\boldsymbol{w}_c \in Z} \boldsymbol{w}_c \boldsymbol{I}_c, \\ &\qquad Z = \left\{ \boldsymbol{w}_c \,\middle|\, \sum_c \boldsymbol{w}_c = 1, \boldsymbol{w}_c \ge 0, c = \{r, g, b\} \right\}, \end{aligned} \tag{4}$$

where ε is a positive constant that supplies numerical stability, $g_x - g_y$ denotes the difference in grayscale value between pixels g_x and g_y, $\boldsymbol{P}$ stands for a pixel pair pool which contains the local and nonlocal candidates. Liu *et al.* (2015) used the discrete searching strategy to find the approximate minimization solution of Eq. (4). Recognizing that slightly varying the weights would not change the grayscale appearance too much, they discretized the solution space of $\{w_r, w_g, w_b\}$ in the range of [0, 1] with an interval of 0.1, giving a searching space of 66. Therefore, they reduced the solution space and searched the possible optimal solution in a discrete range consisting of 66 candidates. This strategy greatly reduces the number of candidate value sets and saves executable time.

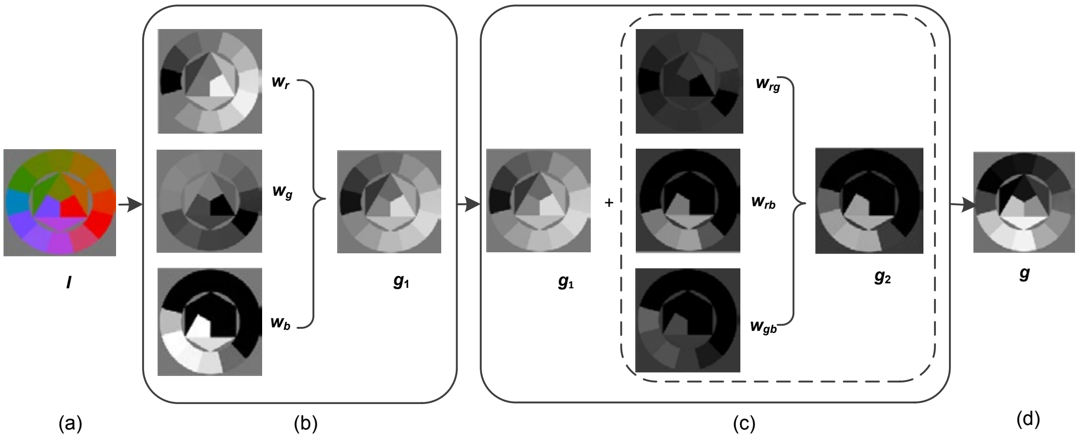

Fig. 3 An overview of the proposed two-stage method: (a) the original input; (b) the first stage: using gradient correlation similarity (Gcs) on the parameter space {w_r, w_g, w_b}; (c) the second stage: using Gcs on the parameter space {w_{rg}, w_{rb}, w_{gb}} combined with the intermediate result; (d) final output g=g_1+g_2 (References to color refer to the online version of this figure)

In this study, we employ a modified Gcs measure in a two-stage fashion as follows: In the first step, g_1 is determined by

$$\min_{g_1} \mathrm{Gcs}(g_1) = -\sum_{(x,y)\in \boldsymbol{P}}\sum_{c}\frac{2\left|I_{c,x}-I_{c,y}\right|\left|g_{1x}-g_{1y}\right|}{\left|I_{c,x}-I_{c,y}\right|^2+\left|g_{1x}-g_{1y}\right|^2+\varepsilon_1}$$

$$\text{s.t. } g_1=\sum_{\boldsymbol{w}_c\in\hat{\boldsymbol{Z}}_1}\boldsymbol{w}_c\boldsymbol{I}_c,\quad \hat{Z}_1=\left\{\boldsymbol{w}_c\left|\sum_c \boldsymbol{w}_c=1,\boldsymbol{w}_c\geq 0, c=\{r,g,b\}\right.\right\}. \tag{5}$$

Then the final result of g is chosen by solving

$$\min_{g} \mathrm{Gcs}(g) = -\sum_{(x,y)\in \boldsymbol{P}}\sum_{c}\frac{2\left|I_{c,x}-I_{c,y}\right|\left|g_{x}-g_{y}\right|}{\left|I_{c,x}-I_{c,y}\right|^2+\left|g_{x}-g_{y}\right|^2+\varepsilon_2}$$

$$\text{s.t. } g=g_1+\sum_{\boldsymbol{w}_c\in\hat{\boldsymbol{Z}}_2}\boldsymbol{w}_c\boldsymbol{m}_c,\quad \hat{Z}_2=\left\{\boldsymbol{w}_c\left|\sum_c \boldsymbol{w}_c=1,1\geq\boldsymbol{w}_c\geq -1, c=\{rg,rb,gb\}\right.\right\}. \tag{6}$$

Comparing Eqs. (5) and (6) with Eq. (4), we observe that both parameters ε_1 and ε_2 exist only in the denominator. This modification prefers to treat the differences between grayed image $|g_x-g_y|$ and color image $|I_{c,x}-I_{c,y}|$ in an equal manner. According to the discrete searching rule that discretizes the solution space of $\{\boldsymbol{w}_c\}$ with an interval of 0.1 and the range constraints for Eqs. (5) and (6), we first select the immediate result from $\hat{Z}_1$ consisting of 66 candidates, and then the final result from $\hat{Z}_2$ consisting of 231 candidates. Clearly, this greatly lowers the computation time, since if we simultaneously search $\hat{Z}_1$ and $\hat{Z}_2$, or parameter spaces of all six, the total computation time would be huge.

3 Experimental results

In this section, the performance of the proposed method is compared with those of different state-of-the-art algorithms for a variety of image styles, including those in conventional user experiments from Čadík's dataset and the Complex Scene Decolorization Dataset (CSDD) (Du *et al.*, 2015). In experiments, parameters ε_1 and ε_2 were empirically selected as 0.01 and 0.05, respectively.

3.1 Qualitative evaluation

The comparison results with Čadík's dataset (Čadík, 2008) are shown in Fig. 4. Color images and their corresponding grayscale images obtained by the Gooch05 method (Gooch *et al.*, 2005) and the Smith08 method (Smith *et al.*, 2008) are shown in Figs. 4b and 4c. Figs. 4d and 4e show in sequence the results of the algorithms developed by CP (Lu *et al.*, 2012a) and RTCP (Lu *et al.*, 2012b). The results obtained using our previous method GcsDecolor2 (denoted as Gcs2) (Liu *et al.*, 2015) are shown in Fig. 4f. Results for saliency-guided decolorization obtained

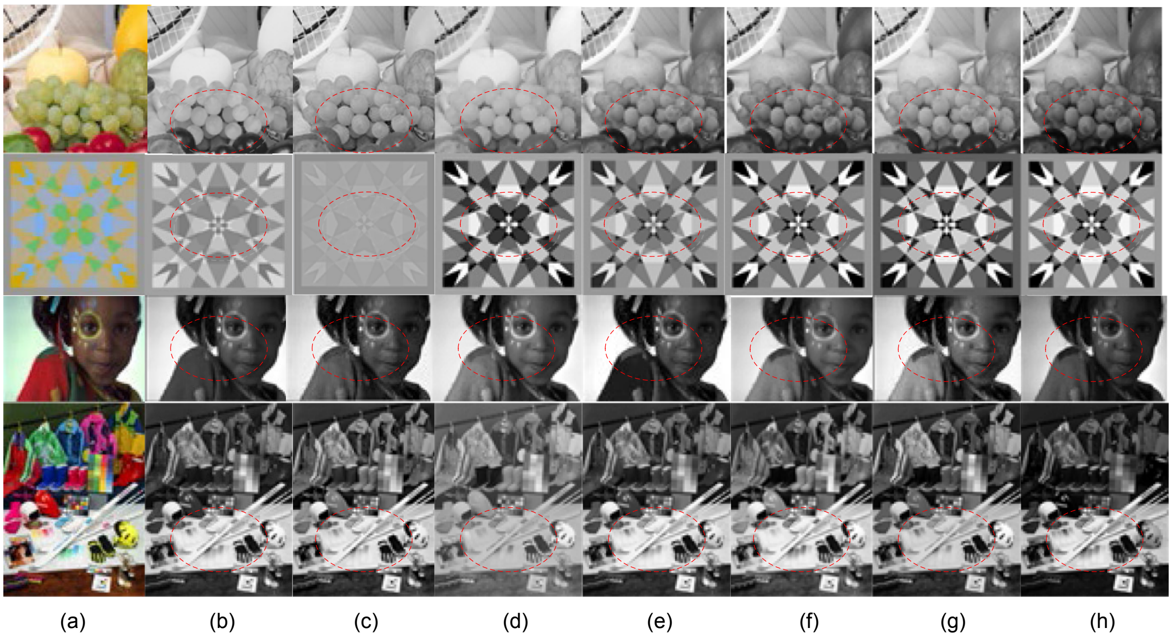

Fig. 4 Color-to-gray conversion comparison of images from Ĉadík's dataset: input color images (a), and the comparison results using Gooch05 (b), Smith08 (c), contrast preserving (CP) (d), real-time CP (RTCP) (e), Gcs2 (f), Du15 (g), and two-stage parametric subspace (TPS) (h). References to color refer to the online version of this figure

using region-based optimization and our TPS are shown in Figs. 4g and 4h, respectively. Methods Gooch05 and Smith08 do not sufficiently consider the salient stimulus, and may generate flat results for some images. For CP and RTCP, the points and edges perceived in a color image cannot be seen in the converted grayscale image (rows 1 and 4 of Fig. 4). The Du15 method (Du *et al.*, 2015) also loses part of the color contrast information, although it uses a region-based optimization. Our methods Gcs2 and TPS use the normalized correlation and preserved the salient features of the color image. In particular, the results of TPS are impressive. It has not only a good feature preservation, ensuring that features in the color image can still be discriminated in the grayscale image, but also an excellent ordering preservation, preserving a desired color ordering in color-to-gray conversion (as indicated by the red circles).

In contrast to some images in Ĉadík's dataset that are relatively simple and contain only a few colors, the CSDD dataset includes 22 different images with abundant colors and patterns. We compared the proposed method with three state-of-the-art methods using CSDD. The results (Fig. 5) show that TPS consistently outperforms the other methods in these complex scenes. The CP method shown in Fig. 5b tends to produce grayscale artifacts in some regions of the images, causing higher local contrasts than that of the original images. A loss of contrast can be seen in Fig. 5c (rows 1–3). In addition, the Du15 method does not preserve the local contrast well (rows 3 and 4 of Fig. 5d). TPS produces better results in terms of perceptual accuracy and color order.

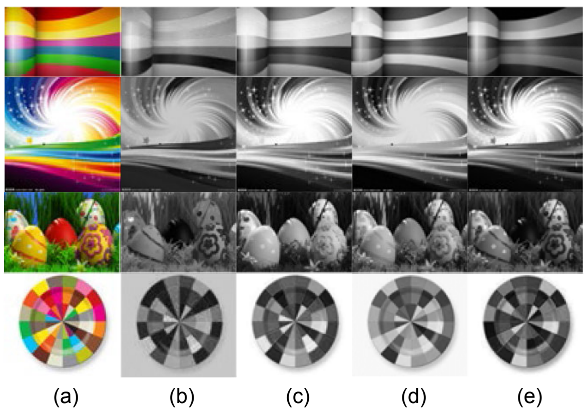

Fig. 5 Decolorization comparison of images from the CSDD dataset: input color images (a), and the comparison results using contrast preserving (CP) (b), real-time CP (RTCP) (c), Du15 (d), and two-stage parametric subspace (TPS) (e) (References to color refer to the online version of this figure)

3.2 Quantitative evaluation

The color contrast preserving ratio (CCPR) is based on the observation that if color difference $\delta_{x,y}$ is smaller than a threshold τ, it becomes nearly invisible in human vision. The aim of contrast-preserving decolorization is therefore to maintain the color change which is perceivable by humans. CCPR is defined as

$$\mathrm{CCPR} = \#\left\{(x,y)\middle|(x,y)\in\Omega,\ \left|g_x-g_y\right|\geq\tau\right\}\Big/|\Omega|, \quad (7)$$

where Ω is the set containing all neighboring pixel pairs with their original color difference $\delta_{x,y}\geq\tau$, and $|\Omega|$ is the number of pixel pairs in Ω. $\#\{(x, y)|(x, y)\in\Omega, |g_x-g_y|\geq\tau\}$ is the number of pixel pairs in Ω that are still distinctive after decolorization.

To evaluate the decolorization algorithms quantitatively in terms of contrast preservation, we show the actual data of CCPR from Ĉadík's dataset and the CSDD dataset to illustrate the significant improvement (Table 1). We present the quantitative CCPR (CP) (Lu *et al.* 2012a) for Ĉadík's dataset in Fig. 6a. For each method (TPS, Gooch05, Smith08, CP, RTCP, Gcs2, and Du15), we calculated the average CCPR for the whole dataset by varying τ from 1 to 15. Our method outperforms the other approaches at almost all different threshold levels. The quantities indicate that Gcs measure used in Gcs2 and TPS can better reflect the structural similarity between color and grayscale images, and the two-step technique improves decolorization. Although both Gooch05 and Du15 use the saliency prior information to prompt feature-preserving decolorization, the results of Du15 seem better because of the use of a two-stage strategy and regional saliency.

A quantitative comparison based on the CSDD dataset is shown in Fig. 6b. The recently reported saliency-guided region-based optimization method Du15 generally outperforms the two gradient error-guided methods CP and RTCP. Nevertheless, our TPS that employs a two-step discrete searching strategy can attain comparable or even better results than Du15. Note that although both CP and TPS determine the weight parameters from the two-order model, CP obtains an iterative solver with the gradient error measure, and TPS uses a two-step discrete search strategy with a gradient correction measure.

3.3 Algorithm complexity

A good color-conversion model is expected to be not only effective but also computationally efficient. For the computational time, Ancuti *et al.* (2011) proposed a saliency-based strategy for video decolorization. Its runtime is linearly dependent on the image resolution and therefore computationally effective and suitable for real-time applications.

Table 1 Average CCPR obtained by different methods using Ĉadík's dataset and the CSDD dataset

Threshold	CCPR using Ĉadík's dataset							CCPR using CSDD dataset			
τ	TPS	Du15	Gcs2	RTCP	CP	Smith08	Gooch05	TPS	RTCP	Du15	CP
1	**0.79**	0.79	0.78	0.75	0.76	0.70	0.69	**0.77**	0.72	0.74	0.72
2	**0.78**	0.75	0.77	0.72	0.73	0.66	0.66	**0.72**	0.65	0.67	0.65
3	**0.78**	0.75	0.77	0.72	0.72	0.64	0.64	**0.69**	0.61	0.65	0.62
4	**0.77**	0.74	0.76	0.71	0.72	0.62	0.63	**0.68**	0.60	0.62	0.59
5	**0.77**	0.73	0.75	0.7	0.72	0.61	0.63	**0.66**	0.58	0.61	0.57
6	**0.76**	0.72	0.74	0.69	0.70	0.59	0.61	**0.64**	0.57	0.59	0.55
7	**0.75**	0.71	0.73	0.68	0.69	0.58	0.60	**0.63**	0.56	0.58	0.53
8	**0.74**	0.70	0.72	0.67	0.68	0.57	0.58	**0.61**	0.55	0.56	0.51
9	**0.73**	0.69	0.71	0.66	0.67	0.56	0.57	**0.60**	0.54	0.55	0.49
10	**0.72**	0.68	0.71	0.65	0.66	0.55	0.55	**0.58**	0.53	0.54	0.48
11	**0.71**	0.67	0.69	0.64	0.65	0.54	0.54	**0.57**	0.52	0.52	0.46
12	**0.70**	0.66	0.69	0.63	0.64	0.53	0.53	**0.55**	0.52	0.51	0.45
13	**0.69**	0.65	0.68	0.62	0.63	0.52	0.52	**0.54**	0.50	0.50	0.44
14	**0.68**	0.64	0.67	0.61	0.62	0.52	0.51	**0.52**	0.49	0.49	0.42
15	**0.67**	0.63	0.67	0.60	0.61	0.51	0.50	**0.51**	0.49	0.48	0.41

Bold number refers to the the highest CCPR value. CCPR: color contrast preserving ratio; TPS: two-stage parametric subspace; CP: contrast preserving; RTCP: real-time CP

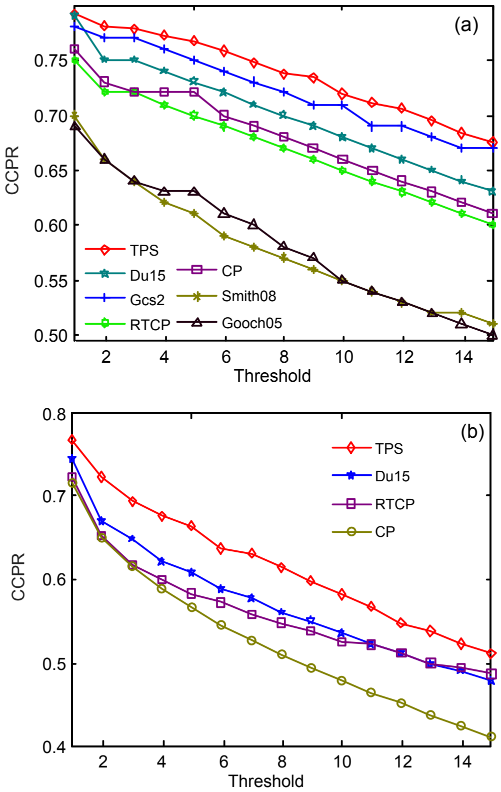

Fig. 6 Comparison of the color contrast preserving ratios (CCPR) obtained by state-of-the-art methods using Čadík's dataset (a) and the CSDD dataset (b)

TPS: two-stage parametric subspace; CP: contrast preserving; RTCP: real-time CP

Furthermore, as far as we know, both the discrete searching methods of RTCP and Gcs2 are currently the fastest. Compared with Gcs2, the extended TPS uses the Gcs2 measure in the first stage to obtain an immediate grayed image. Then, TPS applies the Gcs2 method again to select the optimal result from the immediate grayed image plus the second subspace induced candidate images, leading to a 2–3-fold longer computation time than that with the Gcs2 method. However, implementation on graphics processing units (GPUs) can be exploited to achieve significant reductions in computation time. The computational cost performance using a 550×717 color image as the input is shown in Table 2. The color image was processed on an Intel Core i7 CPU @2.40 GHz with our Matlab implementation (Matlab 7.10.0 with 64 bits). Although the proposed method is slower than RTCP and Gcs2, it still achieves a favorable performance in terms of both accuracy and efficiency.

Table 2 Computational cost evaluations of different methods

Method	Runtime (s)	Method	Runtime (s)
Ancuti	1.547	RTCP	0.054
Kim	0.685	Gcs2	0.051
CP	2.045	TPS	0.143

A 550×717 color image was used as an input. TPS: two-stage parametric subspace; CP: contrast preserving; RTCP: real-time CP

4 Conclusions and future work

In this paper, we presented a two-stage color-to-gray conversion model employing a discrete searching strategy with gradient correlation similarity measures. The two-step model and the discrete searching criterion enhance both the performance and computational efficiency. A comparative study using a wide variety of images indicated that the proposed method provides perceptually more plausible results than most other recent algorithms. Nevertheless, since the proposed method can be seen as a refinement of the previous Gcs method, the improvement of TPS over Gcs2 may be slight or even absent for some images (for instance, the 8th test image in the CSDD dataset, which is not shown here due to space limitations). Besides the Gcs measure, applying this two-stage strategy to the two-order multivariance polynomial model with other measures will be investigated in the future.

In this study, we studied only single-image decolorization. One direction for research extension is video decolorization. An important consideration is that video decolorization can yield better results by imposing temporal coherence. For our model, a possible solution to the extension is to extend the pixel difference defined in Eqs. (5) and (6) to the spatio-temporal dimension, as in the total variation video restoration model described by Chan *et al.* (2011). We could treat a video sequence as a space-time volume and pose a space-time volume difference to enhance the contrast of decolorization. This may be an interesting topic for future study.

References

Ancuti, C.O., Ancuti, C., Hermans, C., *et al.*, 2010. Image and video decolorization by fusion. Asian Conf. on Computer Vision, p.79-92. https://doi.org/10.1007/978-3-642-19315-6_7

Ancuti, C.O., Ancuti, C., Bekaert, P., 2011. Enhancing by saliency-guided decolorization. IEEE Conf. on Computer Vision and Pattern Recognition, p.257-264. https://doi.org/10.1109/CVPR.2011.5995414

Bala, R., Eschbach, R., 2004. Spatial color-to-grayscale transform preserving chrominance edge information. 12th Color and Imaging Conf., p.82-86.

Ĉadík, M., 2008. Perceptual evaluation of color-to-grayscale image conversions. *Comput. Graph. Forum*, **27**(7):1745-1754. https://doi.org/10.1111/j.1467-8659.2008.01319.x

Chan, S.H., Khoshabeh, R., Gibson, K.B., *et al.*, 2011. An augmented Lagrangian method for total variation video restoration. *IEEE Trans. Imag. Process.*, **20**(11):3097-3111. https://doi.org/10.1109/TIP.2011.2158229

Du, H., He, S., Sheng, B., *et al.*, 2015. Saliency-guided color-to-gray conversion using region-based optimization. *IEEE Trans. Imag. Process.*, **24**(1):434-443. https://doi.org/10.1109/TIP.2014.2380172

Gooch, A.A., Olsen, S.C., Tumblin, J., *et al.*, 2005. Color2gray: salience-preserving color removal. *ACM Trans. Graph.*, **24**(3):634-639. https://doi.org/10.1145/1186822.1073241

Grundland, M., Dodgson, N.A., 2007. Decolorize: fast, contrast enhancing, color to grayscale conversion. *Patt. Recogn.*, **40**(11):2891-2896. https://doi.org/10.1016/j.patcog.2006.11.003

Jin, Z., Li, F., Ng, M.K., 2014. A variational approach for image decolorization by variance maximization. *SIAM J. Imag. Sci.*, **7**(2):944-968. https://doi.org/10.1137/130935197

Kim, Y., Jang, C., Demouth, J., *et al.*, 2009. Robust color-to-gray via nonlinear global mapping. *ACM Trans. Graph.*, **28**(5):1-4. https://doi.org/10.1145/1661412.1618507

Kuk, J.G., Ahn, J.H., Cho, N.I., 2010. A color to grayscale conversion considering local and global contrast. Asian Conf. on Computer Vision, p.513-524. https://doi.org/10.1007/978-3-642-19282-1_41

Liu, Q., Liu, P.X., Xie, W., *et al.*, 2015. GcsDecolor: gradient correlation similarity for efficient contrast preserving decolorization. *IEEE Trans. Imag. Process.*, **24**(9): 2889-2904. https://doi.org/10.1109/TIP.2015.2423615

Lu, C., Xu, L., Jia, J., 2012a. Contrast preserving decolorization. IEEE Int. Conf. on. Computational Photography, p.1-7. https://doi.org/10.1109/ICCPhot.2012.6215215

Lu, C., Xu, L., Jia, J., 2012b. Real-time contrast preserving decolorization. ACM SIGGRAPH Asia Technical Briefs, Article 34. https://doi.org/10.1145/2407746.2407780

Neumann, L., Čadík, M., Nemcsics, A., 2007. An efficient perception-based adaptive color to gray transformation. Proc. 3rd Eurographics Conf. on Computational Aesthetics in Graphics, Visualization and Imaging, p.73-80. https://doi.org/10.2312/COMPAESTH/COMPAESTH07/073-080

Rasche, K., Geist, R., Westall, J., 2005. Re-coloring images for gamuts of lower dimension. *Comput. Graph. Forum*, **24**(3):423-432. https://doi.org/10.1111/j.1467-8659.2005.00867.x

Smith, K., Landes, P.E., Thollot, J., *et al.*, 2008. Apparent greyscale: a simple and fast conversion to perceptually accurate images and video. *Comput. Graph. Forum*, **27**(2):193-200. https://doi.org/10.1111/j.1467-8659.2008.01116.x

Song, Y., Bao, L., Xu, X., *et al.*, 2013. Decolorization: is rgb2gray() out? ACM SIGGRAPH Asia Technical Briefs, Article 15. https://doi.org/10.1145/2542355.2542374

Wang, J.G., Yau, W.Y., 2014. Real-time moustache detection by combining image decolorization and texture detection with applications to facial gender recognition. *Mach. Vis. Appl.*, **25**(4):1089-1099. https://doi.org/10.1007/s00138-014-0597-2

Wang, Z., Bovik, A.C., Sheikh, H.R., *et al.*, 2004. Image quality assessment: from error visibility to structural similarity. *IEEE Trans. Imag. Process.*, **13**(4):600-612. https://doi.org/10.1109/TIP.2003.819861