

PRINCIPAL NOISELESS COLOR COMPONENT EXTRACTION BY LINEAR COLOR COMPOSITION WITH OPTIMAL COEFFICIENTS

Takuya Sugimoto, Kazuhiro Fujimori, Keiichiro Shirai, Hidetoshi Miyao, Minoru Maruyama

Faculty of Engineering, Shinshu University, Japan

ABSTRACT

In this paper, we propose a principal color component extraction method that is simply performed by linear color composition (transformation) of R, G, B colors, but its composite coefficients are calculated so as to obtain a noisy-texture-less principal component of RGB color images. Our method is related to *principal component analysis* (PCA) and edge preserving smoothing by *total variation* (TV) minimization. The resultant image becomes a principal color component image with the minimum total variation. We show this problem can be formulated as TV minimization on a spherical manifold for a whitened data matrix. Although this spherical constraint is non-convex, it can be solved by using *alternating direction method of multipliers* (ADMM). As its application, we show the results of text character extraction from ancient wooden tablets, and how our method extracts faint ink characters while reducing wood grain textures. Our method is unsupervised but has performance equivalent to a *linear discriminant analysis* (LDA) method with user-assisted information.

Index Terms— color transformation, component extraction, PCA, total variation, defect detection

1. INTRODUCTION

The proposal method has been mainly developed for a *character extraction* method for ancient wooden tablets [1, 2] as shown in Fig. 1, but is also available for the detection of *stain flaw defect* and *coating defect* based on low contrast color change as shown in Fig. 7.

In the case of wooden tablets in Fig. 1(a) and (d), one can see the images consist of faint characters and noisy wood grain textures. Since the luminance intensity of faint characters is usually weaker than that of wood grains (b), we cannot apply smoothing filters to remove the noisy textures. In such a case, taking account of color differences of R, G, B colors, *i.e.*, chrominance components, is useful and has a possibility to remove unnecessary textures (c). It is because the black ink of characters tends to absorb infrared and red spectrum, and the difference of light absorption can be observed by subtracting B from R component: $R - B$. However, in some tablets, the luminance component, $\frac{1}{3}(R + G + B)$, gives a preferable result (e). Therefore, we would like to choose the appropriate color composition way or obtain more appropriate composite coefficients than the above $\{1, 0, -1\}$ and $\{\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\}$ for $\{R, G, B\}$, which can extract a principal color component without noisy texture as shown in (c) and (e).

As a conventional method for extracting the principal components, *principal components analysis* (PCA) [3] for RGB color distribution is available. In an RGB color distribution of pixel colors, the variation along each principal axis usually corresponds to luminance and chrominance components. Note that the variation is calculated by inner product calculation of an axis (unit vector

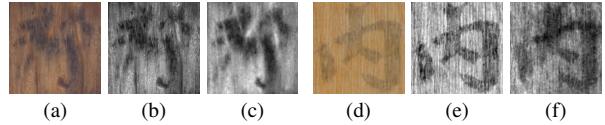


Fig. 1. Results of linear color composition with different sets of composite coefficients. From the left, (a) (d) original images, (b) (e) luminance components by $\frac{1}{3}(R + G + B)$, (c) (f) chrominance components by $R - B$. The intensity of resultant grayscale images is normalized and linearly tone mapped within $[0,1]$. Depending on the coefficients, image noise and wood grains are reduced or emphasized. Preferable results for character recognition are (c) and (e).

$[\alpha_1, \alpha_2, \alpha_3]^T$) and color coordinates (R, G, B) as $\alpha_1R + \alpha_2G + \alpha_3B$, and the vector corresponds to the composite coefficients $\{\alpha_1, \alpha_2, \alpha_3\}$. Therefore, by analyzing the color axes by PCA, and obtaining the principal components in each axis direction, we can narrow down the candidate set of coefficients. In addition to this, if we can give an additional condition to reduce noisy textures, PCA might decide the most preferable composite coefficients.

As an image feature of perceptually clear and noiseless images, many regions have their own color and inside pixels have flat intensity values, and steep changes occur around boundaries of regions. In such a case, numerically, the summation of absolute differences of neighboring pixel values, *i.e.*, *total variation* (TV) [4], becomes small. This condition to minimize TV is now widely used in image restoration problems such as denoising and deblurring and so on. Therefore, adding this condition to the aforementioned PCA has a possibility to achieve our purpose.

In this paper, on the basis of the aforementioned PCA and TV, we propose a method that can extract a noiseless principal component of a RGB color image. Its resultant image is represented as a linear composition of R, G, B components, and composite coefficients are calculated by solving a constrained optimization problem. Although the problem comes down to the minimization of *least absolute deviations* (LAD) on a spherical manifold [5], and is a non-convex problem, we show it can be solved (converged to the global solution) by using *alternating direction methods of multiplier* (ADMM) for convex optimization [6–9].

Note that our method is an unsupervised method, which does not require user assisted information to separate foreground and background regions such as used in *linear discriminant analysis* (LDA) and *image matting*. We show that our method gives performance equivalent to the LDA method with user-assisted information.

2. PREPARATION AND RELATED METHODS

This section describes the data representation of images and a basic method using PCA.

2.1. Color component extraction by linear composition of R, G, B components, and normalization of image contrast

Input images are RGB color (or vector valued) images, each color values at a pixel i is denoted as (r_i, g_i, b_i) , and for more simplicity, the whole pixel colors are dealt with vectors as $\mathbf{r} := [r_1, \dots, r_N]^\top \in \mathbb{R}^N$, and \mathbf{g} , \mathbf{b} as well, where N is the number of pixels. By using the vectors and composition (transformation) coefficients $\{\alpha_1, \alpha_2, \alpha_3\} \in \mathbb{R}^3$, linear composition to obtain luminance and chrominance components is represented as

$$\mathbf{x} := \alpha_1 \mathbf{r} + \alpha_2 \mathbf{g} + \alpha_3 \mathbf{b} = \mathbf{Y}\boldsymbol{\alpha}, \quad (1)$$

where we introduce the matrix form expression using a data matrix $\mathbf{Y} := [\mathbf{r}, \mathbf{g}, \mathbf{b}] \in \mathbb{R}^{N \times 3}$ and a coefficient vector $\boldsymbol{\alpha} := [\alpha_1, \alpha_2, \alpha_3]^\top$. In the above equation, when $\boldsymbol{\alpha} = [\frac{1}{3}, \frac{1}{3}, \frac{1}{3}]^\top$, the resultant \mathbf{x} becomes the luminance component, while $\boldsymbol{\alpha} = [1, 0, -1]^\top$ gives a chrominance component (chroma-orange (Co) of YCoCg color transformation [10]).

The intensity range and contrast of a composed image \mathbf{x} is changed depending on the coefficients, *e.g.*, the contrast of chrominance generally becomes lower than that of luminance. Therefore, its normalization is required, especially followed by a recognition process. In this paper, the normalization is performed by subtracting the mean value $\mu(\mathbf{x}) := \frac{1}{N} \sum_i x_i$ from \mathbf{x} , and then dividing it by the standard deviation $\sigma(\mathbf{x}) := \frac{1}{N} \sum_i (x_i - \mu(\mathbf{x}))^2$:

$$\hat{\mathbf{x}} := \frac{1}{\sigma(\mathbf{x} - \mu(\mathbf{x}))} (\mathbf{x} - \mu(\mathbf{x})) = \frac{1}{\sqrt{\frac{1}{N} \|\mathbf{M}\mathbf{x}\|_2^2}} \mathbf{M}\mathbf{x}, \quad (2)$$

where we introduce the matrix form, $\mathbf{M} := \mathbf{Id} - \frac{1}{N} \mathbf{1}\mathbf{1}^\top \in \mathbb{R}^{N \times N}$ is a centering matrix, \mathbf{Id} is an identity matrix, $\mathbf{1} \in \{1\}^N$ is a vector of ones, and $\|\mathbf{x}\|_p := (\sum_i |x_i|^p)^{1/p}$ is the ℓ_p norm.

2.2. Principal axes analysis of color distribution by PCA

The aforementioned typical composite coefficients generally correspond to vectors of principal axes of color distribution. That is, the equation (1) can be regarded as inner product calculation of color coordinates (r_i, g_i, b_i) and a vector representing an axis $[\alpha_1, \alpha_2, \alpha_3]^\top$, and the obtained value indicates variation along the axis.

To analyze the principal axes of data distribution, *principal component analysis* (PCA) [3] is often used. First, the covariance matrix $\mathbf{C} \in \mathbb{R}^{3 \times 3}$ of the data matrix is computed. Then its eigendecomposition is calculated as

$$\mathbf{C} := \frac{1}{N} \bar{\mathbf{Y}}^\top \bar{\mathbf{Y}} = \mathbf{U}\Lambda\mathbf{U}^\top, \quad (3)$$

where $\bar{\mathbf{Y}} := \mathbf{MY}$ is a mean-centered data matrix, *i.e.*, from each column, its mean value is subtracted. By eigendecomposition, a set of orthogonal eigenvectors $\mathbf{U} := [\mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3] \in \mathbb{R}^{3 \times 3}$ and eigenvalues $\Lambda := \text{diag}(\lambda_1, \lambda_2, \lambda_3) \in \mathbb{R}^{3 \times 3}$ where $\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq 0$ are obtained. The unit eigenvector \mathbf{u}_i corresponds to the i -th principal axis and the eigenvalue λ_i indicates the variance of distribution. The inner product of an eigenvector and RGB colors extracts variations along the principal axis as $\mathbf{x}_i := \bar{\mathbf{Y}}\mathbf{u}_i$.

This PCA based method can be written as an optimization problem known as the *maximization of Rayleigh quotient* [11], and the extraction of the first principal vector is defined as

$$\max_{\boldsymbol{\alpha} \in \{\boldsymbol{\alpha}: \|\boldsymbol{\alpha}\|_2=1\}} \left\{ \frac{1}{N} \|\bar{\mathbf{Y}}\boldsymbol{\alpha}\|_2^2 = \boldsymbol{\alpha}^\top \mathbf{U}\Lambda\mathbf{U}^\top \boldsymbol{\alpha} \right\}, \quad (4)$$

where the constraint $\|\boldsymbol{\alpha}\|_2 = 1$ is added to give a unit vector. The solution that maximize the $\boldsymbol{\alpha}^\top \mathbf{U}\Lambda\mathbf{U}^\top \boldsymbol{\alpha}$ is given by $\boldsymbol{\alpha} = \mathbf{u}_1$ and the

maximum value becomes λ_1 since $\mathbf{U}^\top \mathbf{u}_1 = [1, 0, 0]^\top$ with orthogonality. Additionally, when using *min* instead of *max*, the solution becomes $\boldsymbol{\alpha} = \mathbf{u}_3$. In this way, PCA is related to the above formulation.

3. PROPOSED METHOD

In addition to (1) and (2), we would like to obtain $\boldsymbol{\alpha}$ that minimizes *total variation* (TV) of the resultant image $\hat{\mathbf{x}}$. Therefore, we formulate our method as

$$\min_{\boldsymbol{\alpha} \in \{\boldsymbol{\alpha}: \|\boldsymbol{\alpha}\|_2=1\}} \left\{ \|\hat{\mathbf{x}}\|_{\text{TV}} := \left\| \frac{1}{\sqrt{\frac{1}{N} \|\bar{\mathbf{Y}}\boldsymbol{\alpha}\|_2^2}} \bar{\mathbf{Y}}\boldsymbol{\alpha} \right\|_{\text{TV}} \right\}, \quad (5)$$

where $\|\cdot\|_{\text{TV}}$ is a TV norm described later. Then, $\boldsymbol{\alpha}$ is actually scale invariant since, in $\frac{\bar{\mathbf{Y}}s\boldsymbol{\alpha}}{\sqrt{\|\bar{\mathbf{Y}}s\boldsymbol{\alpha}\|_2^2}} = \frac{\bar{\mathbf{Y}}\boldsymbol{\alpha}}{\sqrt{\|\bar{\mathbf{Y}}\boldsymbol{\alpha}\|_2^2}}$, the scaling factor s is canceled. Therefore we add the following constraint (manifold) $\mathcal{A} \ni \boldsymbol{\alpha}$ so as to make $\frac{1}{\sqrt{\frac{1}{N} \|\bar{\mathbf{Y}}\boldsymbol{\alpha}\|_2^2}} = \text{const.}$ Using this constraint, the above equation can be rewritten as

$$\min_{\boldsymbol{\alpha} \in \mathcal{A}} \|\bar{\mathbf{Y}}\boldsymbol{\alpha}\|_{\text{TV}}, \quad \mathcal{A} := \{\boldsymbol{\alpha} : \|\bar{\mathbf{Y}}\boldsymbol{\alpha}\|_2 = 1, \mathbf{1}^\top \boldsymbol{\alpha} \geq 0\}, \quad (6)$$

where $\mathbf{1}^\top \boldsymbol{\alpha} \geq 0$ is added to remove the inverse signed solution. When using this formulation alone, it can be further simplified by using a whitened data matrix as shown in the next 3.1 and (11).

As for the TV norm, for notational simplicity, we represent it as the following anisotropic type TV with use of differential filter matrices and ℓ_1 norm:

$$\|\hat{\mathbf{x}}\|_{\text{TV}} := \|\mathbf{D}\hat{\mathbf{x}}\|_1 = \|\mathbf{D}_h\hat{\mathbf{x}}\|_1 + \|\mathbf{D}_v\hat{\mathbf{x}}\|_1, \quad (7)$$

where $\mathbf{D} := [\mathbf{D}_h^\top, \mathbf{D}_v^\top]^\top \in \mathbb{R}^{2N \times 3}$ is a set of differential filter matrices for the horizontal and the vertical direction (circular 2-tap forward (or backward) differential filter with coefficients $[-1, 1]$).

As a reference, Fig. 2 shows solutions and cost maps (of the later (11)) obtained by exhaustive search. One can see each cost map has a unique minimum point and smooth gradient. We found a lot of images, even colorful images, have this tendency.

3.1. Simplification by data matrix whitening

The above problem can be rewritten into a simple form by whitening the data matrix. In (5), the covariance matrix at the denominator is denoted as follows using notations in (3) and (4):

$$\frac{1}{N} \|\bar{\mathbf{Y}}\boldsymbol{\alpha}\|_2^2 = (\boldsymbol{\alpha}^\top \mathbf{U}\Lambda^{\frac{1}{2}})(\Lambda^{\frac{1}{2}} \mathbf{U}^\top \boldsymbol{\alpha}) = \hat{\boldsymbol{\alpha}}^\top \hat{\boldsymbol{\alpha}}, \quad (8)$$

where the newly introduced $\hat{\boldsymbol{\alpha}} \in \mathbb{R}^3$ has the following relationship¹: $\hat{\boldsymbol{\alpha}} = \Lambda^{\frac{1}{2}} \mathbf{U}^\top \boldsymbol{\alpha}$ and $\boldsymbol{\alpha} = \mathbf{U}\Lambda^{-\frac{1}{2}} \hat{\boldsymbol{\alpha}}$. Substituting it into (5), we can rewrite $\hat{\mathbf{x}}$ as

$$\hat{\mathbf{x}} = \frac{1}{\|\hat{\boldsymbol{\alpha}}\|_2} \bar{\mathbf{Y}}\mathbf{U}\Lambda^{-\frac{1}{2}} \hat{\boldsymbol{\alpha}} = \frac{1}{\|\hat{\boldsymbol{\alpha}}\|_2} \hat{\mathbf{Y}}\hat{\boldsymbol{\alpha}}, \quad (9)$$

where $\hat{\mathbf{Y}} := \bar{\mathbf{Y}}\mathbf{U}\Lambda^{-1/2} \in \mathbb{R}^{N \times 3}$ indicates the whitened data matrix whose covariance matrix becomes an identity matrix.

Using the above relationship, we can rewrite our main Eq. (6) as

$$\min_{\hat{\boldsymbol{\alpha}} \in \hat{\mathcal{A}}} \|\hat{\mathbf{Y}}\hat{\boldsymbol{\alpha}}\|_1, \quad \hat{\mathcal{A}} := \{\hat{\boldsymbol{\alpha}} : \|\hat{\boldsymbol{\alpha}}\|_2 = 1, \mathbf{1}^\top \mathbf{U}\Lambda^{-\frac{1}{2}} \hat{\boldsymbol{\alpha}} \geq 0\}. \quad (10)$$

Furthermore, by precomputing differential filtering and denoting $\hat{\mathbf{Y}}' := [(\mathbf{D}_h \hat{\mathbf{Y}})^\top, (\mathbf{D}_v \hat{\mathbf{Y}})^\top]^\top \in \mathbb{R}^{2N \times 3}$ and $\mathbf{l}' := \Lambda^{-\frac{1}{2}} \mathbf{U}^\top \mathbf{l} \in \mathbb{R}^3$, we get

$$\min_{\hat{\boldsymbol{\alpha}} \in \hat{\mathcal{A}}} \|\hat{\mathbf{Y}}'\hat{\boldsymbol{\alpha}}\|_1, \quad \hat{\mathcal{A}} := \{\hat{\boldsymbol{\alpha}} : \|\hat{\boldsymbol{\alpha}}\|_2 = 1, \mathbf{l}'^\top \hat{\boldsymbol{\alpha}} \geq 0\}. \quad (11)$$

¹ $\bar{\mathbf{Y}}$ is rarely rank deficient and Λ has zeros on the diagonal. Therefore we set a small infimum value to avoid division-by-zero in Λ^{-1} .

3.2. Solution of (11) using ADMM

The problem in (11) corresponds to *least absolute derivations* (LAD) on a spherical compact manifold [5, 12], or ℓ_1 -norm principal component analysis (ℓ_1 -PCA) [13, 14]. Here, we deal with this problem as the former (minimization problem), and solve it by using *alternating direction method of multipliers* (ADMM) [6–9] as shown in [12, 15].

The equation (11) can be rewritten as a regularized form using the indicator function: $\min_{\hat{\alpha} \in \hat{\mathcal{A}}} \|\hat{\mathbf{Y}}' \hat{\alpha}\|_1 + \iota_{\hat{\mathcal{A}}}(\hat{\alpha})$, where the constraint $\mathbf{l}'^\top \hat{\alpha} \geq 1$ is once removed and finally applied, and the indicator function $\iota_C(\cdot)$ is defined as $\iota_C(\hat{\alpha}) = \begin{cases} 0 & \hat{\alpha} \in C, \\ \infty & \text{otherwise.} \end{cases}$. Then, we replace the variables for $\hat{\mathbf{Y}}' \hat{\alpha} \rightarrow \mathbf{z}_1 \in \mathbb{R}^{2N}$ and $\hat{\alpha} \rightarrow \mathbf{z}_2 \in \mathbb{R}^3$ as $\min_{\hat{\alpha} \in \hat{\mathcal{A}}} \|\mathbf{z}_1\|_1 + \iota_{\hat{\mathcal{A}}}(\mathbf{z}_2)$ s.t. $\hat{\mathbf{Y}}' \hat{\alpha} - \mathbf{z}_1 = \mathbf{0}$, $\hat{\alpha} - \mathbf{z}_2 = \mathbf{0}$, and define its augmented Lagrangian function by introducing dual variable $\mathbf{u}_1 \in \mathbb{R}^{2N}$ and $\mathbf{u}_2 \in \mathbb{R}^3$ as

$$L(\hat{\alpha}, \mathbf{z}_1, \mathbf{z}_2, \mathbf{u}_1, \mathbf{u}_2) := \|\mathbf{z}_1\|_1 + \frac{\rho_1}{2} \|\hat{\mathbf{Y}}' \hat{\alpha} - \mathbf{z}_1 + \mathbf{u}_1\|_2^2 + \iota_{\hat{\mathcal{A}}}(\mathbf{z}_2) + \frac{\rho_2}{2} \|\hat{\alpha} - \mathbf{z}_2 + \mathbf{u}_2\|_2^2, \quad (12)$$

where parameters $\rho_1 > 0$ and $\rho_2 > 0$ control convergence, and ρ_2 requires a large value related to the number of pixels to balance with the ℓ_2 norm of ρ_1 . By using this function, calculation at the t -th iteration of ADMM can be derived as follows:

$$\left| \begin{array}{l} \mathbf{z}_1^{t+1} := \arg \min_{\mathbf{z}_1} L(\hat{\alpha}^t, \mathbf{z}_1, \mathbf{u}_1^t) = \mathcal{S}_{1/\rho_1}(\hat{\mathbf{Y}}' \hat{\alpha}^t + \mathbf{u}_1^t), \\ \mathbf{z}_2^{t+1} := \arg \min_{\mathbf{z}_2} L(\hat{\alpha}^t, \mathbf{z}_2, \mathbf{u}_2^t) = \Pi_{\hat{\mathcal{A}}}(\hat{\alpha}^t + \mathbf{u}_2^t), \\ \mathbf{u}_1^{t+1} := \arg \min_{\mathbf{u}_1} L(\hat{\alpha}^t, \mathbf{z}_1^{t+1}, \mathbf{u}_1) = \mathbf{u}_1^t + \hat{\mathbf{Y}}' \hat{\alpha}^t - \mathbf{z}_1^{t+1}, \\ \mathbf{u}_2^{t+1} := \arg \min_{\mathbf{u}_2} L(\hat{\alpha}^t, \mathbf{z}_2^{t+1}, \mathbf{u}_2) = \mathbf{u}_2^t + \hat{\alpha}^t - \mathbf{z}_2^{t+1}, \\ \hat{\alpha}^{t+1} := \arg \min_{\hat{\alpha}} L(\hat{\alpha}, \mathbf{z}_1^t, \mathbf{z}_2^t, \mathbf{u}_1^{t+1}, \mathbf{u}_2^{t+1}) \\ \quad = (\rho_1 \hat{\mathbf{Y}}'^\top \hat{\mathbf{Y}}' + \rho_2 \mathbf{Id})^{-1} (\rho_1 \hat{\mathbf{Y}}'^\top (\mathbf{z}_1^{t+1} - \mathbf{u}_1^{t+1}) + \rho_2 (\mathbf{z}_2^{t+1} - \mathbf{u}_2^{t+1})), \end{array} \right. \quad (13)$$

where \mathbf{v}_1 and \mathbf{v}_2 are temporal vectors, and the algorithm order is changed to avoid obtaining $\hat{\alpha}^{t=1} = \mathbf{0}$ when using zero initialized vectors $\mathbf{u}_1^0, \mathbf{u}_2^0, \mathbf{z}_1^0, \mathbf{z}_2^0$. The other initial value is $\hat{\alpha}^0 \neq \mathbf{0}$, $\hat{\alpha}^0 \leftarrow \frac{\hat{\alpha}^0}{\|\hat{\alpha}^0\|_2}$. As for the functions, $\mathcal{S}_{1/\rho_1}(\cdot)$ denotes soft-thresholding:

$$[\mathcal{S}_{1/\rho_1}(\mathbf{x})]_i := x_i \left(1 - 1 / \max(1, \rho_1 |x_i|) \right), \quad (14)$$

where $[\cdot]_i$ denotes element-wise operation, $\Pi_{\hat{\mathcal{A}}}(\cdot)$ denotes projection onto the spherical manifold $\hat{\mathcal{A}}$:

$$[\Pi_{\hat{\mathcal{A}}}(\mathbf{x})]_i := x_i / \|\mathbf{x}\|_2, \quad (15)$$

and the inverse matrix $(\cdot)^{-1} \in \mathbb{R}^{3 \times 3}$ is pre-computable since it is constant in the loop. Finally, the optimal unique solution and the resultant composite image are given by

$$\hat{\alpha}^* := \text{sign}(\mathbf{l}'^\top \hat{\alpha}) \cdot \hat{\alpha}, \quad \hat{\mathbf{x}}^* := \hat{\mathbf{Y}} \hat{\alpha}^*. \quad (16)$$

4. EXPERIMENTAL RESULTS

We show some experimental results using images of wooden tablet database [2] (of size $200h \times 200w \sim 500h \times 500w$) and test images used in stain detection. The intensity range of images are normalized within $[0, 1]$.

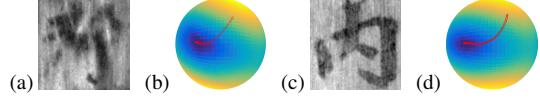


Fig. 2. Optimum solutions of images in Fig. 1. (a)(c) Composition results using optimum sets of coefficients. (b)(d) Cost maps on spherical 3D coordinates $(\hat{\alpha}_r, \hat{\alpha}_g, \hat{\alpha}_b)$ where $\|\hat{\alpha}\|_2 = 1$, and the axes denote (depth, horizontal, vertical) direction. The red dots on the cost maps indicate the trajectory of convergence.

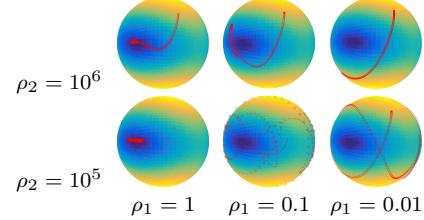


Fig. 3. Parameter setting and trajectories of convergence.

Evaluation of convergence: Figure 2 shows the results of Fig. 1. The parameters of ADMM are set to $\rho_1 = 10$ and $\rho_2 = 10^6$, and the number of iterations is 200. The cost maps (b) and (d) are calculated by exhaustive search, and the dark blue region has the minimum value. The red dots indicates the trajectory of convergence by ADMM, and the initial solution $\hat{\alpha}$ is set at a point distant from the global minimum. One can see that the trajectories correctly converge to the minimum point, and as its resultant image, preferable images shown in Fig. 1(c) and (e) are obtained.

As for the ADMM parameters ρ_1 and ρ_2 that control the convergence, it seems to decide carefully than cases of convex optimization. Figure 3 shows the results using some combinations of parameters. Iteration with large step size tends not to converge to the global minimum. In the case of wooden tablet images, the best combination is $\rho_1 = 10$ and $\rho_2 = 10^6$ that is used in Fig. 2.

Comparison with conventional methods: Figure 4 shows comparison with representative color transformation and decomposition methods. For visibility, the intensity range of each image [min, max] is linearly normalized to $[0, 1]$, and the sign of intensity values is inverted so that the character region color becomes similar. The result of PCA is the version without whitening since whitening does not affect differences when using normalized intensity range. The result of ICA² becomes similar to that of PCA. One can see that the character is well extracted in results of our method, the Cr component $\alpha \approx [1, -1, 0]^\top$, and the Co component $\alpha = [1, 0, -1]^\top$.

Figure 5 shows comparison with a *linear discriminant analysis* (LDA) method [17] with user-assisted information³ as shown in (a-1). In the tri-color map (a-1), each cluster color of the foreground and the back-ground is specified with the black and the white color. The LDA calculates the covariance of each cluster and decides the cluster of each pixel. In the LDA, good resultant images are obtained as shown in (a-2). However, to obtain those preferable character shapes, one has to carefully chose pixels having doubtless cluster colors as shown in (a-1) for each input image. On the other hand, our method gives equivalent results (b-2) in combination with simple Niblack's binarization method [18] for (b-1). Bur-

²As for the ICA, we use a public source code distributed by B. Moore on MATLAB central with a title “PCA and ICA Package”.

³not a supervised LDA (latent Dirichlet allocation).

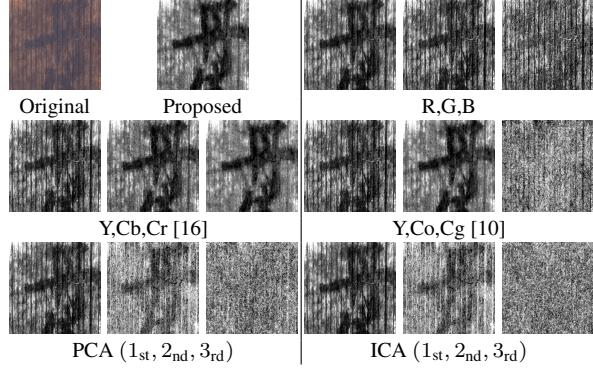


Fig. 4. Comparison with representative color transformation and decomposition methods. Intensity is normalized for displaying.

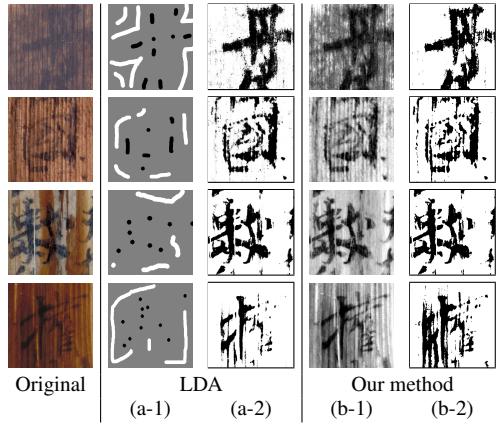


Fig. 5. Comparison with a LDA method [17] with user-assisted information. (a-1) is user assisted information for the LDA and (a-2) is the result. (b-1) is our result and (b-2) is the binarized image processed by Niblack's binarization [18].

dense pixel color selection is not required in our method. The binarized pixel value becomes black if the pixel value x_i is $x_i \leq \text{mean}(\{x_i\}) - \gamma \text{std}(\{x_i\})$, where the functions are defined to give the global mean value and the standard deviation of all the pixel values, and $\gamma = 0.7$ in this paper. The bottom row shows a difficult case for both methods. The difficulty mainly comes from the horizontal gradation of luminance. In the case of our method, since the character shape in (b-1) is perceptually visible, the result may improve by replacing the binarization method.

Figure 6 shows comparison with other standard methods. The results are obtained by (a) learning based matting [19] using the user assisted map and binarization in Fig. 5, (b) ℓ_0 smoothing [20], and (c) is MSER based segmentation [21] often used for text extraction. Note that (b) and (c) are applied to the Cr component of YCbCr. The ℓ_0 smoothing method gives better results, but they cannot outperform the performance of LDA and our method.

Application of stain detection: As another application, we show the results of stain detection in Fig. 7. The purpose is to detect coffee stains on the cloth. (a) is the result of LDA for reference, where the top right image in (a) is the user assisted information. Then, (b-1) and (b-2) are results of PCA. Although the PCA gave the perceptually good result (b-1), it failed in binarization because of luminance unevenness arising over the image (b-2). Finally, (c-1)

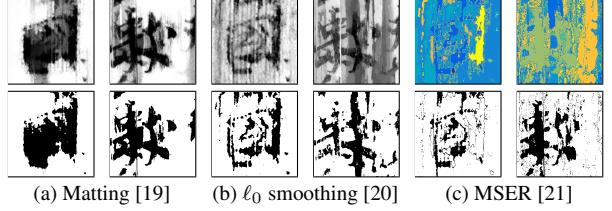


Fig. 6. Comparison with standard methods. The bottom row shows results of binarization.

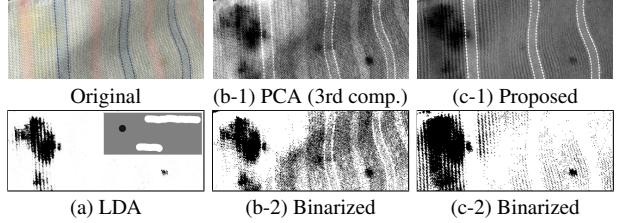


Fig. 7. Stain detection: coffee stain on cloth. (b-2) and (c-2) are binarized in the same way as Fig. 5.

and (c-2) are results of our method. Since the method generated the flat (variation minimized) background region (c-1), the binarization successfully extracted the stain regions (c-2). In our method, the red lines tended to be wrongly extracted as the principle structure component. To deal with this problem, orthogonalization techniques, such as the Gram-Schmitt method, to extract the 2nd or 3rd principal component, which correspond to the color of the stain, might be required.

Execution time: The execution time of our method is proportional to the number of pixels. The time at one iteration of ADMM is about 30 (msec/Mpixs) on our PC environment: Intel Core i7@2.7GHz; implemented on C++ and MATLAB. For example, the image in Fig. 2(c) of size 200h \times 180w requires 100 iterations until convergence (start from the distant point), and it takes 70 (msec). This execution time mainly depends on the multiplication of linear operators \hat{Y}' and \hat{Y}'^\top in (13), and they can be accelerated by parallel computing.

5. CONCLUSION

In this paper, we propose a color component extraction method that is regarded as a modified PCA in combination with TV minimization. The formulation comes down to the minimization of LAD problem on a spherical compact manifold, and ADMM is used to solve it. This type formulation is attracting attention, and seems to be used in non-convex optimization, especially in binary quadratic programming. Considering that resultant images (objects) are finally binarized to extract their shapes, the formulation described in this paper may be applied to such a binary programming method to directly obtain binary images with clear shapes.

Acknowledgement

This work was supported by Nara National Res. Inst. for Cultural Properties in Japan and the Grant-in-Aid for Sci. Res. (S)-20222002. The authors are grateful to the authors of [1], and Dr. Masahiro Okuda (Univ. of Kitakyushu) and Dr. Shunsuke Ono (Tokyo Inst. of Tech.) for fruitful discussions about our method.

6. REFERENCES

- [1] A. Kitadai, K. Shirai, E. Yuki, M. Nakagawa, H. Baba, A. Watanabe, S. Watanabe, and N. Kurushima, “Adaptation of normalization process to historical mokkan images and update of support system to support reading historical mokkans (in japanese),” in *JINMONKON*, 2013, pp. 65–70.
- [2] H. Baba and A. Watanabe, “Ancient japanese wooden tablets (mokkan) image database,” <http://jiten.nabunken.go.jp/index.html>.
- [3] I.T. Jolliffe, *Principal Component Analysis*, Springer Series in Statistics. Springer, 2002.
- [4] L.I. Rudin, S. Osher, and E. Fatemi, “Nonlinear total variation based noise removal algorithms,” *Physica D*, vol. 60, no. 1–4, pp. 259–268, 1992.
- [5] Z. Wen and W. Yin, “A feasible method for optimization with orthogonality constraints,” *Math. Program.*, vol. 142, no. 1, pp. 397–434, 2013.
- [6] D. Gabay and B. Mercier, “A dual algorithm for the solution of nonlinear variational problems via finite elements approximations,” *Computers & Math. with Appl.*, vol. 2, no. 1, pp. 17–40, 1976.
- [7] J. Eckstein and D.P. Bertsekas, “On the Douglas-Rachford splitting method and proximal point algorithm for maximal monotone operators,” *Math. Program.*, vol. 55, pp. 293–318, 1992.
- [8] T. Goldstein and S. Osher, “The split Bregman method for L1-regularized problems,” *SIAM J. Imag. Sci.*, vol. 2, pp. 323–343, 2009.
- [9] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, “Distributed optimization and statistical learning via the alternating direction method of multipliers,” *Found. Trends Mach. Learn.*, vol. 3, no. 1, 2011.
- [10] R. Malvar, G.J. Sullivan, and S. Srinivasan, “Lifting-based reversible color transformations for image compression,” in *SPIE Apps. Digital Image Process.*, 2008, pp. 1–10.
- [11] B.N. Parlett, *The symmetric eigenvalue problem*, vol. 20 of *Classics in Applied Mathematics*, SIAM, 1998.
- [12] Y. Wang, W. Yin, and J. Zeng, “Global convergence of ADMM in nonconvex nonsmooth optimization,” Tech. Rep., Xi’an Jiaotong Univ., 2015.
- [13] N. Kwak, “Principal component analysis based on l1-norm maximization,” *IEEE Trans. Pattern Anal. Machine Intelli. (TPAMI)*, vol. 30, no. 9, pp. 1672–1680, 2008.
- [14] P.P. Markopoulos, G.N. Karystinos, and D.A. Pados, “Optimal algorithms for L1-subspace signal processing,” *IEEE Trans. Signal Process. (TSP)*, vol. 62, no. 19, pp. 5046–5058, 2014.
- [15] B. Wu and B. Ghanem, “ ℓ_p -box ADMM: A versatile framework for integer programming,” arXiv.
- [16] C.A. Poynton, *Technical introduction to digital video*, John Wiley & Sons, 1996.
- [17] R.A. Fisher, “The use of multiple measurements in taxonomic problems,” *Annals. of Human Genetics*, vol. 7, no. 2, pp. 179–188, 1936.
- [18] W. Niblack, *An introduction to digital image processing*, Prentice Hall, 1986.
- [19] Y. Zheng and C. Kambhamettu, “Learning based digital matting,” in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, 2009, pp. 889–896.
- [20] L. Xu, C. Lu, Y. Xu, and J. Jia, “Image smoothing via ℓ_0 gradient minimization,” *ACM Trans. Graph. (TOG)*, vol. 30, no. 5, pp. 174:1–174:12, 2011.
- [21] H. Chen, S.S. Tsai, G. Schroth, D.M. Chen, R. Grzeszczuk, and B. Girod, “Robust text detection in natural images with edge-enhanced maximally stable extremal regions,” in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, 2011, pp. 2609–2612.