# IMAGE COLORIZATION ALGORITHM USING SERIES APPROXIMATED SPARSE FUNCTION

*Kazunori Uruma*[1], *Katsumi Konishi*[2], *Tomohiro Takahashi*[1] *and Toshihiro Furukawa*[1]

[1]Graduate School of Engineering, Tokyo University of Science, Japan
[2]Department of Computer Science, Kogakuin University, Japan
email: uru-kaz@ms.kagu.tus.ac.jp

## ABSTRACT

This paper deals with an image colorization and proposes a new image colorization algorithm. Assuming that the difference of color values between neighbor pixels is given as a monotonically increasing function of the difference of grayscale values between neighbor pixels, a colorization function is proposed, and the colorization problem is formulated as the weighted least squares problem using this function. In order to reduce the dependence on the value of a parameter in the algorithm, this paper utilizes a finite series approximation and provides a fast colorization algorithm. Numerical examples show that the proposed algorithm colorizes a grayscale image efficiently.

***Index Terms—*** image colorization, sparse optimization, series approximation

## 1. INTRODUCTION

This paper deals with a digital image colorization problem, which is to recover a color image from a grayscale image with a few color pixels. For this problem, various algorithms have been proposed [1–6]. In [1–3], colorization algorithms are proposed using the segmentation technique. Because the performance of segmentation technique depends heavily on the property of each image, these algorithms can colorize only particular images. For example, [3] considers only the manga colorization. In [4], Levin et al. propose the colorization algorithm using optimization. This algorithm can colorize any general images if the given image have enough color pixels, however, the algorithm fails to colorize a whole image appropriately if only a few color pixels are given. In [5], the algorithm to find the best pixels to be colorized is proposed to obtain a colorized image with a minimum number of given color pixels.

This paper proposes a new colorization algorithm based on [6], where a colorization algorithm is proposed to colorize a whole image by only a few color pixels under the assumption that each color value changes smoothly if the grayscale intensity value changes smoothly, that is, the neighborhood pixels have the same color when they have equal grayscale intensity values. The colorization problem have been formulated as a mixed $\ell_0/\ell_1$ norm minimization problem, and the algorithm is proposed based on the iterative reweighed least squares [7, 8]. While this algorithm can colorize any general image even if given image have only a few color pixels, the quality of the resulting color image depends much on some parameters, and the algorithm requires a large number of iterations. Therefore this paper modifies this algorithm and provides a new algorithm in order to reduce the dependence of parameter and the calculation cost.

This paper assumes that the difference of color values between neighbor pixels is given as a function of the difference of grayscale values between neighbor pixels and proposes a new colorization algorithm introducing this colorization function, which is derived from a sparse optimization. In order to reduce the dependence on the parameter, this paper also proposes a finite series approximation of the colorization function. Numerical examples show that the proposed algorithm colorizes a grayscale image with only a few color pixels efficiently comparing with some previous studies.

## 2. MAIN RESULTS

### 2.1. Problem Formulation

This paper deals with the digital image colorization problem where a color image is recovered from a grayscale image with a small number of given color pixels. Let $I \in \boldsymbol{R}^{M\times N}$, $I^R \in \boldsymbol{R}^{M\times N}$, $I^G \in \boldsymbol{R}^{M\times N}$ and $I^B \in \boldsymbol{R}^{M\times N}$ denote a grayscale intensity image and its values of red, green and blue, respectively. This paper assumes here that a grayscale image is converted from a color image by forming a weighted sum of the values of red, green and blue as follows,

$$I = a_r I^R + a_g I^G + a_b I^B, \tag{1}$$

where $a_r$, $a_g$ and $a_b$ are constants. Then the colorization problem considered in this paper is formulated as the following

978-1-4799-2893-4/14/$31.00 ©2014 IEEE

matrix completion problem,

$$
\begin{array}{ll}
\text{find} & X = [X^R \; X^G \; X^B] \in \boldsymbol{R}^{M \times 3N} \\
\text{subject to} & X^R \in \boldsymbol{R}^{M \times N},\; X^G \in \boldsymbol{R}^{M \times N},\; X^B \in \boldsymbol{R}^{M \times N}, \\
& I = a_r X^R + a_g X^G + a_b X^B, \\
& [X^R_{i,j} \; X^G_{i,j} \; X^B_{i,j}] = [I^R_{i,j} \; I^G_{i,j} \; I^B_{i,j}],\; \forall (i,j) \in \mathcal{I},
\end{array}
\tag{2}
$$

where $\mathcal{I}$ denotes a given set of matrix indices, which correspond to known color pixels, $A_{i,j}$ denotes the $(i,j)$-element of the matrix $A$, and $X = [X^R \; X^G \; X^B]$ is a design variable. This problem is obviously ill-posed, and therefore we usually provide the additional assumption that each color value changes smoothly. To achieve this assumption, this paper formulates the $\ell_2$ norm minimization problem for the digital image colorization.

Let $\boldsymbol{x}_R = \mathbf{vec}\left(X^R\right)$, $\boldsymbol{x}_G = \mathbf{vec}\left(X^G\right)$, $\boldsymbol{x}_B = \mathbf{vec}\left(X^B\right)$, $\boldsymbol{v}_R = \mathbf{vec}\left(I^R\right)$, $\boldsymbol{v}_G = \mathbf{vec}\left(I^G\right)$, $\boldsymbol{v}_B = \mathbf{vec}\left(I^B\right)$, $\boldsymbol{v}_I = \mathbf{vec}(I)$, $\boldsymbol{x} = [\boldsymbol{x}_R^T \; \boldsymbol{x}_G^T \; \boldsymbol{x}_B^T]^T$ and $\boldsymbol{v} = [\boldsymbol{v}_R^T \; \boldsymbol{v}_G^T \; \boldsymbol{v}_B^T]^T$, where $\mathbf{vec}$ denotes the function which converts a matrix to a vector by stacking the matrix columns successively. Define $U \in \boldsymbol{R}^{(M-1)\times M}$, $V \in \boldsymbol{R}^{M(N-1)\times MN}$, $\bar{U} \in \boldsymbol{R}^{3N(M-1)\times 3MN}$, $\bar{V} \in \boldsymbol{R}^{3M(N-1)\times 3MN}$, $D \in \boldsymbol{R}^{(6MN-3M-3N)\times 3MN}$ and $C \in \boldsymbol{R}^{MN \times 3MN}$ as

$$
U_{i,j} = \begin{cases} 1, & \text{if } i = j \\ -1, & \text{if } i+1 = j \\ 0, & \text{otherwise} \end{cases},\quad
V_{i,j} = \begin{cases} 1, & \text{if } i = j \\ -1, & \text{if } i+M = j, \\ 0, & \text{otherwise} \end{cases}
$$

$\bar{U} = \mathrm{diag}(U, \ldots, U)$, $\bar{V} = \mathrm{diag}(V, V, V)$, $D = [\bar{U}^T \; \bar{V}^T]^T$ and

$$
C = \begin{cases} a_r, & \text{if } i = j \\ a_g, & \text{if } i + MN = j \\ a_b, & \text{if } i + 2MN = j \\ 0, & \text{otherwise} \end{cases}
$$

respectively, where $\mathrm{diag}(A_1, \ldots, A_m)$ denotes a block diagonal matrix consisting of $A_1, \ldots, A_m$. The vectors $\boldsymbol{v}$ and $\boldsymbol{x}$ correspond to $[I^R \; I^G \; I^B]$ and $X = [X^R \; X^G \; X^B]$, respectively, and the matrices $\bar{U}$ and $\bar{V}$ denote vertical and horizontal difference operators. Then $D\boldsymbol{x}$ denotes the differences between the neighbor pixels of a whole image, and the $\ell_2$ norm minimizing colorization problem is formulated as follows,

$$
\begin{array}{ll}
\text{Minimize} & \|D\boldsymbol{x}\|_2^2 \\
\text{subject to} & \boldsymbol{v}_I = C\boldsymbol{x},\; x_i = v_i,\; \forall i \in \bar{\mathcal{I}},
\end{array}
\tag{3}
$$

where $\|\cdot\|_2$ denotes the $l_2$ norm of a vector, $x_i$ and $v_i$ denote the $i$th element of $\boldsymbol{x}$ and $\boldsymbol{v}$, respectively, and $\bar{\mathcal{I}}$ denotes a given set of vector indices corresponding to $\mathcal{I}$.

Experimental results indicate that the $\ell_2$ norm minimization approach (3) colorizes the pixels near the given color pixels correctly, however the further pixels from the given color pixels are recovered with larger errors or remain grayscale. In order to achieve the correct color recovery, this paper assumes that the difference of color values between neighbor pixels is given as a function of the difference of grayscale values between neighbor pixels as follows,

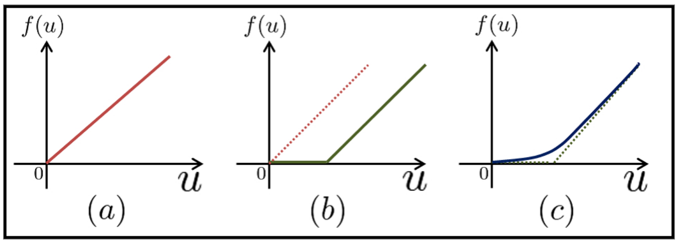


**Fig. 1**. Illustration of the function $f$. (a) (b) and (c) are given by (8), (9) and (11), respectively.

$$
(D\boldsymbol{x})_i = \alpha f((D\tilde{\boldsymbol{v}})_i),
\tag{4}
$$

where $\tilde{\boldsymbol{v}} = [\boldsymbol{v}_I^T \; \boldsymbol{v}_I^T \; \boldsymbol{v}_I^T]^T$, $(\cdot)_i$ denotes the $i$th element of a vector, $\alpha$ is constant, and $f$ is a real valued function. Then we obtain the following equation,

$$
\|D\boldsymbol{x}\|_2^2 = \alpha^2 \sum_{k=1}^{6MN-3M-3N} f((D\tilde{\boldsymbol{v}})_k)^2.
\tag{5}
$$

Because the value of $\sum_{k=1}^{6MN-3M-3N} f((D\tilde{\boldsymbol{v}})_k)^2$ is a constant, we obtain the following equations,

$$
\begin{aligned}
\underset{\boldsymbol{x}, s.t. (4)}{\arg\min} \|D\boldsymbol{x}\|_2^2 &= \underset{\boldsymbol{x}, s.t. (4)}{\arg\min}\, \alpha^2 \\
&= \underset{\boldsymbol{x}, s.t. (4)}{\arg\min} \sum_{k=1}^{6MN-3M-3N} \left( \frac{(D\boldsymbol{x})_k}{f((D\tilde{\boldsymbol{v}})_k)} \right)^2.
\end{aligned}
\tag{6}
$$

Thus the problem (3) under the assumption (4) is represented by the following weighted least squares problem,

$$
\begin{array}{ll}
\text{Minimize} & \|FD\boldsymbol{x}\|_2^2 \\
\text{subject to} & \boldsymbol{v}_I = C\boldsymbol{x},\; x_i = v_i,\; \forall i \in \bar{\mathcal{I}},
\end{array}
\tag{7}
$$

where $F$ is diagonal matrix whose $i$th diagonal element is $1/f((D\tilde{\boldsymbol{v}})_i)$.

Next, we focus on the colorization function $f$. This paper assumes that the difference of color values is described by a monotonically increasing function of the difference of their grayscale values. The simple way to describe this is to give the function $f$ as follows,

$$
f(u) = |u|.
\tag{8}
$$

However, experimental results indicate that this function does not recover the color image correctly and that the further pixels from the given color pixels remain grayscale. In order to colorize a whole image, this paper proposes the following function,

$$
f(u) = \begin{cases} |u| - \nu & (|u| > \nu) \\ 0 & (|u| \le \nu) \end{cases},
\tag{9}
$$

where $\nu > 0$ is a given constant. The function $f$ in (9) implies that pixels with almost the same intensity value have the same color.

In [6], authors have proposed the colorization algorithm based in the mixed $\ell_0/\ell_1$ norm minimization, where the color value $\boldsymbol{x}$ is recovered by minimizing the $\ell_0$ norm of $D\boldsymbol{x}$ in (4), that is, letting $D\boldsymbol{x}$ to be sparse. The experimental results show that the sparse $D\boldsymbol{x}$ represents a good colorized image. The function $f$ in (9) is used in the $\ell_1$ norm minimization such as the forward and backward splitting (FOBOS) algorithm [9] to obtain a sparse vector. Hence it is expected that we can obtain sparser $D\boldsymbol{x}$ and better colorized image by minimizing $\|D\boldsymbol{x}\|_2^2$ with the constraints (4) using $f$ in (9) than without these constraints. Therefore the problem (7) is a sort of the sparse optimization, and the algorithm proposed in the paper is a revised version of that in [6].

Though the value of $\nu$ affects the performance of colorization, we hardly decide the best value because it depends on the property of each image to be colorized. In order to reduce the dependence of parameter, this paper proposes to smooth the function $f$ by series approximation. In the matrix $F$ of (7) the values of $1/f$ is calculated as follows,

$$\frac{1}{f(u)} = \begin{cases} \frac{1}{|u|-\nu} & (|u| > \nu) \\ \infty & (|u| \leq \nu) \end{cases}. \tag{10}$$

We approximate (10) by the following finite series,

$$\frac{1}{f(u)} \approx \frac{1}{\nu} \sum_{k=1}^{L} \frac{1}{\left(\frac{|u|}{\nu}\right)^k}, \tag{11}$$

where $L > 0$ is a given constant. Note that the (11) is exactly equal to (10) if $L \to \infty$ and less sensitive to the value of $\nu$ than (10). Finally, we obtain the colorization function $f$ as follows,

$$\frac{1}{f(u)} = \frac{1}{\nu} \sum_{k=1}^{L} \frac{1}{\left(\frac{|u|}{\nu} + \varepsilon^{1/k}\right)^k}, \tag{12}$$

where the $\varepsilon > 0$ is a small constant to avoid zero divide. The function $f$ presented by (8), (9) and (11) are shown in Figure 1 (a), (b) and (c), respectively.

The problem (7) is a convex optimization and can be solved exactly. This paper aims to provide a fast algorithm, and therefore we relax the problem using the Lagrangian relaxation and propose the following problem,

$$\text{Minimize} \quad \|FD\boldsymbol{x}\|_2^2 + \lambda_1\|\boldsymbol{v}_I - C\boldsymbol{x}\|_2^2 + \lambda_2\|M(\boldsymbol{v}-\boldsymbol{x})\|_2^2 \tag{13}$$

where $\lambda_1 > 0$ and $\lambda_2 > 0$ are given constants, and $M$ is a diagonal matrix whose diagonal element is defined by

$$M_{i,i} = \begin{cases} 1 & \text{if } i \in \bar{\mathcal{I}} \\ 0 & \text{otherwise} \end{cases}.$$

The least squares problem (13) can be solved simply as

$$\boldsymbol{x} = \begin{bmatrix} FD \\ \lambda_1 C \\ \lambda_2 M \end{bmatrix}^{\dagger} \begin{bmatrix} \mathbf{0} \\ \lambda_1 \boldsymbol{v}_{\boldsymbol{I}} \\ \lambda_2 M\boldsymbol{v} \end{bmatrix}, \tag{14}$$

where $\mathbf{0}$ denote the zero vector of size $6MN - 3M - 3N$, and $A^{\dagger}$ denotes the pseudoinverse of a matrix $A$.

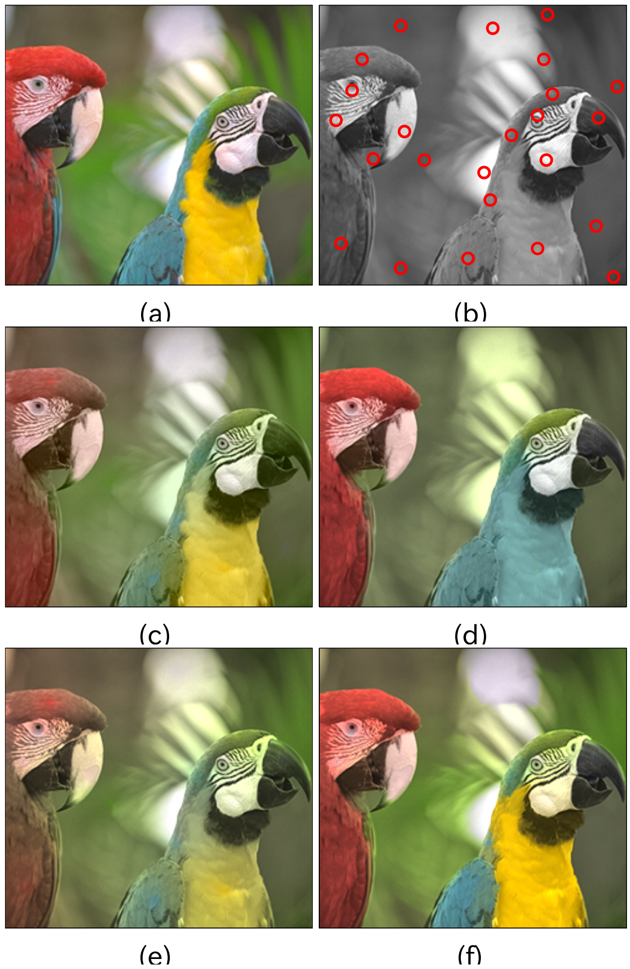


**Fig. 2**. (a) Original parrots image (247 × 225 pixels) and (b) given image with 24 color pixels. Results of (c) Levin's method [4], (d) the mixed $\ell_0/\ell_1$ norm minimization approach [6], (e) the proposed algorithm using (8) and (f) using (12).

## 3. NUMERICAL EXAMPLES

This section shows colorization examples to demonstrate the effectiveness of the proposed algorithm. In all examples we use $\varepsilon = 10^{-8}$, $L = 8$ $\lambda_1 = \lambda_2 = 100$ and $\nu = 6$, which give the best colorization. The values of the constants (1) are adopted as $[a_r\ a_g\ a_b] = [0.29891\ 0.58661\ 0.11448]$, which is usually used in the color conversion from RGB to YCbCr according to ITU-R BT.601.

We compare the proposed algorithm with Levin's method [4], the mixed $\ell_o/\ell_1$ norm minimization approach [6], and proposed algorithm using (8). Figure 2 - 4 show the results of these algorithms using only 24, 24, and 2 color pixels, respectively, and the red circles point out given color pixels. As

**Table 1.** PSNR [dB]

| | Parrots | Sunset | Design | Mandrill | Peppers |
|---|---|---|---|---|---|
| Levin's method [4] | 21.95 | 24.29 | 17.84 | 20.51 | 20.79 |
| Mixed $L_0/L_1$ norm minimization [6] | 16.57 | 23.51 | 18.19 | **22.12** | 21.60 |
| Proposed algorithm using (12) | **23.51** | **26.88** | **36.38** | 21.27 | **21.72** |
| Number of given color pixles | 24 | 24 | 2 | 36 | 36 |
| Image size | 225 × 247 | 180 × 180 | 180 × 180 | 180 × 180 | 180 × 180 |

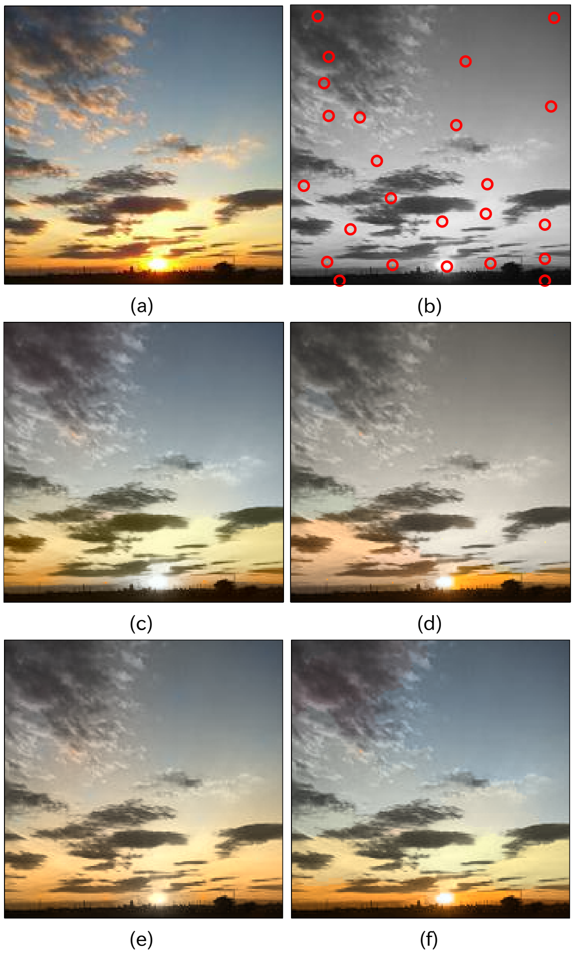

**Fig. 3.** (a) Original sunset image (180 × 180 pixels) and (b) given image with 24 color pixels. Results of (c) Levin's method [4], (d) the mixed $\ell_0/\ell_1$ norm minimization approach [6], (e) the proposed algorithm using (8) and (f) using (12).

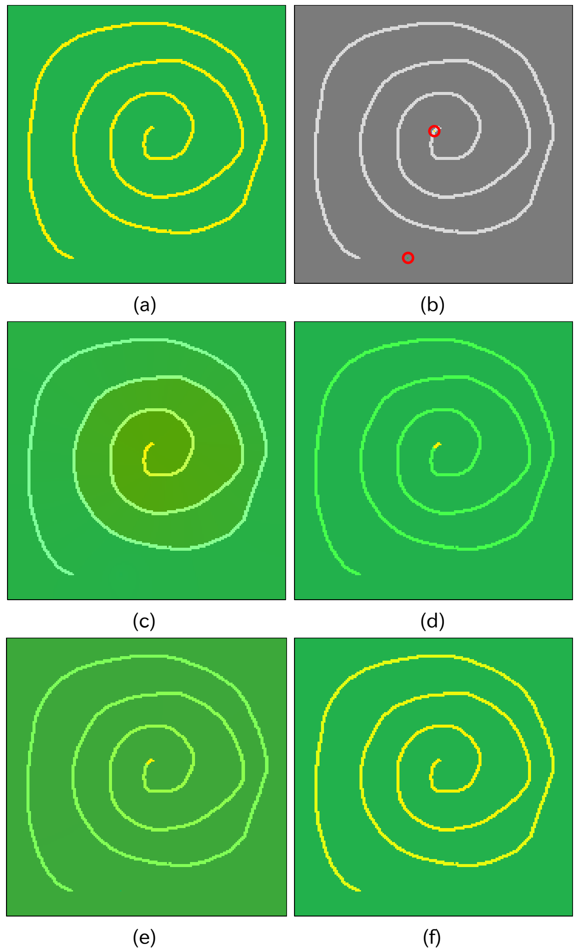

**Fig. 4.** (a) Original design image (180 × 180 pixels) and (b) given image with 2 color pixels. Results of (c) Levin's method [4], (d) the mixed $\ell_0/\ell_1$ norm minimization approach [6], (e) the proposed algorithm using (8) and (f) using (12).

can be seen, the proposed algorithm can colorize the grayscale image efficiently with a small number of color pixels. Table 1 shows the peak signal to noise ratio (PSNR) [dB] of resulting colorized image of each algorithm. Parrots, Sunset, and Design are images in Fig. 2 - 4, and Mandrill and Peppers are images known as 'mandrill' and 'peppers.png' in MATLAB, respectively. We can see that PSNR of the proposed algorithm is the highest value of the three algorithms in all images excepting the Mandrill image.

## 4. CONCLUSION

This paper proposes the digital image colorization algorithm using a colorization function, which is based on the sparse optimization and approximated by series. The image colorization problem is formulated as the weighted least squares problem and can be solved fast. Numerical examples show that the proposed algorithm can colorize a whole image effectively with a small number of color pixels comparing with the other algorithms.

## 5. REFERENCES


[1] N. A. Semary, M. M. Hadhoud, W. S. ElKilani, and N. A. Isamail, "Texture recognition based natural gray images coloring technique," *Proc. of International Conference on Computer Engineering and Systems*, pp. 237–245, 2007.

[2] V. G. Jacob and S. Gupta, "Colorization of grayscale images and videos using a semiautomatic approach," *Proc. of IEEE International Conference on Image Processing*, pp. 1633–1636, 2009.

[3] Y. Qu, T. Wong, and P. A. Heng, "Manga colorization," *ACM Trans.Graph*, vol. 25, pp. 1214–1220, 2006.

[4] A. Levin, D. Lischinski, and Y. Weiss, "Colorization using optimization," *ACM Transactions on Graphics*, vol. 23, no. 3, pp. 689–694, 2004.

[5] C. Rusu and S. A. Tsaftaris, "Estimation of scribble placement for painting colorization," *Proc. of 8th International Symposium on Image and Signal Processing and Analysis*, 2013.

[6] K. Uruma, K. Konishi, T. Takahashi, and T. Furukawa, "Image colorization based on the mixed $l_0/l_1$ norm minimization," proc. of ieee international conference on image processing," *Prof of IEEE International Conference on Image Processing*, pp. 2113–2116, 2012.

[7] R. Chartrand and W. Yin, "Iteratively reweighted algorithms for compressive sensing," *in Proc. IEEE Int. Conf. Acoust . Speech Signal Process.*, pp. 3869–3872, Apr 2008.

[8] I. Daubechies, R. DeVore, M. Fornasier, and C. Gunturk, "Iteratively reweighted least squares minimization for sparse recovery," *Comm. Pure Appl. Math.,*, vol. 63, no. 1, pp. 1–38, 2010.

[9] John Duchi and Yoram Singer, "Efficient online and batch learning using forward backward splitting," *Journal of Machine Learning Research*, vol. 10, pp. 2899–2934, Dec 2009.