

Unified Optimization Framework for L2, L1, and/or L0 Constrained Image Reconstruction

Masayuki Tanaka^a and Masatoshi Okutomi^a

^aTokyo Institute of Technology, Japan

ABSTRACT

In this paper, we propose a unified optimization framework for L2, L1, and/or L0 constrained image reconstruction. First, we generalize cost functions for image reconstruction, which consist of a fidelity term with L2 norm and constraint terms with L2, L1, and/or L0 norms. This generalized cost function covers many types of existing cost functions for image reconstruction. Then, we show that this generalized cost function can be optimized by the alternating direction method of multipliers (ADMM). The ADMM is a well-known iterative optimization approach for convex problems. Experimental results demonstrate that the proposed unified optimization framework is applicable to a wide range of applications.

Keywords: alternating direction method of multipliers, image reconstruction, proximal

1. INTRODUCTION

Image filtering is very essential and important image processing. Many of those filters are reconstruction-based image filter. In the reconstruction-based image filter, the filtering image is reconstructed by minimizing the cost function associated to the filter. In the literature, many reconstruction-based image filters have been proposed.^{1,2,3,4}

Heide et al generalized those cost functions for image filtering as sum of convex functions of linear transformation of the image.⁵ In this paper, we focus sum of L2, L1, and L0 norms of the convolution of the image for the cost function. This cost function with L2, L1, and L0 norms represents many types of filtering and applications. For this cost function, we show a unified optimization framework based on the alternating direction method of multipliers (ADMM).⁶ The ADMM is a well-known iterative optimization approach for convex problems.

In this paper, we propose a unified optimization framework considering practical use. In many image filtering application, we can obtain an input image. The property that the input image is gradually updated by iterative optimization is desired in many situation because we sometimes don't have enough time to apply many iterations. Therefore, in the proposed implementation, we modify the order of the updating sequence in the iterative ADMM algorithm.

As we discuss in section 4, the discrete fourier transform and its inverse are required at each ADMM iteration for our problem. The discrete fourier transform implicitly assumes that the image is cyclic. However, natural images usually have very large intensity difference between left and right boundaries of images. The top and bottom boundaries are same situation. Those intensity gaps cause severe artifacts around image boundaries. Even if we simply padding by replicating nearest pixel value, the intensity gaps are still remained. In order to avoid those artifacts, we propose a seamless image padding. Therefore, we propose the padding algorithm which seamlessly connect image boundaries. We apply the proposed seamless padding algorithm before apply the discrete fourier transform.

Experimental results demonstrate that the proposed unified optimization framework is applicable to a wide range of applications.

Further author information: (Send correspondence to Masayuki Tanaka)
Masayuki Tanaka: E-mail: mtanaka@sc.e.titech.ac.jp

2. ALTERNATING DIRECTION METHOD OF MULTIPLIERS (ADMM)

The ADMM⁶ is very powerful optimization algorithm. The ADMM algorithm can minimize the following cost function,

$$\begin{cases} E(\mathbf{x}) &= f(\mathbf{x}) + g(\mathbf{z}), \\ \text{s.t.} & \mathbf{B}\mathbf{x} - \mathbf{y} - \mathbf{z} = 0, \end{cases} \quad (1)$$

where \mathbf{x} , \mathbf{z} , and \mathbf{y} are vectors, \mathbf{B} is matrix, $f(\mathbf{x})$ is quadric function, and $g(\mathbf{z})$ is convex function. The ADMM is iterative minimization algorithm. The update rules are summarized as

$$\mathbf{x}^{k+1} := \arg \min_{\mathbf{x}} f(\mathbf{x}) + \frac{\rho}{2} \|\mathbf{B}\mathbf{x} - \mathbf{y} - \mathbf{z}^k + \mathbf{u}^k\|_2, \quad (2)$$

$$\mathbf{z}^{k+1} := \arg \min_{\mathbf{z}} g(\mathbf{z}) + \frac{\rho}{2} \|\mathbf{B}\mathbf{x}^{k+1} - \mathbf{y} - \mathbf{z} + \mathbf{u}^k\|_2 = \text{prox}_{g,\rho}(\mathbf{B}\mathbf{x}^{k+1} - \mathbf{y} + \mathbf{u}^k), \quad (3)$$

$$\mathbf{u}^{k+1} := \mathbf{u}^k + (\mathbf{B}\mathbf{x}^{k+1} - \mathbf{y} - \mathbf{z}^{k+1}), \quad (4)$$

where $\text{prox}_{g,\rho}(\mathbf{x})$ is called proximal operator of function g . When ρ is one, we simply denote the proximal operator as $\text{prox}_g(\mathbf{x})$.

2.1 Proximal operator of L1 and L0 norms

In this paper, we mainly discuss L2, L1, and L0 norms. The proximal operator of the quadratic function is not necessary for the ADMM algorithm. Here, we only summarize the proximal operators of L1 and L0 norms, because L2 norm is quadratic function.

The proximal operator of L1 norm, $\text{prox}_{L1,\rho}(\xi)$, is⁶

$$\text{prox}_{L1,\rho}(\xi) = \text{sign}(\xi)(|\xi| - 1/\rho)_+ \quad (5)$$

where

$$\text{sign}(\xi) = \begin{cases} -1 & (\xi < 0) \\ 0 & (\xi = 0) \\ 1 & (\xi > 0) \end{cases}, \quad (6)$$

$$(\xi)_+ = \max(\xi, 0). \quad (7)$$

The proximal operator of L1 norm is associated to soft thresholding. The proximal operator of L0 norm, $\text{prox}_{L0,\rho}(\xi)$, is^{2,3}

$$\text{prox}_{L0,\rho}(\xi) = \begin{cases} 0 & (\xi^2 < \rho/2) \\ \xi & (\xi^2 \geq \rho/2) \end{cases}. \quad (8)$$

The proximal operator of L0 norm is associated to hard thresholding. The L0 norm is not convex function. But, it is empirically known that ADMM can optimize the L0 norm.

3. GENERAL COST FUNCTION FOR IMAGE FILTER

Heide et al have generalized an image optimization problem as a sum of penalty functions of linear transforms of the image.⁵ Here, we focus L2, L1, and L0 norms for the penalty function and the convolution operation for the linear transform. Many image filtering and image reconstruction problem can be generalized by the optimization problem of the following cost function:

$$\begin{aligned} E(\mathbf{x}) &= \sum_{p=0,1,2} \sum_{i=1}^{N_p} \lambda_{p,i} \|\mathbf{b}_{p,i} * \mathbf{x} - \mathbf{y}_{p,i}\|_p \\ &= \sum_{i=1}^{N_2} \lambda_{2,i} \|\mathbf{b}_{2,i} * \mathbf{x} - \mathbf{y}_{2,i}\|_2 + \sum_{i=1}^{N_1} \lambda_{1,i} \|\mathbf{b}_{1,i} * \mathbf{x} - \mathbf{y}_{1,i}\|_1 + \sum_{i=1}^{N_0} \lambda_{0,i} \|\mathbf{b}_{0,i} * \mathbf{x} - \mathbf{y}_{0,i}\|_0, \end{aligned} \quad (9)$$

where \mathbf{x} is the vector representation of the image to be optimized, $\mathbf{b}_{p,i}$ is the vector representation of i -th convolution kernel of p norm term, $\mathbf{y}_{p,i}$ is the vector representation of i -th data or bias of p norm term, $*$ represents the convolution operator, and $\|\mathbf{x}\|_p$ represents the p norm of the vector \mathbf{x} . Here, the p norm is defined by

$$\|\mathbf{x}\|_p = \sum_j |x_j|^p, \quad (10)$$

where x_j is j -th element of the vector \mathbf{x} . In this paper, we focusing to solve the unified optimization problem in Eq. 9.

For example, the cost function of deblurring problem with classical Tikhonov regularization can be expressed as

$$E(\mathbf{x}) = \|\mathbf{b} * \mathbf{x} - \mathbf{y}\|_2 + \lambda \|\mathbf{x}\|_2. \quad (11)$$

where \mathbf{y} is input blurred image, \mathbf{b} is blurring kernel and λ is constraint parameter. We can obtain Eq. 11 by just plugging in the following parameters:

$$\begin{cases} N_2 = 2, & N_1 = 0 & N_0 = 0, \\ \mathbf{b}_{2,1} = \mathbf{b}, & \mathbf{y}_{2,1} = \mathbf{y}, & \lambda_{2,1} = 1, \\ \mathbf{b}_{2,2} = 1, & \mathbf{y}_{2,2} = 0, & \lambda_{2,2} = \lambda. \end{cases} \quad (12)$$

Total variation regularized least-squares deconvolution (TV-deconvolution) is famous deconvolution approach.¹ Although it has several variations, one of the cost functions is

$$E(\mathbf{x}) = \|\mathbf{b} * \mathbf{x} - \mathbf{y}\|_2 + \lambda \{ \|\nabla_h \mathbf{x}\|_1 + \|\nabla_v \mathbf{x}\|_1 \}, \quad (13)$$

where ∇_h and ∇_v are horizontal and vertical derivative operators, respectively. The cost function of the TV-deconvolution can be also derived from Eq. 9 with these parameters:

$$\begin{cases} N_2 = 1, & N_1 = 2 & N_0 = 0, \\ \mathbf{b}_{2,1} = \mathbf{b}, & \mathbf{y}_{2,1} = \mathbf{y}, & \lambda_{2,1} = 1, \\ \mathbf{b}_{1,1} = \mathbf{d}_h, & \mathbf{y}_{1,1} = 0, & \lambda_{1,1} = \lambda, \\ \mathbf{b}_{1,2} = \mathbf{d}_v, & \mathbf{y}_{1,2} = 0, & \lambda_{1,2} = \lambda, \end{cases} \quad (14)$$

where \mathbf{d}_h and \mathbf{d}_v are horizontal and vertical discrete derivative kernels, respectively.

L0 norm constrain is also used for an image filter to extract base structure.² In this paper, we call this filter gradient L0 norm constraint filter. The gradient L0 norm constraint filter can be expressed by optimization problem. The cost function of the gradient L0 norm constraint filter is *

$$E(\mathbf{x}) = \|\mathbf{x} - \mathbf{y}\|_2 + \lambda \{ \|\nabla_h \mathbf{x}\|_0 + \|\nabla_v \mathbf{x}\|_0 \}. \quad (15)$$

This cost function can be also derived from Eq. 9. The parameters for the gradient L0 norm constraint filter are summarized as

$$\begin{cases} N_2 = 1, & N_1 = 0 & N_0 = 2, \\ \mathbf{b}_{2,1} = \boldsymbol{\delta}, & \mathbf{y}_{2,1} = \mathbf{y}, & \lambda_{2,1} = 1, \\ \mathbf{b}_{0,1} = \mathbf{d}_h, & \mathbf{y}_{0,1} = 0, & \lambda_{0,1} = \lambda, \\ \mathbf{b}_{0,2} = \mathbf{d}_v, & \mathbf{y}_{0,2} = 0, & \lambda_{0,2} = \lambda, \end{cases} \quad (16)$$

where $\boldsymbol{\delta}$ represents the delta function.

A gradient-based image processing is very powerful.^{7,8,9,10} The cost function of the gradient-based image processing can be expressed as

$$E(\mathbf{x}) = \varepsilon \|\mathbf{x} - (I_{\min} + I_{\max})/2\|_2 + \|\nabla_h \mathbf{x} - \boldsymbol{\xi}_h\|_2 + \|\nabla_v \mathbf{x} - \boldsymbol{\xi}_v\|_2, \quad (17)$$

*It is not exactly same, but it can be considered as variation.

where ε is a small positive value, I_{\min} and I_{\max} are minimum and maximum value of image intensity range, and ξ_h and ξ_v are desired horizontal and vertical gradient, respectively. The first term in Eq. 17 is necessary to provide an unique solution. The parameters for the gradient-based image editing can be summarized as

$$\begin{cases} N_2 = 1, & N_1 = 0 & N_0 = 2, \\ \mathbf{b}_{2,1} = \boldsymbol{\delta}, & \mathbf{y}_{2,1} = \mathbf{y}, & \lambda_{2,1} = \varepsilon, \\ \mathbf{b}_{2,2} = \mathbf{d}_h, & \mathbf{y}_{2,2} = 0, & \lambda_{2,2} = \xi_h, \\ \mathbf{b}_{2,3} = \mathbf{d}_v, & \mathbf{y}_{2,3} = \xi_v, & \lambda_{2,3} = 1, \end{cases} \quad (18)$$

As discussed in above, many image processing algorithm can be expressed with optimization problem of the cost function in Eq. 9. Therefore, in the next section, we describe the optimization algorithm to minimize the cost function in Eq. 9, so that we can solve many types of image processing problem with unified framework.

4. PROPOSED OPTIMIZATION ALGORITHM

Here, we discuss the optimization algorithm for the cost function in Eq. 9. The generalized optimization problem can be rewritten as the form in Eq. 1. The rewritten problem is

$$\begin{cases} E(\mathbf{x}) = \sum_{i=1}^{N_2} \lambda_{2,i} \|\mathbf{b}_{2,i} * \mathbf{x} - \mathbf{y}_{2,i}\|_2 + \sum_{i=1}^{N_1} \lambda_{1,i} \|\mathbf{z}_{1,i}\|_1 + \sum_{i=1}^{N_0} \lambda_{0,i} \|\mathbf{z}_{0,i}\|_0, \\ s.t. \quad \mathbf{z}_{1,i} = \mathbf{b}_{1,i} * \mathbf{x} - \mathbf{y}_{1,i}, \quad \mathbf{z}_{0,i} = \mathbf{b}_{0,i} * \mathbf{x} - \mathbf{y}_{0,i}, \end{cases} \quad (19)$$

where $\mathbf{z}_{1,i}$ and $\mathbf{z}_{0,i}$ have same dimension as the image \mathbf{x} . Once we can rewrite the optimization problem as the same form as in Eq. 1, we can simply apply the ADMM algorithm.⁶ Actually, the L0 norm is not convex function. But, it is empirically known the ADMM algorithm can optimize the L0 norm. The update rules of the ADMM for the problem in Eq. 19 are summarized as⁵

$$\mathbf{x}^{k+1} := \arg \min_{\mathbf{x}} \sum_{i=1}^{N_2} \lambda_{2,i} \|\mathbf{b}_{2,i} * \mathbf{x} - \mathbf{y}_{2,i}\|_2 + \frac{\rho}{2} \sum_{p=0,1} \sum_{i=1}^{N_p} \lambda_{p,i} \|\mathbf{b}_{p,i} * \mathbf{x} - \mathbf{y}_{p,i} - \mathbf{z}_{p,i}^k + \mathbf{u}_{p,i}^k\|_2, \quad (20)$$

$$\mathbf{z}_{p,i}^{k+1} := \text{prox}_{L_{p,\rho}}(\mathbf{b}_{p,i} * \mathbf{x}^{k+1} - \mathbf{y}_{p,i} + \mathbf{u}_{p,i}^k), \quad (\forall i, p \in \{0, 1\}) \quad (21)$$

$$\mathbf{u}_{p,i}^{k+1} := \mathbf{u}_{p,i}^k + (\mathbf{b}_{p,i} * \mathbf{x} - \mathbf{y}_{p,i} - \mathbf{z}_{p,i}^{k+1}), \quad (\forall i, p \in \{0, 1\}) \quad (22)$$

where \mathbf{x}^0 , $\mathbf{z}_{p,i}^0$, and $\mathbf{u}_{p,i}^0$ are initialized by $\mathbf{0}$. Eqs. 21 and 22 can be easily solved. The proximal operators of L1- and L0- norms in Eq. 21 are given as soft-thresholding in Eq. 5 and hard-thresholding in Eq. 8, respectively. Eq. 20 has a direct closed form solution.^{2,5} The objective function to be minimized in Eq. 20 can be reformulated in frequency domain as

$$J(\tilde{\mathbf{x}}) = \sum_{i=0}^{N_2} \lambda_{2,i} \|\tilde{\mathbf{b}}_{2,i} \otimes \tilde{\mathbf{x}} - \tilde{\mathbf{y}}_{2,i}\|_2 + \frac{\rho}{2} \sum_{p=0,1} \sum_{i=1}^{N_p} \lambda_{p,i} \|\tilde{\mathbf{b}}_{p,i} \otimes \tilde{\mathbf{x}} - \tilde{\mathbf{y}}_{p,i} - \tilde{\mathbf{z}}_{p,i}^k + \tilde{\mathbf{u}}_{p,i}^k\|_2, \quad (23)$$

$$(24)$$

where \otimes represents an element-wise product, and $\tilde{\mathbf{x}}$, $\tilde{\mathbf{y}}$, $\tilde{\mathbf{z}}$ and $\tilde{\mathbf{u}}$ are discrete fourier transform of \mathbf{x} , \mathbf{y} , \mathbf{z} and \mathbf{u} , respectively. The direct closed form solution is

$$\tilde{\mathbf{x}}^{k+1} = \tilde{\mathbf{x}}_N^{k+1} \oslash \tilde{\mathbf{x}}_D^{k+1} \quad (25)$$

$$\tilde{\mathbf{x}}_N^{k+1} = \sum_{i=1}^{N_2} \lambda_{2,i} \tilde{\mathbf{b}}_{2,i}^* \otimes \tilde{\mathbf{y}}_{2,i} + \frac{\rho}{2} \sum_{p=0,1} \sum_{i=1}^{N_p} \lambda_{p,i} \mathbf{b}_{p,i}^* \otimes (\tilde{\mathbf{y}}_{p,i} + \tilde{\mathbf{z}}_{p,i}^k - \tilde{\mathbf{u}}_{p,i}^k), \quad (26)$$

$$\tilde{\mathbf{x}}_D^{k+1} = \sum_{i=1}^{N_2} \lambda_{2,i} \tilde{\mathbf{b}}_{2,i}^* \otimes \tilde{\mathbf{b}}_{2,i} + \frac{\rho}{2} \sum_{p=0,1} \sum_{i=1}^{N_p} \lambda_{p,i} \tilde{\mathbf{b}}_{p,i}^* \otimes \tilde{\mathbf{b}}_{p,i}, \quad (27)$$

where \oslash represents element-wise division, and $\tilde{\mathbf{b}}^*$ is complex conjugate of $\tilde{\mathbf{b}}$. We can obtain the updated \mathbf{x}^{k+1} by simply applying the inverse discrete fourier transformation to $\tilde{\mathbf{x}}^{k+1}$ in Eq. 25

Algorithm 1 Proposed Implementation

- 1: Apply the proposed seamless padding to \mathbf{x}_{inp} and $\mathbf{y}_{p,i}$
 - 2: Initialize $\mathbf{x}^0 = \mathbf{x}_{\text{inp}}$, $\mathbf{u}_{p,i}^0 = \mathbf{0}$
 - 3: **for** $k = 1$ to K **do**
 - 4: Update padded region of \mathbf{x}^k
 - 5: Update $\mathbf{z}_{p,i}^{k+1}$ with \mathbf{x}^k and $\mathbf{u}_{p,i}^k$ by Eq. 21
 - 6: Update $\mathbf{u}_{p,i}^{k+1}$ with \mathbf{x}^k and $\mathbf{z}_{p,i}^{k+1}$ by Eq. 22
 - 7: Update \mathbf{x}^{k+1} with $\mathbf{z}_{p,i}^{k+1}$ and $\mathbf{u}_{p,i}^k$ by Eq. 25
 - 8: **end for**
 - 9: Reverse padding \mathbf{x}^K
-

4.1 Proposed Implementation

The generalized cost function in Eq. 9 can be theoretically solved as mentioned in above.⁵ However, in practice, we need to care for the implementation. The points are an iteration and discrete fourier transformation.

The ADMM algorithm converges regardless initial guess, if the problem is convex. As mentioned in previous section, L0 norm in the generalized cost function is not convex function. In same situation, we can not apply enough number of iteration. In that case, initial guess is very important to get desired reconstructed image. In many image processing problems such as deblurring, denoising, and image filtering, we usually have an input image. In practice, this input image is good choice for the initial guess. In addition, the property that the input image is gradually updated in the iterative optimization is very helpful to understand the filter. In the original ADMM update rules, the image \mathbf{x} is firstly updated with other variables $\mathbf{z}_{p,i}$ and $\mathbf{u}_{p,i}$ as in Eq. 20. Even if we set the input image \mathbf{x}_{inp} for the initial guess, the image \mathbf{x} is updated without the initial guess at very first step. Therefore, we modify an order of updating for the proposed implementation. First, the image \mathbf{x}^0 and variables $\mathbf{u}_{p,i}^0$ are initialized with the input image \mathbf{x}_{inp} and $\mathbf{0}$. Then, updating is performed as follow: the variables $\mathbf{z}_{p,i}$ in Eq. 21, the variables $\mathbf{u}_{p,i}$ in Eq. 22, and the image \mathbf{x} . Using this updating order, we can obtain reasonable output image even if an iteration is insufficient. The proposed implementation is summarized in Alg. 1.

4.2 Proposed Seamless Padding Algorithm

The discrete fourier transform and its inverse are required in each ADMM iteration. The discrete fourier transform implicitly assumes a cyclic convolution. In other words, a cyclic image is assumed. However, in many natural images, intensities at left boundary and right boundary are strongly different as shown in Fig. 1. The situation for the top and bottom boundaries are same. Those strong discontinuity in the cyclic image cause severe artifact, when we iteratively apply the discrete fourier transform and its inverse.

Even if the image is padded by replicating nearest pixel values, the cyclic image of it has still strong discontinuities on the boundaries. Figure 2 is an example of padding with nearest pixel values. Therefore, we propose a seamless padding algorithm which generates seamless boundaries in cyclic image as shown in Fig. 3. In the proposed seamless padding algorithm, a padded region is generated by gradient-based image processing.^{7,8,9,10} The gradient based image processing is known as very powerful tool. Here, we assume that the image is cyclic and that the gradient in the padding region should be zero. The boundary condition is given by pixel values of the original image, which is the Dirichlet boundary condition. Let \mathbf{q} be the vector representation of pixel values in the padding region as shown in Fig. 3-(a). The proposed seamless padding can be formulated as

$$\mathbf{q} = \arg \min_{\mathbf{q}} \|\nabla_h *_c \mathbf{q}\|_2 + \|\nabla_v *_c \mathbf{q}\|_2, \quad (28)$$

where $*_c$ represents a cyclic convolution operator for the padding region \mathbf{q} . This problem is same as Poisson image editing.⁷ We can solve this problem by iterative Gauss-Seidel algorithm. In the ADMM algorithm, each iteration requires the fourier transform and its inverse. Then, we need to solve Eq. 28 for each ADMM iteration. However, the padding results of the previous ADMM iteration is used for the initial guess for the current seamless padding. Although the seamless padding algorithm is an iterative algorithm, this updating is reasonably fast.

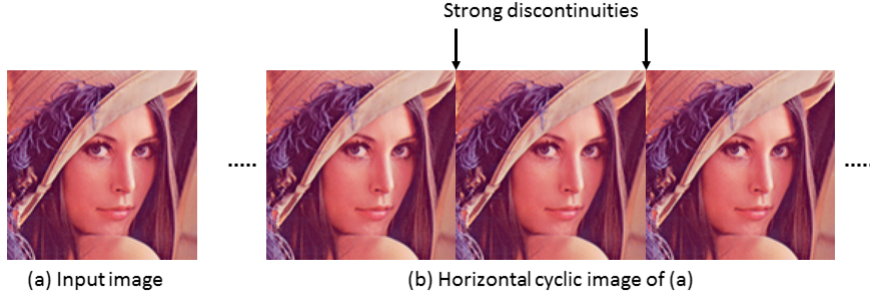


Figure 1. Cyclic image of original image.

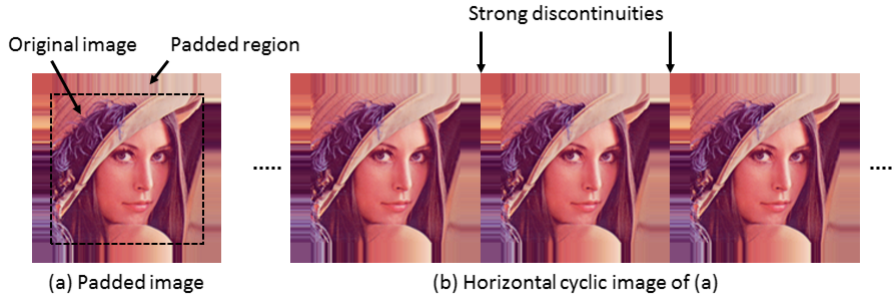


Figure 2. Cyclic image of image padded by replicating nearest pixel values.

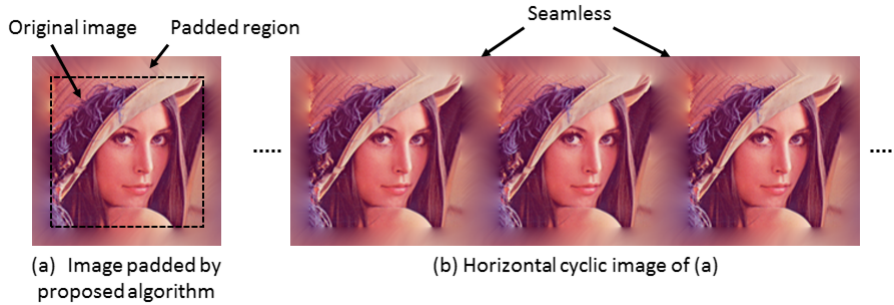


Figure 3. Cyclic image of image padded proposed seamless padding.

5. EXPERIMENTS

We implemented the proposed method with matlab[†]. First, we demonstrate the effectiveness of the proposed update sequence and the proposed seamless padding algorithm. Then, we show applications of the gradient L0 and L1 norm constraint image filter and image enhancement with the proposed general framework.

5.1 Updating Sequence

In the original updating sequence, the initial guess is zero and updating is started from the reconstructed image \mathbf{x} , while the initial guess is the input image and updating of the reconstructed image \mathbf{x} is last in the proposed updating sequence. Noisy lena image, where noise level is 10, is denoised by minimizing the cost function:

$$E(\mathbf{x}) = \|\mathbf{x} - \mathbf{y}\|_2 + \lambda_1 \{ \|\nabla_h \mathbf{x}\|_1 + \|\nabla_v \mathbf{x}\|_1 \} + \lambda_0 \{ \|\nabla_h \mathbf{x}\|_0 + \|\nabla_v \mathbf{x}\|_0 \}, \quad (29)$$

where \mathbf{y} is the input noisy image. Here, we set $\lambda_1 = 15$ and $\lambda_0 = 15^2$. Figure 4 shows that reconstructed images of first three iteration. Both original updating sequence and proposed updating sequence converge. However,

[†] The matlab code is available online: <http://www.ok.sc.e.titech.ac.jp/res/IC/ImReconProx/>

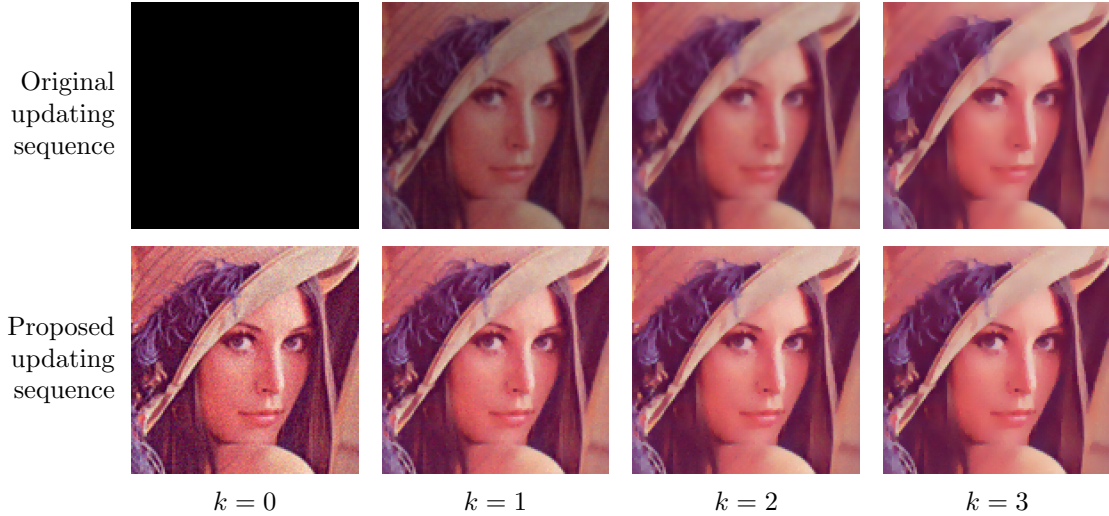


Figure 4. Reconstructed images at each iteration k , where top and bottom rows are existing and proposed implementation, respectively.



Figure 5. L1 norm constrained deblurring results, where (a) is an image deblurred with linear motion kernel, (b) is the deblurring result without padding, and (c) is the deblurring result with the proposed seamless padding

the reconstructed images in the original updating sequence are dark until $k = 2$ because the initial guess is zero, while the reconstructed images are gradually updated from the input image in the proposed updating sequence. This property of the proposed implementation is very important in practice, especially real time processing.

5.2 Seamless Padding

We demonstrate the effect of the proposed seamless padding algorithm with the deblurring problem. Here, we apply the L1 norm constrained deblurring.^{2,4} The cost function can be expressed as

$$E(\mathbf{x}) = \|\mathbf{b} * \mathbf{x} - \mathbf{y}\|_2 + \lambda_1 \{ \|\nabla_h \mathbf{x}\|_1 + \|\nabla_v \mathbf{x}\|_1 \}, \quad (30)$$

where \mathbf{y} is the input blurred image, and \mathbf{b} is the linear motion blur kernel. Here, we set $\lambda_1 = 0.01$. Figure 5 shows deblurring results. If we did not apply the image padding before fourier transformation to solve Eq. 20 in the ADMM algorithm, we can obtain severe artifacts around borders as shown in Fig. 5-(b). On the contrast, those artifacts are disappeared by applying the proposed seamless padding as shown in Fig. 5-(c).

5.3 Image Filtering Applications

We demonstrate the image filter which minimizes Eq. 29 with different constraint parameters. Filtering results are summarized in Fig. 6. We can obtain various types of effect just by changing parameters with same optimization framework.

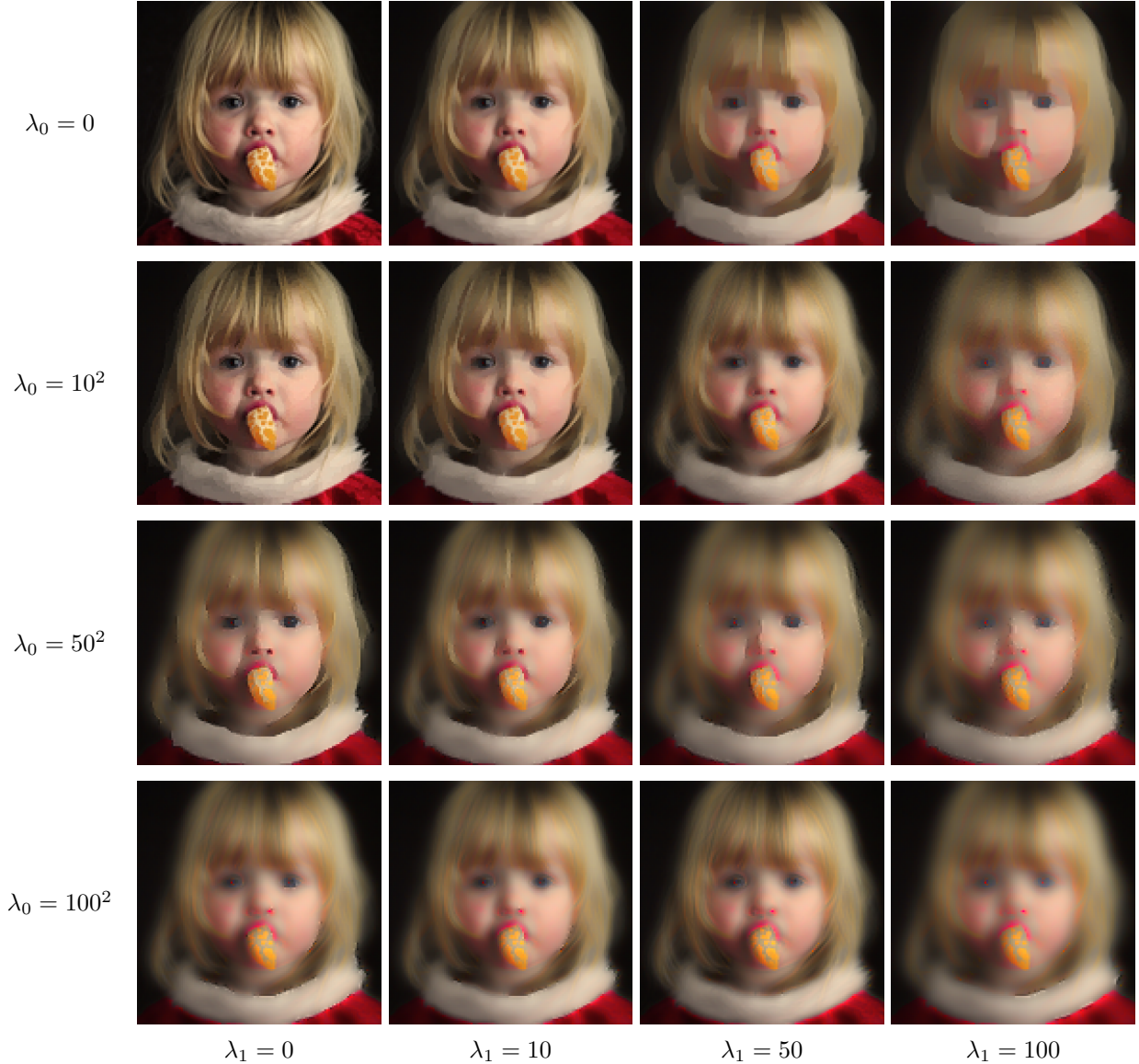


Figure 6. Results of L1 norm and L0 norm gradient constraint filter with different constraint parameters, where top-left image is associated th original image.

Our framework can be used for the image enhancement. Although we can consider various types of cost functions for the image enhancement, we demonstrate following two types of cost functions:

$$E_1(\mathbf{x}) = \{ \|\nabla_h \mathbf{x} - \alpha \nabla_h \mathbf{y}\|_1 + \|\nabla_v \mathbf{x} - \alpha \nabla_v \mathbf{y}\|_1 \} + \varepsilon \|\mathbf{x} - \mathbf{y}\|_2, \quad (31)$$

$$E_2(\mathbf{x}) = \{ \|\nabla_h \mathbf{x} - \alpha \nabla_h \mathbf{y}\|_2 + \|\nabla_v \mathbf{x} - \alpha \nabla_v \mathbf{y}\|_2 \} + \varepsilon \|\mathbf{x} - \mathbf{y}\|_2, \quad (32)$$

where \mathbf{y} is the input image, ε is constraint parameter and α is enhancement parameter. Here, we set two for α . If ε is large enough, the filtering results is identical to the input image. Note that Eq. 32 is same form of the modified Poisson image reconstruction.⁸ They used for image editing in their paper.⁸ Figure 7 shows the enhancement results. As demonstrated in here, one can generate various types of effect image filter just changing the cost function with the proposed optimization framework.

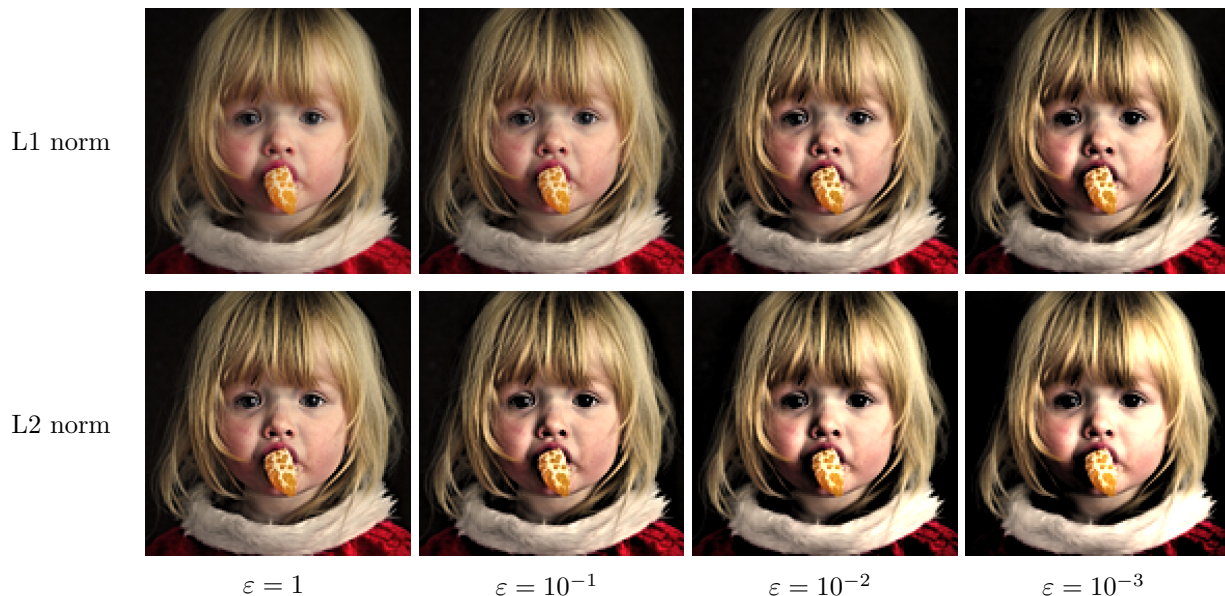


Figure 7. Results of image gradient-based image enhancement with different parameters.

6. CONCLUSION

We have generalized the cost function of image filtering by sum of L2-, L1, and L0- norm of the convolution of the image. It is shown that many applications are expressed with this generalized cost. Then, we discuss the ADMM solution for the generalized cost function. For the proposed implementation, we have modified order of updating sequence from the original ADMM updating rules, so that the input image is smoothly change to the output filtering image. We have also proposed the seamless padding algorithm. The proposed seamless padding algorithm can effectively reduce the severe artifacts around image boundary which caused by the fourier transform. The proposed framework is applied to several application. It is demonstrated that one can generate various types of image filtering just by changing the parameters of the generalized cost function.

REFERENCES

- [1] Chan, S. H., Khoshabeh, R., Gibson, K. B., Gill, P. E., and Nguyen, T. Q., “An augmented lagrangian method for total variation video restoration,” *IEEE Transactions on Image Processing* **20**(11), 3097–3111 (2011).
- [2] Xu, L., Lu, C., Xu, Y., and Jia, J., “Image smoothing via l0 gradient minimization,” in [*ACM Transactions on Graphics (TOG)*], **30**(6), 174, ACM (2011).
- [3] Blumensath, T. and Davies, M. E., “Iterative thresholding for sparse approximations,” *Journal of Fourier Analysis and Applications* **14**(5-6), 629–654 (2008).
- [4] Beck, A. and Teboulle, M., “Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems,” *IEEE Transactions on Image Processing* **18**(11), 2419–2434 (2009).
- [5] Heide, F., Diamond, S., Nießner, M., Ragan-Kelley, J., Heidrich, W., and Wetzstein, G., “Proximal: Efficient image optimization using proximal algorithms,” *ACM Transactions on Graphics (TOG)* **35**(4), 84 (2016).
- [6] Boyd, S., Parikh, N., Chu, E., Peleato, B., and Eckstein, J., “Distributed optimization and statistical learning via the alternating direction method of multipliers,” *Foundations and Trends® in Machine Learning* **3**(1), 1–122 (2011).
- [7] Pérez, P., Gangnet, M., and Blake, A., “Poisson image editing,” in [*ACM Transactions on Graphics (TOG)*], **22**(3), 313–318, ACM (2003).
- [8] Tanaka, M., Kamio, R., and Okutomi, M., “Seamless image cloning by a closed form solution of a modified poisson problem,” in [*SIGGRAPH Asia 2012 Posters*], 15, ACM (2012).

- [9] Bhat, P., Zitnick, C. L., Cohen, M., and Curless, B., “Gradientshop: A gradient-domain optimization framework for image and video filtering,” *ACM Transactions on Graphics (TOG)* **29**(2), 10 (2010).
- [10] Shibata, T., Tanaka, M., and Okutomi, M., “Gradient-domain image reconstruction framework with intensity-range and base-structure constraints,” in [*IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*], 2745–2753 (2016).