EDGE-PRESERVING FILTERING BY PROJECTION ONTO L0 GRADIENT CONSTRAINT

Shunsuke Ono

Tokyo Institute of Technology

ABSTRACT

We propose an edge-preserving filtering method with a novel use of the L_0 gradient. Our method, termed as the L_0 gradient projection, is formulated as the minimization of a quadratic data-fidelity to an input image subject to the constraint that the L_0 gradient, the number of non-zero gradients, of the output image is less than a user-given parameter α . This strategy is much more intuitive than the conventional approach, the L_0 gradient minimization, that minimizes the sum of the L_0 gradient plus the quadratic data-fidelity, because one can directly impose a desired degree of flatness by α , which is impossible in the L_0 gradient minimization. We also provide an efficient algorithm based on the so-called alternating direction method of multipliers for solving the nonconvex optimization problem associated with the L_0 gradient projection. The utility of the L_0 gradient projection is illustrated by experiments.

Index Terms— L_0 gradient, edge-preserving filtering, constrained optimization, nonconvex optimization

1. INTRODUCTION

Edge-preserving filtering is a basic tool in image processing, and a variety of edge-preserving filtering methods have been proposed. Among them, the L_0 gradient minimization [1], which minimizes the number of non-zero gradients, i.e., L_0 gradient, of the output image plus a quadratic data-fidelity to an input image, is known to have a remarkable ability of edge-preserving *flattening*, so that it offers many applications, such as edge extraction, clip-art compression artifact removal and detail enhancement provided in the original paper [1], and more [2–6]. At the same time, due to the nonconvexity of the L_0 gradient minimization, a number of algorithms for approximately solving it have also been developed [7–9].

In the L_0 gradient minimization, the degree of flatness of the output image is controlled by a user-given parameter λ that balances the relative importance of the L_0 gradient to the quadratic data-fidelity. Arguably, selecting a suitable value of λ is a difficult task because λ does not directly correspond to the degree of flatness of the output image, as it just balances the two terms. Although the larger λ results in the smaller L_0 gradient value of the output image, the explicit relation between them is unclear, and users cannot specify the L_0 gradient value of the output image in advance. Let us give an example: we show two input images (almost the same resolution) in Fig. 1(a), and the output images generated by the L_0 gradient minimization with λ adjusted so that both images achieve almost the same L_0 gradient value in Fig. 1(b).¹ One sees that λ is quite different for each image, which means that users cannot impose the same degree of flatness for different input images by simply setting the same value of λ .



Fig. 1. L_0 gradient minimization versus L_0 gradient projection: we show the L_0 gradient values (Grad_{L₀}) and the parameters used.

To resolve the difficulty, we propose an edge-preserving filtering method based on a new use of L_0 gradient, termed as the L_0 gradient projection. Specifically, we formulate the filtering as a constrained optimization, where the quadratic data-fidelity to an input image is minimized subject to the constraint that the L_0 gradient value of the output image is less than a user-given parameter α . In contrast to the L_0 gradient minimization, the parameter controlling the degree of flatness in the L_0 gradient projection, i.e., α , is the L_0 gradient value of the output image itself, so that users can directly impose a desired degree of flatness by α (see Fig. 1(c)). Moreover, users can determine α based on the number of the pixels or the L_0 gradient value of the input image.² We also develop an efficient algorithm based on the so-called alternating direction method of multipliers (ADMM) [16–18] for solving the L_0 gradient projection, where the problem is split into two subproblems, and they are solved alternately with a dual variable update. Although one of the subproblems is still a constrained nonconvex optimization problem, we show that a closed-form solution is available, yielding an efficient algorithmic solution to the L_0 gradient projection. Experimental results demonstrate the utility of the L_0 gradient projection.

2. PRELIMINARIES

For notational convenience, we treat a color image of size $N_v \times N_h$ as a vector $\mathbf{u} \in \mathbb{R}^{3N}$ $(N = N_v N_h)$ by stacking its columns on top

The work was partially supported by JSPS Grants-in-Aid (15H06197) and JST-PRESTO.

¹We used the algorithm proposed in [9] for optimization.

²Such advantages of constrained formulation over unconstrained one in terms of parameter setting have also been addressed in the literature of image restoration based on convex optimization [10–15].

of one another, in the order of the R, G and B channels.

2.1. L_0 gradient

For a given color image \mathbf{u} , the L_0 gradient [1] is defined by

$$\operatorname{Grad}_{L_0}(\mathbf{u}) := \sum_{i=1}^{N_v} \sum_{j=1}^{N_h} F\left(\sum_{c=1}^3 \left(|u_{i+1,j,c} - u_{i,j,c}| + |u_{i,j+1,c} - u_{i,j,c}| \right) \right),$$

where $u_{i,j,c}$ denotes the *c*th channel component of the pixel at (i, j), and F(x) := 1, if $x \neq 0$; F(x) := 0, otherwise. Note that $|u_{i+1,j,c} - u_{i,j,c}| := 0$ ($|u_{i,j+1,c} - u_{i,j,c}| := 0$) if $i + 1 > N_v$ $(j + 1 > N_h)$. In a word, $\operatorname{Grad}_{L_0}$ equals to the number of pixels whose vertical and/or horizontal difference is nonzero, and thus it quantifies the degree of flatness of **u**.

2.2. Alternating direction method of multipliers (ADMM)

The Alternating direction method of multipliers (ADMM) [16–18] is an algorithm that can solve the following optimization problem:

$$\min_{\mathbf{x},\mathbf{y}} f(\mathbf{x}) + g(\mathbf{y}) \text{ subject to } \mathbf{y} = \mathbf{L}\mathbf{x},$$
(1)

by iterating

$$\mathbf{x}^{(n+1)} = \underset{\mathbf{x}}{\operatorname{argmin}} f(\mathbf{x}) + \frac{1}{2\gamma} \|\mathbf{y}^{(n)} - \mathbf{L}\mathbf{x} - \mathbf{z}^{(n)}\|^{2}$$
$$\mathbf{y}^{(n+1)} = \underset{\mathbf{y}}{\operatorname{argmin}} g(\mathbf{y}) + \frac{1}{2\gamma} \|\mathbf{y} - \mathbf{L}\mathbf{x}^{(n+1)} - \mathbf{z}^{(n)}\|^{2} \qquad (2)$$
$$\mathbf{z}^{(n+1)} \leftarrow \mathbf{z}^{(n)} + \mathbf{L}\mathbf{x}^{(n+1)} - \mathbf{y}^{(n+1)}.$$

Although ADMM was developed for convex optimization, it works well in practice for nonconvex optimization, as reported in [18–23].

3. PROPOSED METHOD

3.1. Formulation of L_0 gradient projection

For a given input image $\bar{\mathbf{u}} \in \mathbb{R}^{3N}$, the L_0 gradient projection is formulated as follows:

Find
$$\mathbf{u}^{\star} \in \underset{\mathbf{u} \in \mathbb{R}^{3N}}{\operatorname{argmin}} \frac{1}{2} \|\mathbf{u} - \bar{\mathbf{u}}\|^2$$
 subject to $\operatorname{Grad}_{L_0}(\mathbf{u}) \le \alpha$, (3)

where α is a user-given parameter being the least upper bound of $\operatorname{Grad}_{L_0}(\mathbf{u}^*)$, i.e., a desired degree of flatness imposed on the output image \mathbf{u}^* , and the cost function is the quadratic data-fidelity to the input image $\bar{\mathbf{u}}$. If we set $\alpha \geq \operatorname{Grad}_{L_0}(\bar{\mathbf{u}})$, then the optimal solution of Prob. (3) equals to $\bar{\mathbf{u}}$ itself. Otherwise, the optimal solution(s) of Prob. (3) must be as close to $\bar{\mathbf{u}}$ as possible, so that \mathbf{u}^* would satisfy $\operatorname{Grad}_{L_0}(\mathbf{u}^*) = \alpha$. Hence, it characterizes the best approximation of the input image $\bar{\mathbf{u}}$ within a user-given degree of flatness α .

Remark 1 (Comparison with L_0 gradient minimization). The L_0 gradient minimization [1] is formulated as follows:

Find
$$\mathbf{u}_{\min}^{\star} \in \operatorname*{argmin}_{\mathbf{u} \in \mathbb{R}^{3N}} \lambda \operatorname{Grad}_{L_0}(\mathbf{u}) + \frac{1}{2} \|\mathbf{u} - \bar{\mathbf{u}}\|^2,$$
 (4)

where $\lambda > 0$ is a user-given parameter that balances the two terms in (4). As addressed in Sec. 1, the parameter λ does not directly correspond to the degree of flatness of \mathbf{u}_{\min}^* , meaning that one cannot specify $\operatorname{Grad}_{L_0}(\mathbf{u}_{\min}^*)$ in advance. In addition, $\operatorname{Grad}_{L_0}(\mathbf{u}_{\min}^*)$ varies depending on $\bar{\mathbf{u}}$ even if λ is fixed at a certain value. By contrast, the parameter α in Prob. (3) forces \mathbf{u}^* to satisfy $\operatorname{Grad}_{L_0}(\mathbf{u}^*) = \alpha$, so that one can always obtain the output image of the desired degree of flatness, independent of $\bar{\mathbf{u}}$, as shown in Fig. 1(c). In addition, one can set α using the information on $\bar{\mathbf{u}}$, such as a certain percentage of the number of pixels N or $\operatorname{Grad}_{L_0}(\bar{\mathbf{u}})$.

3.2. Optimization of L₀ gradient projection

Since Prob. (3) is nonconvex due to the L_0 gradient, we establish an efficient algorithm to approximately solve Prob. (3) based on ADMM by reformulating the problem into (1). To this end, we start with introducing another expression of the L_0 gradient. Let $\mathbf{u} \in \mathbb{R}^{3N}$ be a color image, and $\mathbf{D} \in \mathbb{R}^{6N \times 3N}$ be a discrete difference operator with periodic boundary that maps all the channels of a color image to their vertical and horizontal discrete differences. We also define the mixed $L_{1,0}$ pseudo-norm as follows.

Definition 1 (Mixed $L_{1,0}$ pseudo-norm). Let y be a vector of \mathbb{R}^M , and let $\mathcal{G}_1, \ldots, \mathcal{G}_K$ $(1 \le K \le M)$ be index sets such that

- Each \mathcal{G}_k is a subset of $\{1, \ldots, M\}$.
- $\mathcal{G}_k \cap \mathcal{G}_l = \emptyset$ for any $k \neq l$.
- $\cup_{k=1}^{K} \mathcal{G}_k = \{1, \ldots, M\}.$

Suppose that $\mathbf{y}_{\mathcal{G}_k}$ $(k \in \{1, \ldots, K\})$ denotes a subvector of \mathbf{y} with the entries specified by \mathcal{G}_k . Then, the mixed $L_{1,0}$ pseudo-norm of \mathbf{y} is defined as

$$\|\mathbf{y}\|_{1,0}^{\mathcal{G}} := \|(\|\mathbf{y}_{\mathcal{G}_1}\|_1, \cdots, \|\mathbf{y}_{\mathcal{G}_K}\|_1)\|_0,$$
(5)

where $\|\cdot\|_1$ and $\|\cdot\|_0$ denote the L_1 norm and the L_0 pseudo-norm, respectively.

In short, the mixed $L_{1,0}$ pseudo-norm counts the number of subvectors whose L_1 norm values are nonzero. Then, another expression of the L_0 gradient is given by

$$\operatorname{Grad}_{L_0}(\mathbf{u}) = \|\mathbf{M}\mathbf{D}\mathbf{u}\|_{1,0}^{\mathcal{G}'},\tag{6}$$

where $\mathbf{M} \in \mathbb{R}^{6N \times 6N}$ is a diagonal matrix with binary entries (0 or 1) that forces discrete differences between opposite boundaries (due to the periodic boundary condition of **D**) to be zero. Here, the number of the subvectors equals to N, i.e., $\mathcal{G}'_1, \ldots, \mathcal{G}'_N$, and each \mathcal{G}'_k contains the indices corresponding to the vertical and horizontal differences at the *k*th pixel (to avoid confusion, we use \mathcal{G}' to represent the specific index sets for the L_0 gradient, instead of \mathcal{G}).

Using the expression in (6) and introducing an auxiliary variable $\mathbf{v} = \mathbf{D}\mathbf{u}$, we can rewrite Prob. (3) as

Find
$$\mathbf{u}^{\star} \in \underset{\mathbf{u} \in \mathbb{R}^{3N}}{\operatorname{argmin}} \frac{1}{2} \|\mathbf{u} - \bar{\mathbf{u}}\|^2$$

subject to $\|\mathbf{M}\mathbf{v}\|_{1,0}^{\mathcal{G}'} \leq \alpha$ and $\mathbf{v} = \mathbf{D}\mathbf{u}$. (7)

Next, we define the indicator function of the inequality constraint on the mixed $L_{1,0}$ pseudo-norm composed with the operator M:

$$\mathbb{E}_{\{\|\mathbf{M}\cdot\|_{1,0}^{\mathcal{G}} \le \alpha\}}(\mathbf{y}) := \begin{cases} 0, & \|\mathbf{M}\mathbf{y}\|_{1,0}^{\mathcal{G}} \le \alpha, \\ \infty, & \text{otherwise.} \end{cases}$$
(8)

Then, Prob. (7) is further reformulated as follows:

$$\mathbf{u}^{\star}, \mathbf{v}^{\star}) \in \underset{\mathbf{v} \in \mathbb{R}^{3N}}{\operatorname{argmin}} \frac{1}{2} \|\mathbf{u} - \bar{\mathbf{u}}\|^{2} + \iota_{\{\|\mathbf{M}\cdot\|_{1,0}^{\mathcal{G}'} \leq \alpha\}}(\mathbf{v})$$

$$\mathbf{v} \in \mathbb{R}^{6N}$$
subject to $\mathbf{v} = \mathbf{D}\mathbf{u}.$ (9)

Find (

3.2.1. Algorithm

Now one can see that Prob. (9) is identical to Prob. (1) by letting $f := \|\cdot -\bar{\mathbf{u}}\|^2$, $g := \iota_{\{\|\mathbf{M}\cdot\|_{1,0}^{G'} \leq \alpha\}}$ and $\mathbf{L} := \mathbf{D}$. Thus, we can apply ADMM to Prob. (9), yielding an abstract version of our algorithm: for arbitrarily chosen $\mathbf{v}^{(0)}$, $\mathbf{w}^{(0)}$ and $\gamma > 0$, the algorithm iterates

$$\mathbf{u}^{(n+1)} = \underset{\mathbf{u}\in\mathbb{R}^{3N}}{\operatorname{argmin}} \frac{1}{2} \|\mathbf{u} - \bar{\mathbf{u}}\|^{2} + \frac{1}{2\gamma} \|\mathbf{v}^{(n)} - \mathbf{D}\mathbf{u} - \mathbf{w}^{(n)}\|^{2}$$
$$\mathbf{v}^{(n+1)} = \underset{\mathbf{v}\in\mathbb{R}^{6N}}{\operatorname{argmin}} \iota_{\{\|\mathbf{M}\cdot\|_{1,0}^{\mathcal{G}'} \leq \alpha\}}(\mathbf{v}) + \frac{1}{2\gamma} \|\mathbf{v} - \mathbf{D}\mathbf{u}^{(n+1)} - \mathbf{w}^{(n)}\|^{2}$$
$$\mathbf{w}^{(n+1)} \leftarrow \mathbf{w}^{(n)} + \mathbf{D}\mathbf{u}^{(n+1)} - \mathbf{v}^{(n+1)}.$$
(10)

In what follows, we show that closed-form solutions to the subproblems in (10) are available, and describe the detailed procedures of our algorithm.

Since the first subproblem, the update of $\mathbf{u}^{(n)}$, is a strictly convex quadratic minimization, it boils down to solving a system of linear equations, yielding

$$\mathbf{u}^{(n+1)} = (\mathbf{I} + \gamma^{-1} \mathbf{D}^{\mathsf{T}} \mathbf{D})^{-1} (\bar{\mathbf{u}} + \gamma^{-1} \mathbf{D}^{\mathsf{T}} (\mathbf{v}^{(n)} - \mathbf{w}^{(n)})).$$
(11)

Thanks to the periodic boundary condition of \mathbf{D} , the matrix inversion in (11) can be computed efficiently via the 2D fast Fourier transform, i.e., the inversion in Eq. (11) can be calculated as

$$\mathbf{u}^{(n+1)} = \mathbf{F}^* (\mathbf{I} + \gamma^{-1} \mathbf{\Lambda})^{-1} \mathbf{F} (\bar{\mathbf{u}} + \gamma^{-1} \mathbf{D}^\top (\mathbf{v}^{(n)} - \mathbf{w}^{(n)})),$$
(12)

where **F** and **F**^{*} are the 2D discrete Fourier transform matrix and its inverse, respectively, and **A** is a diagonal matrix with its entries being the Fourier-transformed Laplacian filter kernel. Since $\mathbf{I} + \gamma^{-1} \mathbf{A}$ is a diagonal matrix, its inversion is reduced to entry-wise division.

By noticing the definition of the indicator function in (8), the second subproblem, the update of $\mathbf{v}^{(n)}$, can be rewritten as follows:

$$\mathbf{v}^{(n+1)} = \underset{\mathbf{v} \in \mathbb{R}^{6N}}{\operatorname{argmin}} \|\mathbf{v} - \mathbf{D}\mathbf{u}^{(n+1)} - \mathbf{w}^{(n)}\|^2$$

subject to $\|\mathbf{M}\mathbf{v}\|_{1,0}^{\mathcal{G}'} \le \alpha$, (13)

where the weight $\frac{1}{2\gamma}$ is removed since it is irrelevant to the optimization. Prob. (13) might appear to be difficult, but fortunately its optimal solution can be computed in a closed form, which is guaranteed by the following result.

Proposition 1 (Projection onto mixed $L_{1,0}$ pseudo-norm constraint with binary mask). Let \mathbf{z} be a vector of \mathbb{R}^M , let α be a positive integer, let $\mathbf{S} \in \mathbb{R}^{M \times M}$ be a diagonal matrix with its diagonal entries being binary. and let $\mathcal{G}_1, \ldots, \mathcal{G}_K$ ($1 \leq K \leq M$) be index sets satisfying the conditions in Definition. 1. Without loss of generality, we can assume that $\mathbf{S}\mathbf{z} = (\mathbf{z}_{\mathcal{G}_1}^\top \cdots \mathbf{z}_{\mathcal{G}_K}^\top)^\top$. In addition, we denote the subvectors $\mathbf{z}_{\mathcal{G}_1}, \ldots, \mathbf{z}_{\mathcal{G}_K}$ sorted in descending order in terms of their L_2 norm values by $\mathbf{z}_{\mathcal{G}_{(1)}}, \ldots, \mathbf{z}_{\mathcal{G}_{(K)}}$, i.e, $\|\mathbf{z}_{\mathcal{G}_{(1)}}\| \geq \|\mathbf{z}_{\mathcal{G}_{(2)}}\| \geq \cdots \geq \|\mathbf{z}_{\mathcal{G}_{(K)}}\|$. Consider the problem:

Find
$$\mathbf{y}^* \in \underset{\mathbf{y} \in \mathbb{R}^M}{\operatorname{argmin}} \|\mathbf{y} - \mathbf{z}\|^2$$
 subject to $\|\mathbf{S}\mathbf{y}\|_{1,0}^{\mathcal{G}} \le \alpha.$ (14)

Then, one of the optimal solutions of the problem is given by

$$\mathbf{y}^{\star} = \begin{cases} \mathbf{z}, & \text{if } \|\mathbf{S}\mathbf{z}\|_{1,0}^{\mathcal{G}} \leq \alpha, \\ (\tilde{\mathbf{z}}_{\mathcal{G}_{1}}^{\top} \cdots \tilde{\mathbf{z}}_{\mathcal{G}_{K}}^{\top})^{\top} + (\mathbf{I} - \mathbf{S})\mathbf{z}, & \text{if } \|\mathbf{S}\mathbf{z}\|_{1,0}^{\mathcal{G}} > \alpha, \end{cases}$$
(15)

Algorithm 1: L_0 gradient projection by ADMM input : $\bar{\mathbf{u}}, \mathbf{v}^{(0)} = \mathbf{w}^{(0)} = \mathbf{D}\bar{\mathbf{u}}, \gamma > 0$, and $0 < \eta < 1$ while $|\operatorname{Grad}_{L_0}(\mathbf{u}^{(n)}) - \alpha| > \varepsilon$ do

$$\mathbf{u}^{(n+1)} = \mathbf{F}^* (\mathbf{I} + \gamma^{-1} \mathbf{\Lambda})^{-1} \mathbf{F} (\bar{\mathbf{u}} + \mathbf{D}^\top (\mathbf{v}^{(n)} - \mathbf{w}^{(n)}));$$

$$\mathbf{v}^{(n+1)} = \mathbf{M} (\mathbf{D} \mathbf{u}^{(n+1)} - \mathbf{w}^{(n)});$$

Compute (1),..., (N) by sorting $\mathbf{v}_{\mathcal{G}_1'}^{(n+1)}$,..., $\mathbf{v}_{\mathcal{G}_N'}^{(n+1)}$ in
descending order in terms of their L_2 norm values;
Set $\mathbf{v}_{\mathcal{G}_{(\alpha+1)}}^{(n+1)} = \mathbf{0}$,..., $\mathbf{v}_{\mathcal{G}_N'}^{(n+1)} = \mathbf{0}$ in $\mathbf{v}^{(n+1)};$

$$\mathbf{v}^{(n+1)} \leftarrow \mathbf{v}^{(n+1)} + (\mathbf{I} - \mathbf{M}) (\mathbf{D} \mathbf{u}^{(n+1)} - \mathbf{w}^{(n)});$$

$$\mathbf{w}^{(n+1)} = \mathbf{w}^{(n)} + \mathbf{D} \mathbf{u}^{(n+1)} - \mathbf{v}^{(n+1)};$$

 $\gamma \leftarrow \eta \gamma;$
 $n \leftarrow n + 1;$
output: $\mathbf{u}^{(n)}$

where

$$\tilde{\mathbf{z}}_{\mathcal{G}_k} := \begin{cases} \mathbf{z}_{\mathcal{G}_k}, & \text{if } k \in \{(1), \dots, (\alpha)\}, \\ \mathbf{0}, & \text{if } k \in \{(\alpha+1), \dots, (K)\}. \end{cases}$$
(16)

Due to the space limit, we leave the proof to the journal version of the paper [24].³

Recalling that **M** in (6) is a special case of **S**, the above result states that the closed-form solution to the second subproblem is available as follows: (i) compute $(1), \ldots, (N)$ by sorting the subvectors of $\mathbf{M}(\mathbf{Du}^{(n+1)} - \mathbf{w}^{(n)})$ in terms of their L_2 norm values; (ii) substitute zero vectors for the subvectors specified by $\mathcal{G}'_{(\alpha+1)}, \ldots, \mathcal{G}'_{(N)}$, and then (iii) add $(\mathbf{I} - \mathbf{M})(\mathbf{Du}^{(n+1)} - \mathbf{w}^{(n)})$.

Finally, our algorithm is detailed in Algorithm 1. In the algorithm, a scalar η is introduced to gradually decreases the value of γ (we recommend $\eta \in [0.95, 0.99]$), which stabilizes ADMM for nonconvex optimization. Similar strategies are also employed in existing L_0 gradient minimization algorithms [1,7–9].

We will show in Sec. 4 that $|\operatorname{Grad}_{L_0}(\mathbf{u}^{(n)}) - \alpha|$ decreases monotonically by the algorithm, so that the L_0 gradient value of the output image is expected to be α , where the stopping criterion ε determines the allowable error from α .

Remark 2 (Computational cost of Algorithm 1). At the update of $\mathbf{u}^{(n+1)}$, we can use FFT to solve the matrix inversion efficiently, and thus the cost is $\mathcal{O}(N \log N)$ time. At the update of $\mathbf{v}^{(n+1)}$, the sorting of the L_2 norm values of N subvectors is most expensive, which requires $\mathcal{O}(N \log N)$ time. The cost of the update of $\mathbf{w}^{(n+1)}$ is simply $\mathcal{O}(N)$ time. As a result, the cost of each iteration of Algorithm 1 is $\mathcal{O}(N \log N)$ time.

4. EXPERIMENTS

To illustrate the utility of the L_0 gradient projection, we conducted experiments on image smoothing with various α . Specifically, we compare output images generated with α set to 16%, 8%, 4% and 2% of the number of pixels of input images. The input images are of size (approximately) 640×420 , which were took from a large database of royalty-free images [25]. All experiments were performed using

³In [24], we also provide (i) deeper discussions on our method, including the effectiveness of our algorithm in the sense of the L_0 gradient minimization and the relation to existing L_0 gradient minimization algorithms; and (ii) extensive experiments, where we compare our algorithm with several L_0 gradient minimization algorithms and present illustrative applications.



Fig. 2. Input images (the left column), and the output images generated by L_0 gradient projection with α set to various percentages of the number of pixels (we also show PSNR[dB] between input and output images for reference).



(a) CPU time versus gap from α (b) CPU time versus PSNR[dB]

Fig. 3. Convergence profiles of our algorithm on computing the output images shown the top row of Fig. 2.

MATLAB (R2014a, 64bit), on a Windows 10 (64bit) laptop computer with an Intel Core i7 2.6 GHz processor and 8 GB of RAM. For the parameters of Algorithm 1, we set $\gamma = 3$, $\eta = 0.97$, and $\varepsilon = 0.0002N$ in all the experiments.

The results are shown in Fig. 2, where for each row, the left is an input image, and the others are the output images with their PSNR[dB]. One sees that setting α to the same percentage of N (each column of Fig. 2) generates the output images of the same degree of flatness, which offers an easy setting of α in many applications, such as segmentation, character recognition, and color quantization.

Fig. 3(a) depicts convergence plots of the gap from α (on Par-

rot), i.e., $|\text{Grad}_{L_0}(\mathbf{u}^{(n)}) - \alpha|$, versus the CPU time. We observe that for every α , the gap decreases monotonically and converges to zero, implying that output images satisfies user-given L_0 gradient values. Fig. 3(b) shows the evolution of PSNR between $\mathbf{u}^{(n)}$ and $\bar{\mathbf{u}}$, i.e., the convergence of the cost function (quadratic data-fidelity), versus the CPU time. One can see that PSNR also converges to a certain value for every α , The reason why PSNR in early iterations is higher than the converged value is that the initial variable $\mathbf{u}^{(0)}$ is set to the input image $\bar{\mathbf{u}}$. These observations demonstrate that our algorithm works well in practice for the L_0 gradient projection.

5. CONCLUSION

We have proposed a new edge-preserving filtering method based on the L_0 gradient, named the L_0 gradient projection. In contrast to the L_0 gradient minimization, our L_0 gradient projection framework is very intuitive because one can directly impose a desired degree of flatness (i.e, L_0 gradient value) on the output image. We established an ADMM-based algorithm that can solve the L_0 gradient projection in $\mathcal{O}(N \log N)$ time, and we empirically showed that filtered images generated by the algorithm satisfies user-given L_0 gradient values. We expect that our framework facilitates the use of L_0 gradient-based image flattening in a variety of applications.

6. REFERENCES

- [1] Li Xu, C. Lu, Yi Xu, and J. Jia, "Image smoothing via L_0 gradient minimization," *ACM Trans. Graph.*, vol. 30, no. 6, pp. 174:1–174:12, 2011.
- [2] S.-C. Pei, C.-T. Shen, and T.-Y. Lee, "Visual enhancement using constrained L₀ gradient image decomposition for low backlight displays," *IEEE Signal Process. Lett.*, vol. 19, no. 12, pp. 813–816, 2012.
- [3] Li Xu, S. Zheng, and J. Jia, "Unnatural L₀ sparse representation for natural image deblurring," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2013, pp. 1107–1114.
- [4] L. He and S. Schaefer, "Mesh denoising via L_0 minimization," *ACM Trans. Graph.*, vol. 32, no. 4, pp. 64:1–64:8, 2013.
- [5] J. Pan, Z. Hu, Z. Su, and M.-H. Yang, "Deblurring text images via L₀regularized intensity and gradient prior," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2014, pp. 2901–2908.
- [6] M. Storath, A. Weinmann, J. Frikel, and M. Unser, "Joint image reconstruction and segmentation using the potts model," *Inverse Problems*, vol. 31, no. 2, 2015, article ID: 025003.
- [7] X. Cheng, M. Zeng, and X. Liu, "Feature-preserving filtering with L₀ gradient minimization," *Computers & Graphics*, vol. 38, pp. 150–157, 2014.
- [8] M. Storath, A. Weinmann, and L. Demaret, "Jump-sparse and sparse recovery using potts functionals," *IEEE Trans. Signal Process.*, vol. 62, no. 14, pp. 3654–3666, 2014.
- [9] R. Nguyen and M. Brown, "Fast and effective L_0 gradient minimization by region fusion," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, 2015, pp. 208–216.
- [10] P.L. Combettes and J.-C. Pesquet, "Image restoration subject to a total variation constraint," *IEEE Trans. Image Process.*, vol. 13, no. 9, pp. 1213–1222, 2004.
- [11] J. M. Fadili and G. Peyré, "Total variation projection with first order schemes," *IEEE Trans. Image Process.*, vol. 20, no. 3, pp. 657–669, 2011.
- [12] M. Afonso, J. Bioucas-Dias, and M. Figueiredo, "An augmented Lagrangian approach to the constrained optimization formulation of imaging inverse problems," *IEEE Trans. Image Process.*, vol. 20, no. 3, pp. 681–695, 2011.
- [13] S. Ono and I. Yamada, "Second-order total generalized variation constraint," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.* (ICASSP), 2014, pp. 4938–4942.
- [14] G. Chierchia, N. Pustelnik, J.-C. Pesquet, and B. Pesquet-Popescu, "Epigraphical projection and proximal tools for solving constrained convex optimization problems," *Signal, Image and Video Process.*, vol. 9, no. 8, pp. 1737–1749, 2015.
- [15] S. Ono and I. Yamada, "Signal recovery with certain involved convex data-fidelity constraints," *IEEE Trans. Signal Process.*, vol. 63, no. 22, pp. 6149–6163, 2015.
- [16] D. Gabay and B. Mercier, "A dual algorithm for the solution of nonlinear variational problems via finite elements approximations," *Comput. Math. Appl.*, vol. 2, pp. 17–40, 1976.
- [17] 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.
- [18] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2011.
- [19] Z. Wen, C. Yang, X. Liu, and S. Marchesini, "Alternating direction methods for classical and ptychographic phase retrieval," *Inverse Problems*, vol. 28, no. 11, 2012, article ID: 115010.
- [20] D. L. Sun and C. Fevotte, "Alternating direction method of multipliers for non-negative matrix factorization with the beta-divergence," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, 2014, pp. 6201–6205.

- [21] F. Heide, W. Heidrich, and G. Wetzstein, "Fast and flexible convolutional sparse coding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2015, pp. 5135–5143.
- [22] B. Zhang, A. Perina, V. Murino, and A. Del Bue, "Sparse representation classification with manifold constraints transfer," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2015, pp. 4557–4565.
- [23] S. Ono and I. Yamada, "Color-line regularization for color artifact removal," *IEEE Trans. Comput. Imag.*, vol. 2, no. 3, pp. 204–217, 2016.
- [24] S. Ono, " L_0 gradient projection," *IEEE Trans. Image Process.*, 2017, 11 pages, accepted for publication.
- [25] "Pixabay," https://pixabay.com/en/.